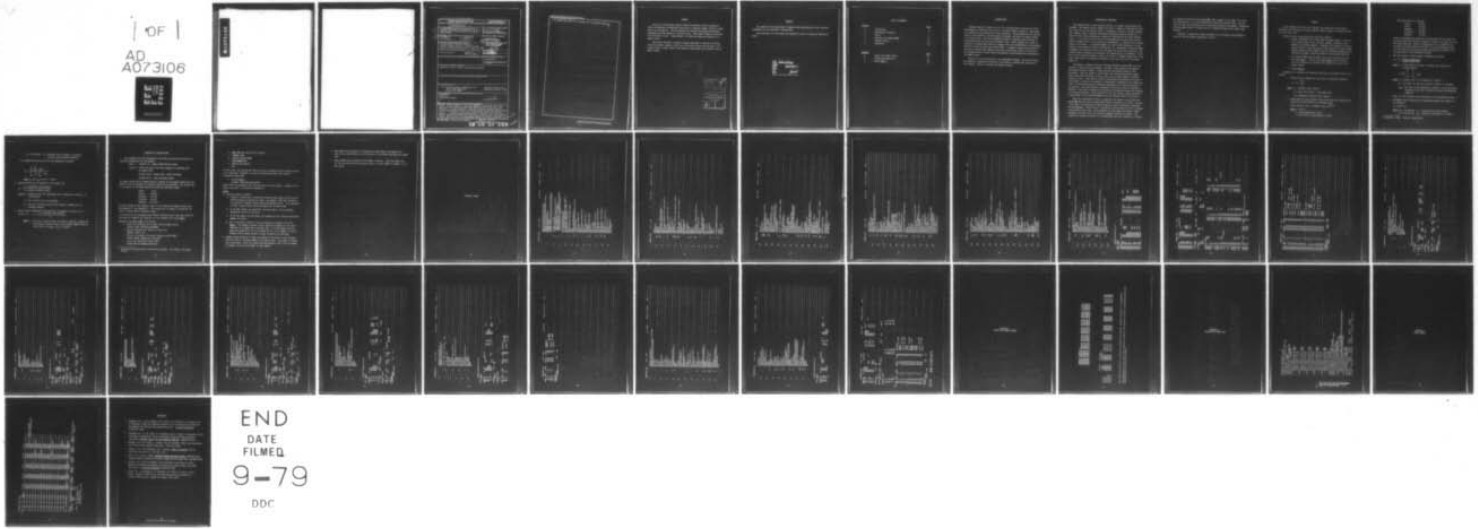


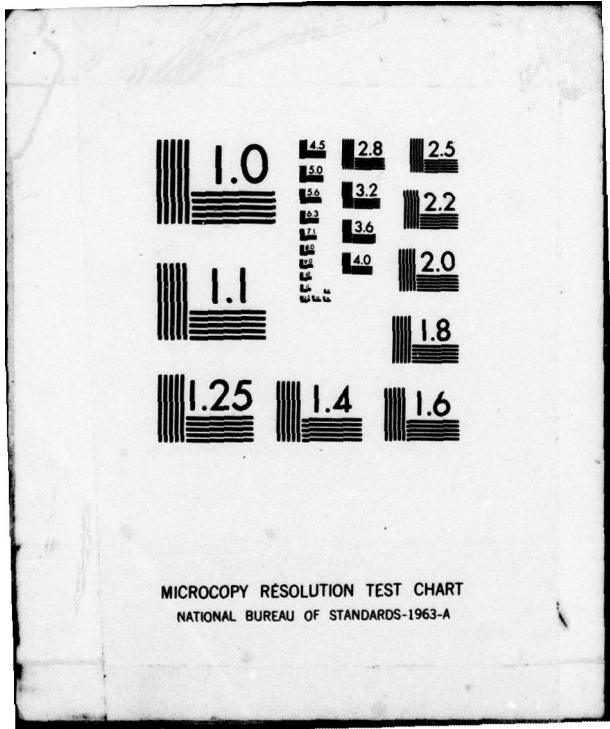
AD-A073 106

AEROSPACE MEDICAL RESEARCH LAB WRIGHT-PATTERSON AFB OH F/G 6/16
A COMPUTERIZED METHOD FOR DETERMINATION OF REGIONAL BLOOD FLOW --ETC(U)
MAR 79 K C SMITH, S WARD, K J GREENLEES F33615-76-C-5001
AMRL-TR-79-1 NL

UNCLASSIFIED

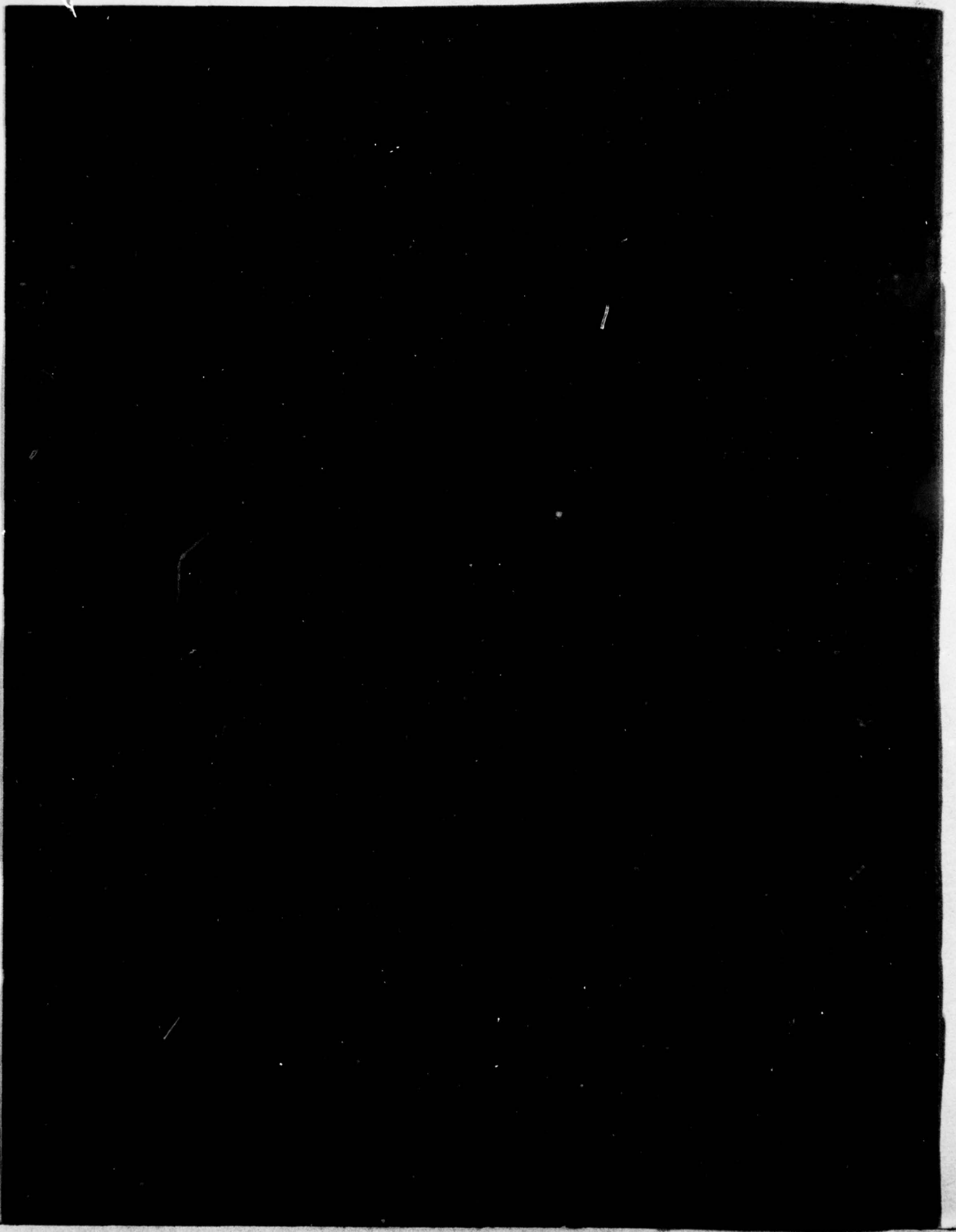
OF
AD
A073106
SERIAL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD A 073106



REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER (14) AMRL-TR-79-1	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) (6) A COMPUTERIZED METHOD FOR DETERMINATION OF REGIONAL BLOOD FLOW USING RADIOACTIVE TRACER MICROSPHERES	5. TYPE OF REPORT & PERIOD COVERED (9) Technical Report	
7. AUTHOR(s) (10) K. C. Smith S. Ward* K. J. Greenlees C. M. Oloff	8. CONTRACT OR GRANT NUMBER(s) (15) In part under F33615-76-C-5001	
9. PERFORMING ORGANIZATION NAME AND ADDRESS Aerospace Medical Research Laboratory Aerospace Medical Division, Air Force Systems Command, Wright-Patterson Air Force Base, OH. 45433	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62202F, 7222-10-50 (16) (17)	
11. CONTROLLING OFFICE NAME AND ADDRESS (Same as Block 9) (12) 42p.	12. REPORT DATE (11) Mar 1979	
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	13. NUMBER OF PAGES 42	
	15. SECURITY CLASS. (of this report) Unclassified	
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE	
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES *From Systems Research Laboratories, Inc. 2800 Indian Ripple Road Dayton, Ohio 45440 Supported in part by the Laboratory Director's Fund.		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Microspheres Blood flow Computerized program (18) +G sub 2		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) 2 Studies utilizing radioactive tracer microspheres to determine blood flow to a specific region, tissue, or organ under the acceleration were done in this Laboratory. The microspheres were injected into animal subjects and were distributed throughout the body. They were trapped in capillary beds according to size. Tissue samples from areas of interest were taken, and radiation measured. A computer program was developed to calculate corrected counts per minute, regional blood flow, and cardiac output from raw data. This report includes a brief outline of the experimental design, the theory behind the		

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

Block 20. Abstract (cont'd)

9 program, execution of the program, and the program listing. 4

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

SUMMARY

Studies in the Aerospace Medical Research Laboratory utilize radioactive tracer microspheres to determine blood flow to a specific region, tissue or organ under $+G_2$ acceleration. The microspheres are injected and become distributed throughout the body similar to red blood cells. They become trapped in capillary beds according to size. Tissue samples are taken and radiation measured in a gamma counter.

This report describes a computer program developed to calculate corrected counts per minute, regional blood flow, and cardiac output from raw data. The program offers a rapid, easily used method for processing data obtained from the gamma counter.

preliminary
check
copy

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DDC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution/	
Availability Codes	
Dist.	Avail and/or special
A	

PREFACE

The authors of this paper wish to express their appreciation to Mrs. Patricia Lewandowski for her editorial contributions.

Work reported on in this paper was supported in part by Laboratory Director's Funds.

A *laboratory*
M *director's*
R *fund*
L

TABLE OF CONTENTS

<u>Section</u>		<u>Page</u>
1	Introduction	4
2	Experimental Procedure	5
3	Theory	7
4	Execution of Program GAMMA	10
5	Program Listing	13
6	References	38
<u>Appendix</u>		<u>Page</u>
A	Output from Gamma Counter	32
B	Abbreviated Sample Run	34
C	Sample <u>Tape 2</u>	36

INTRODUCTION

Studies similar to those of Sostre (7) utilizing plastic radioactive tracer microspheres to calculate blood flow to a specific region, tissue, or organ under $+G_z$ acceleration are being done in this Laboratory. The microspheres are injected into the left ventricle, and distributed throughout the body in a manner similar to red blood cells. They are entrapped in specific capillary beds according to blood distribution and the size microsphere used. Tissue samples from areas of interest are weighed and placed in a gamma counter for determination of microsphere density in terms of counts per minute (CPM). A computer program was developed to calculate corrected CPM, regional blood flow, and cardiac output from raw data. This program offers a rapid, easily used method for processing data obtained from the gamma counter.

Section 2 is a brief outline of the experimental design. The theory behind the program is presented in Section 3, and Section 4 contains the execution of the program. Section 5 includes the program listing.

EXPERIMENTAL PROCEDURE

The animals used, in this instance baboons, are lightly anesthetized with ketamine. Two femoral artery cutdowns are made; a catheter is inserted into the left ventricle via one femoral artery. Microsphere injections are made through this catheter using a remotely controlled microsphere injection system (2). A small silicone coated polyethylene catheter is inserted into the other femoral artery. It is placed with the tip in the descending aorta just below the arch. Fifteen seconds prior to the injection of microspheres, a constant rate reference sample is withdrawn. A Harvard Pump $\text{\textcircled{R}}$ (7) is used in this Laboratory, but any constant rate withdrawal system is acceptable. (6) Withdrawal of blood continues throughout the injection series until one minute after the final injection. The volume of blood withdrawn is calculated, and radioactivity is determined. Blood flow to this reference sample is taken to be that of a "known" organ; flow to any other organ then be calculated by relating to the blood flow of the reference sample (1).

The animal is placed in the Oloff Primate Restraint System (PRS) (5), and allowed to recover from the ketamine. The PRS is then placed on the Dynamic Environment Simulator (DES), a large man-rated centrifuge located in this Laboratory. When the animal is fully awake, the acceleration profile begins. Peak acceleration is either at $+3G_z$ or $+5G_z$, with an equal number of animals exposed to each. Five types of 15 micron radiolabeled microspheres are used for each animal: Cerium 141, Chromium 51, Strontium 85, Scandium 46, and Iodine 125. 0.025 to 0.050 microcuries are injected, depending on the relative activity of the isotope. The first isotope is injected at $+1G_z$ and is used to establish a baseline. When peak acceleration is reached, the remaining isotopes are injected at 30 second intervals. After injection of the last isotope, peak acceleration is maintained for 30 seconds bringing total $+G_z$ exposure time to 120 seconds.

When the acceleration profile is completed, the animal is sacrificed with Uthanol $\text{\textcircled{R}}$ and the brain is removed and fixed in formalin for 72 hours. One half of the brain is sectioned into various regions, such as brainstem, cerebellum, and hypothalamus. The other half is placed in several vials and used to estimate total cerebral blood flow. Tissue samples are also taken from various organs throughout the body, with each location assigned a specific ID number. All samples are weighed and placed in an appropriate vial to be counted in a Packard

Auto-Gamma Scintillation Spectrometer (R). When a sample is too large to fit into one counting vial, it can be sectioned and placed into several vials. All the vials can then be incorporated into a group and treated as one sample. Information from the gamma counter is printed out on a standard teletype with a punch tape capacity.

Disposal of radioactive animal carcasses is in accordance with Appendix B, Title 10, Code of Federal Regulations, Part 20.

THEORY

Program GAMMA was written for a Control Data Corporation 6600 computer located at the Aeronautical Systems Division Computer Center at Wright-Patterson Air Force Base, Ohio.

There are three steps involved in using this program:

1. Create a permanent file of data from the gamma counter. This file will contain counts of all reference samples, standards, assay samples, and residue samples. Standards are aliquots of the microsphere solution of the same age as those injected into the animals. Residue samples include samples from the withdrawn catheters, post-experiment syringes, injection catheters, and manifold washouts.
2. Run program GAMMA using the above file, which is given the local file name Tape 1. A local file named Tape 2 will be created, this file contains the corrected CPM, regional blood flow, and cardiac output.
3. Catalog Tape 2.

Computation of corrected CPM, regional blood flow, and cardiac output is as follows:

1. Counts from all samples are corrected for background radiation:

$$c'_i = c_i - \frac{e_i^2}{c_i}$$

where c_i = original count, window i

e_i = mean count window i, from empty vials

c'_i = background corrected count, window i

2. Counts made on days previous to the assay counts are corrected for radioactive decay during the intervening time:

$$\text{decay corrected count} = \text{original count} \times e^{-(ct/h)}$$

where $c = \ln(.5)$

t = days elapsed since count

h = half-life of given isotope, in days

Half-life values (4) used are:

Iodine	60 days
Cerium	32 days
Chromium	28 days
Strontium	64 days
Scandium	84 days

Counts of standards are corrected for decay using the same half-life for each window; the isotope is assumed to decay at the same rate over the entire spectrum. Counts of samples that are not standards are corrected for decay using the half-life of the corresponding isotope for each window; the decay rate within each window is assumed to be the decay rate of the predominant isotope.

3. The total amount of each isotope is computed as follows:

$$z_i = y_i \times \frac{\text{volume injected (ml)}}{\text{volume counted (ml)}}$$

where z_i = total amount of isotope i injected, not corrected for residue samples

$$y_i = \frac{1}{n_i} \sum_{k=1}^n \sum_{j=1}^5 c_{ijk}$$

where y_i = average count of the standard for isotope i

n_i = the number of times the count for isotope i is repeated

c_{ijk} = the count of the standard for isotope i in the jth window repeated the kth time, corrected for decay and background radiation

Thus y_i is the average of the total counts in all windows from isotope i.

4. Corrected counts (corrected for overlapping windows) are computed by the formula*:

$$T = (F'F)^{-1} F'C$$

where T is a 5x1 matrix; t_i = corrected counts from isotope i

F is a 5x5 matrix; f_{ij} = proportion of isotope i in window j

* Rosenblatt, Judah. Personal communication

C is a 5x1 matrix; c_j = observed count in window j, corrected
for decay and background radiation

F is estimated from the counts of the standards as follows:

$$f_{ij} = \frac{\frac{1}{n_i} \sum_{k=1}^{n_i} c_{ijk}}{\sum_{i=1}^5 \frac{1}{n_i} \sum_{k=1}^{n_i} c_{ijk}}$$

where n_i and c_{ijk} are as in 3 above

5. Regional blood flow is computed by the formula (5):

$$g_i = \frac{t_i \times \text{withdrawal rate (ml/min)}}{r_i \times \text{weight of sample (grams)}}$$

where g_i = regional blood flow immediately after injection of isotope i, in
[ml/min/gram]

t_i = the corrected count from isotope i

r_i = the sum of corrected counts from isotope i, summed over all
reference samples

6. Cardiac output immediately after injection of isotope i is given by the
formula (3): $s_i = \frac{w_i \times \text{withdrawal rate (ml/min)}}{r_i}$

where w_i = the total corrected count of isotope i injected, computed by
subtracting corrected counts of all residue samples from the
total count of isotope i (z_i in 3 above).

EXECUTION OF PROGRAM GAMMA

The permanent data file described in the theory section should contain the following information for each sample:

Line 1 Columns 1-6: sample identification number

Lines 2-6 Counts for each of the five isotopes in ascending order
 by energy level

 Columns 15-20: energy level, center of window

 Columns 50-55: count for given window

The sample identification number should be unique; no two samples should have the same ID, with the exception of multiple counts of the standards. The energy level at the center of the window should be within the following ranges:

Iodine 29-35

Cerium 142-148

Chromium 317-323

Strontium 511-517

Scandium 1117-1123

The above format is the standard output of the Packard Auto-Gamma Scintillation Spectrometer R (see Appendix A). The program can be changed to accommodate the use of a counter with a different format.

If the data from an experiment are on several different paper tapes they should be read into the computer and combined into one permanent file using Editor.

Execution of program GAMMA is as follows:

1. Attach the permanent file of data from the gamma counter
ATTACH, TAPE1, PFN,CY=XX,ID=XXXXXX
2. Request permanent file space for output file
REQUEST, TAPE2, *PF
3. Attach the permanent file which contains program GAMMA
ATTACH, GAMMA, WARD35,CY=1,ID=L740530
4. Attach the subroutine library IMSL*
ATTACH, IMSL, ID=LIBRARY, SN=ASD, MR=1

* International Mathematical & Statistical Libraries. ASD Computer Subprogram Library

5. Make IMSL part of the local library

LIBRARY, IMSL

6. Compile program GAMMA

FTN(I=GAMMA,L=0)

7. Execute the program

LGO

At this point, the program will ask for certain information which should be typed in as requested. Appendix B contains an abbreviated sample with typed-in information underlined.

8. Catalog Tape 2

CATALOG, TAPE2, PFN, CY=XX, RP=999

where PFN is the permanent file name and XX is the cycle number. A sample of the output from Tape 2 is included in Appendix C.

NOTES:

1. Entering sample ID numbers

- a. A group of numbers may be entered by putting a minus sign in front of the selected number satisfying the range. For example, "580 -592" specifies all thirteen ID numbers between and including 580 and 592. It is important to leave a space between the first number and the minus sign.
- b. If too many numbers are specified, the extra numbers will be dropped starting at the end of the list.
- c. If too few numbers are specified, the program will not continue until more numbers are given.
- d. Do not include ID numbers in the list which do not exist in the file Tape 1. For example, suppose a group consists of samples 33 -45, and 47 and there is no sample 46 in the data file. If the ID numbers for the group are specified as "33 -47" instead of "33 -45 47" sample 47 will be dropped from the group because too many ID numbers are specified.

2. Samples counted previous to assay are grouped by the number of days elapsed from the count of the given sample to the assay counts. Any number of groups may be specified. After all groups have been entered, enter 0 for the number of samples, and 0 for the number of days elapsed.

3. When sample ID and weight are requested for each sample, the sample ID to enter is the identification of the location in the animal from which the sample came.
4. Assay samples may be grouped in any number of groups. After the sample IDs for the last group have been entered, enter 0 for the number of samples in the next group.

PROGRAM LISTING

00 00 000000
 01 01 000000
 02 02 000000
 03 03 000000
 04 04 000000
 05 05 000000
 06 06 000000
 07 07 000000
 08 08 000000
 09 09 000000
 10 10 000000
 11 11 000000
 12 12 000000
 13 13 000000
 14 14 000000
 15 15 000000
 16 16 000000
 17 17 000000
 18 18 000000
 19 19 000000
 20 20 000000
 21 21 000000
 22 22 000000
 23 23 000000
 24 24 000000
 25 25 000000
 26 26 000000
 27 27 000000
 28 28 000000
 29 29 000000
 30 30 000000
 31 31 000000
 32 32 000000
 33 33 000000
 34 34 000000
 35 35 000000
 36 36 000000
 37 37 000000
 38 38 000000
 39 39 000000
 40 40 000000
 41 41 000000
 42 42 000000
 43 43 000000
 44 44 000000
 45 45 000000
 46 46 000000
 47 47 000000
 48 48 000000
 49 49 000000
 50 50 000000
 51 51 000000
 52 52 000000
 53 53 000000
 54 54 000000
 55 55 000000
 56 56 000000
 57 57 000000
 58 58 000000
 59 59 000000
 60 60 000000
 61 61 000000
 62 62 000000
 63 63 000000
 64 64 000000
 65 65 000000
 66 66 000000
 67 67 000000
 68 68 000000
 69 69 000000
 70 70 000000
 71 71 000000
 72 72 000000
 73 73 000000
 74 74 000000
 75 75 000000
 76 76 000000
 77 77 000000
 78 78 000000
 79 79 000000
 80 80 000000
 81 81 000000
 82 82 000000
 83 83 000000
 84 84 000000
 85 85 000000
 86 86 000000
 87 87 000000
 88 88 000000
 89 89 000000
 90 90 000000
 91 91 000000
 92 92 000000
 93 93 000000
 94 94 000000
 95 95 000000
 96 96 000000
 97 97 000000
 98 98 000000
 99 99 000000
 100 100 000000

```

1  PROGRAM GAMMA(INPUT,OUTPUT,TAPES,1APE2)
   DIMENSION BACK(5),CVOL(5),TVOL(5),Q15(5)
   DIMENSION REF(5),NS(50),KGC(200),NGP(200),KGP(200,2)
   DIMENSION KSI(50),NTOT(5)
   DIMENSION NSDEL(200),DAY(200)
   DIMENSION KDM(200),IM(5),ICT(5),LHM(5),ACT(5),RCT(5)
   DIMENSION TOT(5),SSGP(200,5),FLOM(5),KREF(20),PER(5,5)

*   TO CHANGE # ISOTOPES
*   CHANGE DATA STATEMENTS FOR LIMM, REF AND BACK
*   CHANGE DIMENSIONS OF SSGP,PER AND Q
*   CHANGE VALUE OF "NIS" -- # OF ISOTOPES
*   INCREASE DIMENSIONS OF ALL RELEVANT VARS IF NECESSARY
*   CHANGE DIMENSIONS IN SUBROUTINES INV AND ICLS

   DATA LIMM/32,145,320,514,1120/
   DATA LIMM/32,320,514,1120,0/
   DATA REF/5M125,5MCE145,5MCR141,5MSR65,5MSC46 /
   DATA REF/5M125,5MCR141,5MSR65,5MSC46,5M /
   DATA BACK/79,74,19,32,4,4/
   PRINT 1
   FORMAT(*1CORRECTED MICROSPHERE COUNTS*/)
   C  NIS=4
   C  NIS=5
   PRINT 400,NIS,(REF(I),I=1,NIS)
   FORMAT(* NUMBER OF ISOTOPES ASSUMED TO BE*,I3/
1  * ISOTOPES *,10A6)
   PRINT 5
   C  FORMAT(* ENTER SUBJECT ID *)
   READ 3,SUBJ
   PRINT 2
   FORMAT(* ENTER EXPERIMENT DATE *)
   READ 3,DATE
   PRINT 4
   FORMAT(* ENTER ASSAY DATE *)
   READ 3,ASSAY
   WRITE(2,107)SUBJ,DATE,ASSAY
   FORMAT(3A10)
   IT=0
   DO 62 I=1,NIS
   PRINT 7,I,REF(I)
   FORMAT(* ISOTOPE *,I2,1X,A5)
   PRINT 403
   FORMAT(5X,*ENTER VOLUME COUNTED *)
   READ *,CVOL(I)
   PRINT 404
   FORMAT(5X,*ENTER VOLUME INJECTED *)
   READ *,TVOL(I)
   PRINT 405
   FORMAT(5X,*ENTER # SAMPLE IDS *)
   READ*,NKS
   IF(NKS.LE.0)GO TO 62
   PRINT 14
   CALL NUMBING(KDM)
   DO 63 J=1,NKS

```

```

IT=IT+1
KSI(I)=KDUH(J)
63 KSI(I)=I
62 CONTINUE
120 PRINT 120
FORMAT(* ENTER # REFERENCE SAMPLES *)
CALL REFIO(REF,KREF,RCI,REF,NIS)
65 PRINT 6
FORMAT(* ENTER WITHDRANAL RATE, REFERENCE SAMPLE #)
READ *,WITHD
PRINT 10
10 FORMAT(* ENTER # SAMPLES FROM RESIDUE *)
READ *,NCATH
IF(NCATH.LE.0)GO TO 110
PRINT 14
CALL NUMB(NCATH,KC)
110 PRINT 106
108 FORMAT(* ENTER # SAMPLES COUNTED PREVIOUS TO ASSAY*)
I=0
53 PRINT 50
FORMAT(3X,* ENTER # SAMPLES AND # DAYS ELAPSED *)
50 READ *,NSS,DDD
IF(NSS.LE.0)GO TO 51
PRINT 14
CALL NUMB(NSS,KDUH)
DO 52 K=1,NSS
I=I+1
NSDEL(I)=KDUH(K)
52 DAY(I)=DDD
GO TO 53
51 NOEL=I
NGPS=IE=0
IB=I-1
90 PRINT 11,I
11 FORMAT(* ENTER # SAMPLES IN GROUP*,I3,I1X)
READ *,NGP(I)
IF(NGP(I).LE.0)GO TO 13
NGPS=I
PRINT 14
14 FORMAT(5X,*ENTER SAMPLE IDS *)
CALL NUMB(NGP(I),KDUH)
IE=IB-1+NGP(I)
100 K=0
DO 15 J=IB,IE
K=K+1
KGP(J,1)=KDUH(K)
15 KGP(J,2)=I
IB=IE+1
I=I+1
GO TO 16
*
*
*
110 PROCESS STANDARDS AND REFERENCE SAMPLES
REWINO 1
MREF=0
DO 130 I=1,NTS
NTOT(I)=0

```

```

115     TOT(I)=0.
116     DO 130 J=1,NIS
117     PER(I,J)=0.
118     READ(1,17)IID,((IM(I),ICT(I)),I=1,NIS)
119     FORMAT(I6,(14X,I6,29X,I6))
120     IF(EOF(1))131,132
121     * CHECK IF A STANDARD
122     * DO 133 I=1,IY
123     * IF(IID.NE.MS(I))GO TO 133
124     * CALL WINDOW(IID,IN,ICT,LIMH,NIS,BACK)
125     * IS=KST(I)
126     * IF(IS.LE.0.OR.IS.GT.NIS)STOP "INDEX"
127     * CALL HALF(IID,ICT,MODEL,NSDEL,DAY,IS,NIS)
128     * NTOT(IS)=NTOT(IS)+1
129     * DO 134 J=1,NIS
130     * * TOT(IS) = SUM FOR ISOTOPE IS OVER ALL WINDOWS
131     * *
132     * *
133     * *
134     * *
135     * *
136     * *
137     * *
138     * *
139     * *
140     * *
141     * *
142     * *
143     * *
144     * *
145     * *
146     * *
147     * *
148     * *
149     * *
150     * *
151     * *
152     * *
153     * *
154     * *
155     * *
156     * *
157     * *
158     * *
159     * *
160     * *
161     * *
162     * *
163     * *
164     * *
165     * *
166     * *
167     * *
168     * *
169     * *
170     * *
171     * *
172     * *
173     * *
174     * *
175     * *
176     * *
177     * *
178     * *
179     * *
180     * *
181     * *
182     * *
183     * *
184     * *
185     * *
186     * *
187     * *
188     * *
189     * *
190     * *
191     * *
192     * *
193     * *
194     * *
195     * *
196     * *
197     * *
198     * *
199     * *
200     * *
201     * *
202     * *
203     * *
204     * *
205     * *
206     * *
207     * *
208     * *
209     * *
210     * *
211     * *
212     * *
213     * *
214     * *
215     * *
216     * *
217     * *
218     * *
219     * *
220     * *
221     * *
222     * *
223     * *
224     * *
225     * *
226     * *
227     * *
228     * *
229     * *
230     * *
231     * *
232     * *
233     * *
234     * *
235     * *
236     * *
237     * *
238     * *
239     * *
240     * *
241     * *
242     * *
243     * *
244     * *
245     * *
246     * *
247     * *
248     * *
249     * *
250     * *
251     * *
252     * *
253     * *
254     * *
255     * *
256     * *
257     * *
258     * *
259     * *
260     * *
261     * *
262     * *
263     * *
264     * *
265     * *
266     * *
267     * *
268     * *
269     * *
270     * *
271     * *
272     * *
273     * *
274     * *
275     * *
276     * *
277     * *
278     * *
279     * *
280     * *
281     * *
282     * *
283     * *
284     * *
285     * *
286     * *
287     * *
288     * *
289     * *
290     * *
291     * *
292     * *
293     * *
294     * *
295     * *
296     * *
297     * *
298     * *
299     * *
300     * *
301     * *
302     * *
303     * *
304     * *
305     * *
306     * *
307     * *
308     * *
309     * *
310     * *
311     * *
312     * *
313     * *
314     * *
315     * *
316     * *
317     * *
318     * *
319     * *
320     * *
321     * *
322     * *
323     * *
324     * *
325     * *
326     * *
327     * *
328     * *
329     * *
330     * *
331     * *
332     * *
333     * *
334     * *
335     * *
336     * *
337     * *
338     * *
339     * *
340     * *
341     * *
342     * *
343     * *
344     * *
345     * *
346     * *
347     * *
348     * *
349     * *
350     * *
351     * *
352     * *
353     * *
354     * *
355     * *
356     * *
357     * *
358     * *
359     * *
360     * *
361     * *
362     * *
363     * *
364     * *
365     * *
366     * *
367     * *
368     * *
369     * *
370     * *
371     * *
372     * *
373     * *
374     * *
375     * *
376     * *
377     * *
378     * *
379     * *
380     * *
381     * *
382     * *
383     * *
384     * *
385     * *
386     * *
387     * *
388     * *
389     * *
390     * *
391     * *
392     * *
393     * *
394     * *
395     * *
396     * *
397     * *
398     * *
399     * *
400     * *
401     * *
402     * *
403     * *
404     * *
405     * *
406     * *
407     * *
408     * *
409     * *
410     * *
411     * *
412     * *
413     * *
414     * *
415     * *
416     * *
417     * *
418     * *
419     * *
420     * *
421     * *
422     * *
423     * *
424     * *
425     * *
426     * *
427     * *
428     * *
429     * *
430     * *
431     * *
432     * *
433     * *
434     * *
435     * *
436     * *
437     * *
438     * *
439     * *
440     * *
441     * *
442     * *
443     * *
444     * *
445     * *
446     * *
447     * *
448     * *
449     * *
450     * *
451     * *
452     * *
453     * *
454     * *
455     * *
456     * *
457     * *
458     * *
459     * *
460     * *
461     * *
462     * *
463     * *
464     * *
465     * *
466     * *
467     * *
468     * *
469     * *
470     * *
471     * *
472     * *
473     * *
474     * *
475     * *
476     * *
477     * *
478     * *
479     * *
480     * *
481     * *
482     * *
483     * *
484     * *
485     * *
486     * *
487     * *
488     * *
489     * *
490     * *
491     * *
492     * *
493     * *
494     * *
495     * *
496     * *
497     * *
498     * *
499     * *
500     * *
501     * *
502     * *
503     * *
504     * *
505     * *
506     * *
507     * *
508     * *
509     * *
510     * *
511     * *
512     * *
513     * *
514     * *
515     * *
516     * *
517     * *
518     * *
519     * *
520     * *
521     * *
522     * *
523     * *
524     * *
525     * *
526     * *
527     * *
528     * *
529     * *
530     * *
531     * *
532     * *
533     * *
534     * *
535     * *
536     * *
537     * *
538     * *
539     * *
540     * *
541     * *
542     * *
543     * *
544     * *
545     * *
546     * *
547     * *
548     * *
549     * *
550     * *
551     * *
552     * *
553     * *
554     * *
555     * *
556     * *
557     * *
558     * *
559     * *
560     * *
561     * *
562     * *
563     * *
564     * *
565     * *
566     * *
567     * *
568     * *
569     * *
570     * *
571     * *
572     * *
573     * *
574     * *
575     * *
576     * *
577     * *
578     * *
579     * *
580     * *
581     * *
582     * *
583     * *
584     * *
585     * *
586     * *
587     * *
588     * *
589     * *
590     * *
591     * *
592     * *
593     * *
594     * *
595     * *
596     * *
597     * *
598     * *
599     * *
600     * *
601     * *
602     * *
603     * *
604     * *
605     * *
606     * *
607     * *
608     * *
609     * *
610     * *
611     * *
612     * *
613     * *
614     * *
615     * *
616     * *
617     * *
618     * *
619     * *
620     * *
621     * *
622     * *
623     * *
624     * *
625     * *
626     * *
627     * *
628     * *
629     * *
630     * *
631     * *
632     * *
633     * *
634     * *
635     * *
636     * *
637     * *
638     * *
639     * *
640     * *
641     * *
642     * *
643     * *
644     * *
645     * *
646     * *
647     * *
648     * *
649     * *
650     * *
651     * *
652     * *
653     * *
654     * *
655     * *
656     * *
657     * *
658     * *
659     * *
660     * *
661     * *
662     * *
663     * *
664     * *
665     * *
666     * *
667     * *
668     * *
669     * *
670     * *
671     * *
672     * *
673     * *
674     * *
675     * *
676     * *
677     * *
678     * *
679     * *
680     * *
681     * *
682     * *
683     * *
684     * *
685     * *
686     * *
687     * *
688     * *
689     * *
690     * *
691     * *
692     * *
693     * *
694     * *
695     * *
696     * *
697     * *
698     * *
699     * *
700     * *
701     * *
702     * *
703     * *
704     * *
705     * *
706     * *
707     * *
708     * *
709     * *
710     * *
711     * *
712     * *
713     * *
714     * *
715     * *
716     * *
717     * *
718     * *
719     * *
720     * *
721     * *
722     * *
723     * *
724     * *
725     * *
726     * *
727     * *
728     * *
729     * *
730     * *
731     * *
732     * *
733     * *
734     * *
735     * *
736     * *
737     * *
738     * *
739     * *
740     * *
741     * *
742     * *
743     * *
744     * *
745     * *
746     * *
747     * *
748     * *
749     * *
750     * *
751     * *
752     * *
753     * *
754     * *
755     * *
756     * *
757     * *
758     * *
759     * *
760     * *
761     * *
762     * *
763     * *
764     * *
765     * *
766     * *
767     * *
768     * *
769     * *
770     * *
771     * *
772     * *
773     * *
774     * *
775     * *
776     * *
777     * *
778     * *
779     * *
780     * *
781     * *
782     * *
783     * *
784     * *
785     * *
786     * *
787     * *
788     * *
789     * *
790     * *
791     * *
792     * *
793     * *
794     * *
795     * *
796     * *
797     * *
798     * *
799     * *
800     * *
801     * *
802     * *
803     * *
804     * *
805     * *
806     * *
807     * *
808     * *
809     * *
810     * *
811     * *
812     * *
813     * *
814     * *
815     * *
816     * *
817     * *
818     * *
819     * *
820     * *
821     * *
822     * *
823     * *
824     * *
825     * *
826     * *
827     * *
828     * *
829     * *
830     * *
831     * *
832     * *
833     * *
834     * *
835     * *
836     * *
837     * *
838     * *
839     * *
840     * *
841     * *
842     * *
843     * *
844     * *
845     * *
846     * *
847     * *
848     * *
849     * *
850     * *
851     * *
852     * *
853     * *
854     * *
855     * *
856     * *
857     * *
858     * *
859     * *
860     * *
861     * *
862     * *
863     * *
864     * *
865     * *
866     * *
867     * *
868     * *
869     * *
870     * *
871     * *
872     * *
873     * *
874     * *
875     * *
876     * *
877     * *
878     * *
879     * *
880     * *
881     * *
882     * *
883     * *
884     * *
885     * *
886     * *
887     * *
888     * *
889     * *
890     * *
891     * *
892     * *
893     * *
894     * *
895     * *
896     * *
897     * *
898     * *
899     * *
900     * *
901     * *
902     * *
903     * *
904     * *
905     * *
906     * *
907     * *
908     * *
909     * *
910     * *
911     * *
912     * *
913     * *
914     * *
915     * *
916     * *
917     * *
918     * *
919     * *
920     * *
921     * *
922     * *
923     * *
924     * *
925     * *
926     * *
927     * *
928     * *
929     * *
930     * *
931     * *
932     * *
933     * *
934     * *
935     * *
936     * *
937     * *
938     * *
939     * *
940     * *
941     * *
942     * *
943     * *
944     * *
945     * *
946     * *
947     * *
948     * *
949     * *
950     * *
951     * *
952     * *
953     * *
954     * *
955     * *
956     * *
957     * *
958     * *
959     * *
960     * *
961     * *
962     * *
963     * *
964     * *
965     * *
966     * *
967     * *
968     * *
969     * *
970     * *
971     * *
972     * *
973     * *
974     * *
975     * *
976     * *
977     * *
978     * *
979     * *
980     * *
981     * *
982     * *
983     * *
984     * *
985     * *
986     * *
987     * *
988     * *
989     * *
990     * *
991     * *
992     * *
993     * *
994     * *
995     * *
996     * *
997     * *
998     * *
999     * *
1000    * *

```

```

DO 65 J=1,NIS
PER(I,J)=PER(I,J)*XX
IF(NREF.EQ.NREF)GO TO 160
FRINT 161,NREF,NREF
FORMAT(6 REF SAMPLES FOUND*,I3,5X,6 SPECIFIED*,I3)
CALL REFID(NREF,KREF,RCF,REF,NIS)
GO TO 13
DO 140 I=1,NIS
SUM=0.
DO 141 J=1,NIS
SUM=SUM+PER(I,J)
SUM=1./SUM
DO 142 J=1,NIS
PER(I,J)=PER(I,J)*SUM
CONTINUE
CALL INV(PER,NIS,0)
COMPUTE CORRECTED CPHS FOR REFERENCE SAMPLES
DO 143 I=1,NIS
IGT(I)=RCT(I)
CALL OPM(IGT,RCF,PER,NIS,0)
PRINT 61,(IGT(I),I=1,NIS)
FORMAT(/% REF CPM *,5I10)
FORMAT(/% CORR CPM *,5F10.0)
CO 411 I=1,NIS
TOT(I)=TOT(I)+TVOL(I)/CVOL(I)
PRINT 506,TOT
FORMAT(/% SYRINGE*,5G12.5/)
PROCESS ASSAY SAMPLES AND RESIDUE SAMPLES
REWIND 1
REWIND 2
DO 22 J=1,200
MGP(J)=0
DO 22 I=1,NIS
SSCP(I,J)=0.
LET=1NS
READ(I,1)IIO,((IM(I),IGT(I)),I=1,NIS)
IF(IOP(I))100,101
CHECK IF DELAY CORRECTION
CALL HALF(IIO,IGT,NDEL,NSDEL,DAY,0,NIS)
CHECK IF A STANDARD
DO 23 I=1,IT
IF(IIO.EQ.KS(I))160 TO 25
CONTINUE
CHECK IF A REFERENCE SAMPLE
IF(NREF.LE.0)160 TO 170
DO 171 I=1,NREF
IF(IIO.EQ.KREF(I))GO TO 25
CONTINUE
CALL WINDOW(IIO,IM,IGT,LIM,NIS,BACK)
CALL OPM(IGT,XCI,PER,NIS,0)
CHECK IF FROM RESIDUE SAMPLE
IF(NCATH.LE.0)GO TO 34
DO 30 I=1,NCATH
IF(IIO.NE.KC(I))160 TO 30

```

```

230      DO 31 J=1,NIS
          TOT(J)=TOT(J)-XCT(J)
          GO TO 25
231      CONTINUE
          CHECK IF IN A GROUP
          IF(IE.LE.0)GO TO 35
232      DO 33 I=1,IE
          IF(IID.NE.KGP(I,1))GO TO 33
          IGP=KGP(I,2)
          PRINT 230,IID,IGP
          FOFMAT(I5,*,IN GROUP*,I3)
233      DO 36 J=1,NIS
          SSGP(IGP,J)=SSGP(IGP,J)+ICT(J)
          MGP(IGP)=MGP(IGP)+1
          GO TO 25
234      CONTINUE
          PRINT 40,IID
          FORMAT(* SAMPLE*,IS,* -- ENTER SAMPLE ID AND WEIGHT (G) *)
          READ *,KID,WT
          DO 42 I=1,NIS
          FLOW(I)=0.
          FLOW(I)=XCT(I)*WTHD/(KCT(I)*WT)
235      CONTINUE
          WRITE (2,44)LET,IID,KID,WT,(XCT(I),I=1,NIS)
          FORMAT(A1,2I5,6D12.5)
          WRITE (2,401)(FLOW(I),I=1,NIS)
          FORMAT(23X,5D12.5)
          GO TO 25
236      LET=ING
          IFL=0
          IB=1
          IF(IE.LE.0)GO TO 333
237      DO 200 I=1,NGFS
          IF(MGP(I).EQ.0)GO TO 203
          IFL=1
          PRINT 201,I,MGP(I),NGP(I)
          FORMAT(* GROUP*,I3,* # SAMPLES FOUND*,I3,* # SPECIFIED*,I3)
          IF(IFL.EQ.0)GO TO 204
          PRINT 11,1
          READ *,NGP(I)
          CALL NUMB(NGP(I),KOUN)
          IE=IB-1+NGP(I)
          K=0
          DO 202 J=IB,IE
          K=K+1
          KGP(J,1)=KOUN(K)
          KGP(J,2)=I
          IB=IE+1
          GO TO 200
238      IB=IB+NGP(I)
          CONTINUE
          IF(IFL.NE.0)GO TO 100
          DO 198 I=1,NGFS
          DO 191 J=1,NIS
          ICT(J)=SSGP(I,J)

```


VARIABLES	SN	TYPE	MODE	FILE NAMES
13001 NGP		INTEGER	MIXED	
12276 NIS		INTEGER		
12306 NREF		INTEGER		
12311 NSS		INTEGER		
12717 PER		REAL		
12734 PCT		REAL		
12716 SSGP		REAL		
12323 SUN		REAL		
12344 TVOL		REAL		
12330 HT		REAL		
12324 XX		REAL		

EXTERNALS	CPM	EXIT	INV	REFID
0 INPUT	5	0	3	5

STATEMENT LABELS	SN	TYPE	MODE	FILE NAMES
11447 1		FMT		
11523 4		FMT		
11553 7		FMT		
10501 13		FMT		
10441 15		FMT		
0 23		FMT		
0 31		FMT		
11137 35		FMT		
11152 42		FMT		
10435 51		FMT		
12034 61		FMT		
0 65		FMT		
11544 107		FMT		
11626 120		FMT		
0 132		FMT		
10522 135		FMT		
0 140		FMT		
0 143		FMT		
10716 160		FMT		
0 171		FMT		
0 191		FMT		
11244 200		FMT		
11205 203		FMT		
12074 230		FMT		
12242 242		FMT		
0 301		FMT		
11461 400		FMT		
11562 403		FMT		
0 411		FMT		

EXTERNALS	CPM	EXIT	INV	REFID
0 INPUT	5	0	3	5

STATEMENT LABELS	SN	TYPE	MODE	FILE NAMES
11507 2		FMT		
11473 5		FMT		
11633 10		FMT		
11741 14		FMT		
11757 17		FMT		
11025 25		FMT		
11134 33		FMT		
0 36		FMT		
12127 44		FMT		
0 52		FMT		
10304 52		FMT		
11167 100		FMT		
11672 108		FMT		
0 130		FMT		
10572 133		FMT		
10616 136		FMT		
0 141		FMT		
10660 150		FMT		
12016 161		FMT		
11093 180		FMT		
12175 192		FMT		
12147 201		FMT		
11242 204		FMT		
12231 240		FMT		
12265 243		FMT		
12062 302		FMT		
12136 401		FMT		
11575 404		FMT		
12050 500		FMT		

STATEMENT LABELS	SN	TYPE	MODE	FILE NAMES
11534 3		FMT		
11636 6		FMT		
11725 11		FMT		
0 15		FMT		
0 22		FMT		
11104 30		FMT		
11107 34		FMT		
12103 40		FMT		
11703 50		FMT		
10412 53		FMT		
0 63		FMT		
0 101		FMT		
10407 110		FMT		
10621 131		FMT		
0 134		FMT		
0 137		FMT		
0 142		FMT		
0 151		FMT		
11063 170		FMT		
0 190		FMT		
11277 193		FMT		
0 202		FMT		
11331 220		FMT		
11332 241		FMT		
11766 300		FMT		
11316 333		FMT		
12040 402		FMT		
11610 405		FMT		

STATEMENT LABELS	SN	TYPE	MODE	FILE NAMES
12315 NGPS		INTEGER		
12304 NKS		INTEGER		
14220 NSDEL		INTEGER		
14213 NLOT		INTEGER		
12351 Q		REAL		
12402 REF		REAL		
12308 SUBJ		REAL		
15711 TOT		REAL		
12307 WITHD		REAL		
15677 XCT		REAL		

STATEMENT LABELS	SN	TYPE	MODE	FILE NAMES
11507 2		FMT		
11473 5		FMT		
11633 10		FMT		
11741 14		FMT		
11757 17		FMT		
11025 25		FMT		
11134 33		FMT		
0 36		FMT		
12127 44		FMT		
0 52		FMT		
10304 52		FMT		
11167 100		FMT		
11672 108		FMT		
0 130		FMT		
10572 133		FMT		
10616 136		FMT		
0 141		FMT		
10660 150		FMT		
12016 161		FMT		
11093 180		FMT		
12175 192		FMT		
12147 201		FMT		
11242 204		FMT		
12231 240		FMT		
12265 243		FMT		
12062 302		FMT		
12136 401		FMT		
11575 404		FMT		
12050 500		FMT		

STATEMENT LABELS	SN	TYPE	MODE	FILE NAMES
10317 62		FMT		
10357 63		FMT		
10427 52		FMT		
10470 15		FMT		

LOOPS LABEL	INDEX	FROM-TO	LENGTH	PROPERTIES
10505	130	* I	158	NOT INNER
10514	130	J	113 117	INSTACK
10525	* I	116 117	28	EXT REFS
10542	* I	118 118	108	EXT REFS
10564	* J	122 136	338	EXT REFS
10577	* I	129 134	58	EXT REFS
10611	* J	139 147	229	EXT REFS
10622	* I	144 145	38	EXT REFS
10630	* J	149 164	418	EXT REFS
10640	* J	151 152	38	EXT REFS
10654	* J	157 161	168	EXT REFS
10701	* J	168 173	248	EXT REFS
10717	* J	172 173	38	EXT REFS
10725	* I	179 186	308	EXT REFS
10740	* J	181 182	38	EXT REFS
10753	* J	184 185	38	EXT REFS
10775	* I	189 190	38	EXT REFS
11010	* J	196 197	38	EXT REFS
11016	* I	205 208	148	EXT REFS
11030	* I	207 208	28	EXT REFS
11046	* I	210 210	108	EXT REFS
11056	* I	215 217	58	EXT REFS
11072	* I	220 222	58	EXT REFS
11100	* I	227 232	158	EXT REFS
11112	* J	229 230	38	EXT REFS
11125	* I	235 244	258	EXT REFS
11146	* J	240 241	48	EXT REFS
11174	* I	246 252	58	EXT REFS
11232	* J	262 261	538	EXT REFS
11251	* J	274 277	48	EXT REFS
11256	* I	283 297	458	EXT REFS
11273	* J	284 285	38	EXT REFS
11326	* I	290 294	58	EXT REFS
11326	* I	301 306	58	EXT REFS

STATISTICS
 PROGRAM LENGTH 74748 3900
 BUFFER LENGTH 102608 4272
 520008 CH USED

```

1  SUBROUTINE CPH(ICT,XCT,PER,NIS,Q)
   DIMENSION ICT(NIS),XCT(NIS),PER(NIS,NIS)
   DIMENSION Q(NIS,NIS)
   PER IS INV(A *A TRANS) * A
   * WHERE A(I,J) IS THE PROPORTION OF COUNTS FROM ISOTOPE
   * I IN KINDOM J
   * SUM OF A(I,J) OVER J FOR FIXED I IS 1.0
10  DO 5 I=1,NIS
   XCT(I)=0.
   DO 4 J=1,NIS
   XCT(I)=XCT(I)+PER(I,J)*ICT(J)
5  CONTINUE
   CALL ICLS(Q,XCT,NIS)
   RETURN
15  ENC

```

SYMBOLIC REFERENCE MAP (R=1)

ENTRY POINTS
3 CPH

22 VARIABLES SM TYPE
54 I INTEGER
55 J INTEGER
0 PER REAL
0 XCT REAL

EXTERNALS ICLS TYPE ARGS
3

STATEMENT LABELS
0 4

LOOPS LABEL * I INDEX FROM-TO LENGTH PROPERTIES
20 5 I 0 12 178 NOT INNER
30 4 J 10 11 48 INSTACK

STATISTICS PROGRAM LENGTH 728 58
52000 CM USED

RELOCATION 0 ICT
0 NIS
0 0
F.P.
F.P.
F.P.

INTEGER ARRAY
INTEGER REAL
REAL ARRAY
F.P.
F.P.

0 5

```

1 SUBROUTINE NUMB(NLIM,KC)
  DIMENSION KC(200)
  I=0
  15 READ *,KK
  IF(KK.GT.0)GO TO 11
  IF(I.LE.0)GO TO 12
  IB=KC(I)+1
  IE=IABS(KK)
  DO 13 J=IB,IE
    I=I+1
  13 KC(I)=J
  GO TO 14
  12 KK=IABS(KK)
  I=I+1
  15 KC(I)=KK
  IF(I-NLIM)15,20,21
  20 RETURN
  21 PRINT 22,I
  22 FORMAT(' SPECIFIED # IS ',I5)
  RETURN
  END
  
```

23

SYMBOLIC REFERENCE MAP (R=1)

ENTRY POINTS
3 NUMB

VARIABLES	SN	TYPE	RELOCATION
61 I		INTEGER	63 2B
64 IE		INTEGER	65 J
0 KC		INTEGER	62 KK
8 NLIM		INTEGER	F.P.
		ARRAY	F.P.

FILE NAMES MODE INPUT OUTPUT FMT

INLINE FUNCTIONS TYPE ARGS INTRIN

STATEMENT LABELS

32 11	30 12	0 13
36 14	6 15	0 20
42 21	55 22	

LOOPS LABEL INDEX FROM-TO LENGTH PROPERTIES

24 13	J	9 11	28	INSTACK
-------	---	------	----	---------

STATISTICS

PROGRAM LENGTH 52008 CH USED 668 54

```

1 SUBROUTINE REFTD(NREF,KREF,RCF,REF,NIS)
  DIMENSION KREF(200),RCF(NIS),REF(NIS)
  DO 1 I=1,NIS
    RCF(I)=0.
  1 READ *,NREF
  PRINT 121
  121 FORMAT(* ENTER SAMPLE IDS FOR REFERENCE *)
  CALL NUMB(NREF,KREF)
  RETURN
  END
  
```

SYMBOLIC REFERENCE MAP (R=1)

ENTRY POINTS
3 REFID

VARIABLES	SN	TYPE	RELOCATION	0	KREF	INTEGER	ARRAY	F.P.
43 I		INTEGER		0	KREF	INTEGER	ARRAY	F.P.
8 NIS		INTEGER		0	NREF	INTEGER	ARRAY	F.P.
8 RCF		REAL		0	REF	REAL	ARRAY	F.P.

FILE NAMES
INPUT FREE

EXTERNALS
NUMB TYPE ARGS

STATEMENT LABELS
0 1

LOOPS LABEL	INDEX	FROM-TO	LENGTH	PROPERTIES
11 1	1	3 4	28	INSTACK

STATISTICS
PROGRAM LENGTH 448 36
PROGRAM LENGTH 52808 CH USED

```

1 SUBROUTINE WINDOW (IID,IM,ICT,LIMM,NIS,BACK)
  DIMENSION BACK(NIS)
  DIMENSION IH(NIS),ICT(NIS),LIMM(NIS)
  DO 18 I=1,NIS
  IF (ABS(IH(I))-LIMM(I)).LE.3)GO TO 18
  PRINT 19,IID,IM(I),LIMM(I),ICT(I)
  19 FORMAT(* WINDOW WRONG*)
  1 5X,*SAMPLE*,I3,* WINDOW*,I2,* CENTROID*,I5/
  2 5X,*SHOULD BE*,I5,* COUNT*,I8)
  PRINT 20
  20 FORMAT(* ENTER CORRECT CENTROID AND COUNT *)
  READ *,IH(I),ICT(I)
  GO TO 21
  18 CONTINUE
  CORRECT FOR BACKGROUND COUNTS
  DO 1 I=1,NIS
  IF (ICT(I).LE.0)GO TO 1
  XX=BACK(I)+BACK(I)/ICT(I)
  XX=ICT(I)-XX
  IF (XX.LE.0.)XX=0.
  ICT(I)=XX+.5
  1 CONTINUE
  RETURN
  END
  
```

SYMBOLIC REFERENCE MAP (R=1)

ENTRY POINTS	SYMBOLIC REFERENCE MAP (R=1)	SYMBOLIC REFERENCE MAP (R=1)	SYMBOLIC REFERENCE MAP (R=1)	SYMBOLIC REFERENCE MAP (R=1)	SYMBOLIC REFERENCE MAP (R=1)	SYMBOLIC REFERENCE MAP (R=1)	SYMBOLIC REFERENCE MAP (R=1)	SYMBOLIC REFERENCE MAP (R=1)	SYMBOLIC REFERENCE MAP (R=1)
3 WINDOW									
VARIABLES	SN	TYPE	RELOCATION						
0 BACK	REAL	ARRAY	F.P.	126	I	INTEGER			
0 ICT	INTEGER	ARRAY	F.P.	0	IID	INTEGER			
0 IM	INTEGER	ARRAY	F.P.	0	LIMM	INTEGER			
0 NIS	INTEGER	ARRAY	F.P.	127	XX	REAL			
FILE NAMES	MODE	OUTPUT	FREE						
INPUT	FREE	OUTPUT	FREE						
INLINE FUNCTIONS	TYPE	ARGS							
IABS	INTEGER	1	INTRIN						
STATEMENT LABELS									
61	1			41	18				
113	20	FMT		16	21				
LOOPS	LABEL	INDEX	FROM-TO	LENGTH	PROPERTIES	EXT	REFS		
16	18	1	4	14	268				
51	1	1	16	22	118	OPT			
STATISTICS	PROGRAM	LENGTH	52008	CH	USED	1508			184

```

1 SUBROUTINE HALF(IID,ICT,NGEL,NSDEL,DAY,IS,NIS)
  DIMENSION ICI(NIS),DAY(200),NSDEL(200),HL(5)
  DATA HL/60.,32.,28.,64.,84.,/
  DATA ML/60.,28.,64.,84.,0.,/
  IF(NGEL.LE.0)RETURN
  C=ALOG(1.50)
  DO 1 I=1,NSDEL
    IF(IIO.NE.NSDEL(I))GO TO 1
    DO 2 J=1,NIS
      HH=HL(J)
      IF(IS.GT.0)HH=HL(IS)
      ICT(J)=ICT(J)*EXP(C*DAY(I)/HH)+.5
    * PRINT 3,IID,ICT
    * FO:MAT(* SAMPLE*,IS,* -- DECAY CORRECTED COUNTS*,517)
    * RETURN
  1 CONTINUE
  RETURN
  END
  15
  
```

SYMBOLIC REFERENCE MAP (R=1)

26 ENTRY POINTS
3 HALF

VARIABLES	SN	TYPE	RELOCATION	0	DAY	REAL	ARRAY	F.P.
50 C		REAL		54	ML	REAL	ARRAY	F.P.
53 HH		REAL			ICT	INTEGER	ARRAY	F.P.
51 I		INTEGER			IS	INTEGER		F.P.
6 IID		INTEGER			NOEL	INTEGER		F.P.
52 J		INTEGER			NSDEL	INTEGER	ARRAY	F.P.
0 MIS		INTEGER						F.P.

EXTERNALS	ALOG	TYPE	ARGS	REAL	EXP	LIBRARY
1		REAL	1	REAL		1 LIBRARY

STATEMENT LABELS	FROM-TO	LENGTH	PROPERTIES
44 1			0 2

LOOPS LABEL	INDEX	FROM-TO	LENGTH	EXT REFS	NOT INNER
22 1	* I	7 16	258	EXT REFS	
26 2	* J	9 12	168	EXT REFS	

STATISTICS
PROGRAM LENGTH 778 63
52000 CH USED

```

1 SUBROUTINE INV (PER, NIS, Q)
  DIMENSION PER(NIS, NIS), Q(NIS, NIS)
  DIMENSION A(5, 5), M(10), B(5, 5)

5 * COMPUTE INVERSE (PER * PER TRANS) * PER
  *
  * A = PER * PER TRANS
  DO 1 I=1, NIS
  DO 1 J=1, NIS
  A(I, J) = 0.
  DO 1 K=1, NIS
  A(I, J) = A(I, J) + PER(I, K) * PER(J, K)
  INVERSE A
  CALL LINV3F(A, B, 1, NIS, NIS, D1, D2, M, IER)
  DO 4 I=1, NIS
  DO 4 J=1, NIS
  Q(I, J) = A(I, J)
  DO 2 I=1, NIS
  DO 2 J=1, NIS
  B(I, J) = 0.
  DO 2 K=1, NIS
  B(I, J) = B(I, J) + A(I, K) * PER(K, J)
  DO 3 I=1, NIS
  DO 3 J=1, NIS
  PER(I, J) = B(I, J)
  RETURN
  END
  
```

27

SYMBOLIC REFERENCE MAP (R=1)

ENTRY POINTS
3 INV

VARIABLES	SN	TYPE	RELOCATION	224	B	REAL	ARRAY	157	D2	REAL	160	IER	INTEGER	155	K	PER	F.P.	212	M	REAL	ARRAY	F.P.	REAL	ARRAY
161 A		REAL	ARRAY																					
156 D1		REAL																						
153 I		INTEGER																						
154 J		INTEGER																						
Q NIS		INTEGER																						
B Q		REAL																						

EXTERNALS 9
LINV3F 9

STATEMENT LABELS

0	1	2	3
0	0	0	0
1	1	2	3
2	3	4	5

LOOPS	LABEL	INDEX	FROM-TO	LENGTH	PROPERTIES
20	1	I	6 12	258	NOT INNER
21	1	J	9 12	228	NOT INNER
34	1	K	11 12	38	INSTACK
52	4	I	15 17	158	NOT INNER

LOOPS	LABEL	INDEX	FROM-TO	LENGTH	PROPERTIES
61	4	J	16 17	38	INSTACK
70	2	* I	18 22	308	NOT INNER
71	2	* J	19 22	248	NOT INNER
106	2	K	21 22	48	INSTACK
121	3	* I	23 25	158	NOT INNER
130	3	J	24 25	38	INSTACK

STATISTICS

PROGRAM LENGTH 2768 190

52008 CH USED

```

SUBROUTINE ICLS(Q,BETA,N)
DIMENSION Q(N,N),BETA(N),Y(5),Z(5),B(5),M(5),ZMD(5)
DO 2 I=1,N
DO 2 J=1,N
B(I,J)=0.
DO 1 I=1,N
Y(I)=0.
B(I,1)=1.
Z(I)=M(I)-BETA(I)
IND(I)=1
DO 4 I=1,N
DO 4 J=1,N
BINV(I,J)=B(I,J)
ICSS=LEI=0
DO 30 I=1,N
M(I)=0.
DO 5 J=1,N
M(I)=M(I)+BINV(I,J)*Z(J)
IF(ABS(M(I)).LE.1.0E-8)M(I)=0.0
CONTINUE
IT=0
XMAX=0.
DO 6 I=1,N
IF(ZMD(I).GT.0)GO TO 6
IF(M(I).EQ.0.0)GO TO 6
ICSS=1
IF(XMAX.GE.ABS(M(I)))GO TO 6
XMAX=ABS(M(I))
IT=I
CONTINUE
IF(ICSS.GT.0)GO TO 7
DO 8 I=1,N
IF(M(I).GE.0.0)GO TO 8
ICSS=2
IF(ABS(M(I)).LE.XMAX)GO TO 8
XMAX=ABS(M(I))
IT=I
CONTINUE
IF(ICSS.LE.0)GO TO 100
DO 50 I=1,N
SQ(I)=0.
DO 9 J=1,N
SQ(I)=SQ(I)+Q(I,J)*BINV(I,J)
IF(ABS(SQ(I)).LE.1.0E-8)SQ(I)=0.
CONTINUE
DO 10 I=1,N
IF(SQ(I).NE.0.0)GO TO 11
CONTINUE
THET0=SIGN(1.0,M(IT))
GO TO 17
THET0=0.
DO 13 I=1,N
THET6=THET0+SQ(I)*BINV(I,I)
IF(ABS(THET0).LE.1.0E-8)GO TO 61
THET0=M(IT)/THET6
ITT=0

```

1

5

10

15

20

25

30

35

40

45

50

55

17

```

100  T0=1.0E20
101  DO 14 I=1,N
102  GEN=THET0*BINV(I,I)
103  IF(CEN.LE.1.0E-8)GO TO 14
104  IF(Y(I).EQ.0.0)GO TO 14
105  DEN=Y(I)/DEN
106  IF(DEN.GE.T0)GO TO 14
107  ITT=I
108  T0=DEN
109  CONTINUE
110  IF(Y0.LE.1.0)GO TO 15
111  THET=THET0
112  ILET=1
113  DO 21 I=1,N
114  B(I,IT)=SQ(I)
115  GO TO 22
116  THET=THET0*T0
117  ILET=2
118  DO 23 I=1,N
119  B(I,IT)=0.
120  9(I,IT)=1.0
121  DO 25 I=1,N
122  Y(I)=Y(I)-THET*BINV(IT,I)
123  Z(I)=Z(I)-THET*SQ(I)
124  IF(ABS(Z(I)).LE.1.0E-8)Z(I)=0.0
125  IF(ABS(Y(I)).LE.1.0E-8)Y(I)=0.0
126  CONTINUE
127  DO 24 I=1,N
128  CO 24 J=1,N
129  BINV(I,J)=B(I,J)
130  CALL LINV3F(BINV,C,1,N,M,D1,D2,MK,IER)
131  IF(ICS.EQ.1.AND.ILET.EQ.2)IND(I)=1
132  IF(ICS.EQ.2.AND.ILET.EQ.1)IND(I)=0
133  GO TO 3
134  DO 101 I=1,N
135  DO 200 J=1,N
136  BETA(I)=BETA(I)+Q(I,J)*Y(J)
137  IF(ABS(BETA(I)).LE.1.0E-8)BETA(I)=0.
138  CONTINUE
139  RETURN
140  END

```

SYMBOLIC REFERENCE MAP (R=1)

ENTRY POINTS	VARIABLES	SM	TYPE	RELOCATION	BETA	REAL	ARRAY	F.P.
3	457	8	REAL	ARRAY	0	REAL	ARRAY	
	522	BINV	REAL	ARRAY	565	REAL	ARRAY	
	440	DEN	REAL		442	REAL		
	443	D2	REAL		427	INTEGER		
	431	ICS	INTEGER		444	INTEGER		

VARIABLES	SN	TYPE	RELOCATION	515	IND	INTEGER	ARRAY
432 ILET		INTEGER		436	ITT	INTEGER	
433 IT		INTEGER		0	H	INTEGER	
439 J		REAL		572	SO	REAL	
441 THET		REAL	F.P.	435	THETO	REAL	
437 T0		REAL		510	H	REAL	
553 HK		REAL		434	XMAX	REAL	
445 Y		REAL		452	Z	REAL	

EXTERNALS LIN3F TYPE ARGS
 9

INLINE FUNCTIONS	REAL	TYPE	ARGS	INTRIN	SIGN	REAL	2	INTRIN
ABS		REAL	1					

STATEMENT LABELS

LINE	ABS	REAL	INTRIN	SIGN	REAL	2	INTRIN
0 1							
0 2							
0 3							
0 4							
0 5							
146 7							
207 11							
265 15							
304 22							
0 24							
0 25							
0 60							
0 61							
0 101							
0 200							

31

LOOPS	LABEL	INDEX	FROM-TO	LENGTH	PROPERTIES
28 2		I	4 6	138	
25 2		J	5 6	28	NOT INNER
62 1		I	7 11	68	INSTACK
51 4		I	12 14	148	INSTACK
56 4		J	13 14	38	NOT INNER
78 30		I	16 21	238	INSTACK
77 5		J	18 19	48	NOT INNER
128 6		I	24 31	109	INSTACK
136 8		I	33 39	78	INSTACK
151 88		I	41 46	258	NOT INNER
162 9		J	43 44	48	INSTACK
177 18		I	47 49	48	INSTACK
215 13		I	53 54	48	INSTACK
235 14		I	59 67	128	INSTACK
261 21		I	71 72	28	OPT
276 23		I	76 77	28	INSTACK
313 25		I	79 84	148	INSTACK
331 24		I	85 87	148	OPT
336 24		J	86 87	38	NOT INNER
366 101		I	92 96	228	INSTACK
374 288		J	93 94	48	NOT INNER

STATISTICS PROGRAM LENGTH 52008 OH USED 6238 403

***** ZA4KGE4 *****
 ***** ZA4KGE4 *****
 ***** END OF LIST *****
 ***** END OF LIST *****

**APPENDIX A
OUTPUT FROM GAMMA COUNTER**

Channel	Count	Rate	Energy	Source
1	10	0.1	100	Na-22
2	20	0.2	200	Na-22
3	30	0.3	300	Na-22
4	40	0.4	400	Na-22
5	50	0.5	500	Na-22
6	60	0.6	600	Na-22
7	70	0.7	700	Na-22
8	80	0.8	800	Na-22
9	90	0.9	900	Na-22
10	100	1.0	1000	Na-22
11	110	1.1	1100	Na-22
12	120	1.2	1200	Na-22
13	130	1.3	1300	Na-22
14	140	1.4	1400	Na-22
15	150	1.5	1500	Na-22
16	160	1.6	1600	Na-22
17	170	1.7	1700	Na-22
18	180	1.8	1800	Na-22
19	190	1.9	1900	Na-22
20	200	2.0	2000	Na-22
21	210	2.1	2100	Na-22
22	220	2.2	2200	Na-22
23	230	2.3	2300	Na-22
24	240	2.4	2400	Na-22
25	250	2.5	2500	Na-22
26	260	2.6	2600	Na-22
27	270	2.7	2700	Na-22
28	280	2.8	2800	Na-22
29	290	2.9	2900	Na-22
30	300	3.0	3000	Na-22
31	310	3.1	3100	Na-22
32	320	3.2	3200	Na-22
33	330	3.3	3300	Na-22
34	340	3.4	3400	Na-22
35	350	3.5	3500	Na-22
36	360	3.6	3600	Na-22
37	370	3.7	3700	Na-22
38	380	3.8	3800	Na-22
39	390	3.9	3900	Na-22
40	400	4.0	4000	Na-22
41	410	4.1	4100	Na-22
42	420	4.2	4200	Na-22
43	430	4.3	4300	Na-22
44	440	4.4	4400	Na-22
45	450	4.5	4500	Na-22
46	460	4.6	4600	Na-22
47	470	4.7	4700	Na-22
48	480	4.8	4800	Na-22
49	490	4.9	4900	Na-22
50	500	5.0	5000	Na-22
51	510	5.1	5100	Na-22
52	520	5.2	5200	Na-22
53	530	5.3	5300	Na-22
54	540	5.4	5400	Na-22
55	550	5.5	5500	Na-22
56	560	5.6	5600	Na-22
57	570	5.7	5700	Na-22
58	580	5.8	5800	Na-22
59	590	5.9	5900	Na-22
60	600	6.0	6000	Na-22
61	610	6.1	6100	Na-22
62	620	6.2	6200	Na-22
63	630	6.3	6300	Na-22
64	640	6.4	6400	Na-22
65	650	6.5	6500	Na-22
66	660	6.6	6600	Na-22
67	670	6.7	6700	Na-22
68	680	6.8	6800	Na-22
69	690	6.9	6900	Na-22
70	700	7.0	7000	Na-22
71	710	7.1	7100	Na-22
72	720	7.2	7200	Na-22
73	730	7.3	7300	Na-22
74	740	7.4	7400	Na-22
75	750	7.5	7500	Na-22
76	760	7.6	7600	Na-22
77	770	7.7	7700	Na-22
78	780	7.8	7800	Na-22
79	790	7.9	7900	Na-22
80	800	8.0	8000	Na-22
81	810	8.1	8100	Na-22
82	820	8.2	8200	Na-22
83	830	8.3	8300	Na-22
84	840	8.4	8400	Na-22
85	850	8.5	8500	Na-22
86	860	8.6	8600	Na-22
87	870	8.7	8700	Na-22
88	880	8.8	8800	Na-22
89	890	8.9	8900	Na-22
90	900	9.0	9000	Na-22
91	910	9.1	9100	Na-22
92	920	9.2	9200	Na-22
93	930	9.3	9300	Na-22
94	940	9.4	9400	Na-22
95	950	9.5	9500	Na-22
96	960	9.6	9600	Na-22
97	970	9.7	9700	Na-22
98	980	9.8	9800	Na-22
99	990	9.9	9900	Na-22
100	1000	10.0	10000	Na-22

```

000274 00001.0
000016 000160 000032 001662 000048 000108 000000 013791
000128 000086 000144 000221 000160 000212 000000 003312
000304 000096 000320 000216 000336 000243 000000 003388
000496 000048 000514 000050 000530 000048 000000 000790
001104 000041 001120 000045 001136 000050 000000 000750

```

	Counter ID	Counting time (min)	Centroid	XXXXXX	XXXXXX	XXXXXX	XXXXXX	Total CPM
Iodine	XXXXXX	XXXXXX	Centroid	XXXXXX	XXXXXX	XXXXXX	XXXXXX	Total CPM
Cerium	XXXXXX	XXXXXX	Centroid	XXXXXX	XXXXXX	XXXXXX	XXXXXX	Total CPM
Chromium	XXXXXX	XXXXXX	Centroid	XXXXXX	XXXXXX	XXXXXX	XXXXXX	Total CPM
Strontium	XXXXXX	XXXXXX	Centroid	XXXXXX	XXXXXX	XXXXXX	XXXXXX	Total CPM
Scandium	XXXXXX	XXXXXX	Centroid	XXXXXX	XXXXXX	XXXXXX	XXXXXX	Total CPM

The centroid is used by the computer to identify the specific isotope with which it is associated.

Total CPM refers to all counts from a specific isotope in the region of interest, accumulated during the counting time--in this case one minute.

CORRECTED MICROSPHERE COUNTS
 NUMBER OF ISOTOPES ASSUMED TO BE 5
 ISOTOPES I125 CE145 CR141 SR85 SC46
 ENTER SUBJECT I) F-30
 ENTER EXPERIMENT DATE 27 JUNE 78
 ENTER ASSAY DATE 30 JUNE 78
 ISOTOPE 1 I125
 ENTER VOLUME COUNTED .003
 ENTER VOLUME INJECTED .11
 ENTER # SAMPLE IDS 2
 ENTER SAMPLE IDS 1 2
 ISOTOPE 2 CE145
 ENTER VOLUME COUNTED .003
 ENTER VOLUME INJECTED .10
 ENTER # SAMPLE IDS 2
 ENTER SAMPLE IDS 3 4
 ISOTOPE 3 CR141
 ENTER VOLUME COUNTED .003
 ENTER VOLUME INJECTED .28
 ENTER # SAMPLE IDS 2
 ENTER SAMPLE IDS 5 6
 ISOTOPE 4 SR85
 ENTER VOLUME COUNTED .003
 ENTER VOLUME INJECTED .25
 ENTER # SAMPLE IDS 2
 ENTER SAMPLE IDS 7 8
 ISOTOPE 5 SC46
 ENTER VOLUME COUNTED .003
 ENTER VOLUME INJECTED .60
 ENTER # SAMPLE IDS 2
 ENTER SAMPLE IDS 9 10
 ENTER # REFERENCE SAMPLES 14
 ENTER SAMPLE IDS FOR REFERENCE 50 -63
 ENTER WITHDRAWAL RATE, REFERENCE SAMPLE 15.3
 ENTER # SAMPLES FROM RESIDUE 11
 ENTER SAMPLE IDS 11 -15 150 -155
 ENTER # SAMPLES COUNTED PREVIOUS TO ASSAY
 ENTER # SAMPLES AND # DAYS ELAPSED 77 3
 ENTER SAMPLE IDS 1 -10 11 -15 150 -155 50 -63 101 -139 141 -143
 ENTER # SAMPLES AND # DAYS ELAPSED 0 0
 ENTER SAMPLES IN GROUP 1 0
 ENTER SAMPLE IDS 271 -273
 ENTER # SAMPLES IN GROUP 2 0
 REF CPM 147707 194252
 CORR CPM 15794. 94106.
 SYPINGE .28140E+07 .35366E+07 .32361E+07 .41069E+07 .69244E+07

THIS PAGE IS BEST QUALITY PRACTICABLE
 FROM COPY FURNISHED TO DDC

APPENDIX C
SAMPLE TAPE 2

Counter Location		Io-125	Ce-141	Cr-51	Sr-85	Sc-46
Code ID	ID (grams)					
S 252	52 2.9500	3939.5	7080.8	3671.5	3705.4	15309.
		1.2937	.39024	.2373	.12607	.15958
S 253	53 6.2790	8699.7	18804.	10315.	9971.9	43458.-----Corrected CPM
		1.3422	.38332	.31326	.15940	.21283-----Blood flow(ml/min/g)
S 254	54 7.2590	7785.1	14660.	5652.8	7613.5	26195.
		1.0389	.25666	.18650	.10527	.11097
S 255	55 6.3860	7444.8	12441.	5662.0	5113.6	21875.
		1.0324	.28953	.15455	.73488E-01	.96290E-01
S 256	56 5.0220	6176.7	10083.	6252.5	6974.2	27582.
		1.1915	.32641	.23742	.13938	.16889
S 257	57 6.3780	9501.1	15414.	7733.1	6549.8	26132.
		1.3392	.36435	.21440	.95578E-01	.11684
S 258	58 1.0710	1723.2	2277.9	3175.2	3274.9	21311.
		1.5587	.34560	.56535	.30690	.61189
S 259	59 .93400	1289.3	2848.4	3558.0	5198.3	21392.
		1.3373	.49582	.72644	.55861	.70431
S 260	60 .50400	803.88	2074.8	2776.3	2988.1	15222.
		1.5451	.66930	1.0504	.99506	.92879
S 261	61 2.3790	1753.9	2830.6	1446.7	1138.9	5236.0
		.71418	.19345	.11596	.48049E-01	.67681E-01
S 262	62 2.1480	3026.0	5020.2	6280.4	6615.9	32979.
		1.3647	.44053	.55756	.30914	.47214
S 263	63 .66700	848.91	991.22	1181.5	1884.0	8789.4
		1.2329	.24161	.33779	.28350	.40522
S 264	64 2.6880	2560.2	4315.7	3289.3	3326.6	13043.
		.92267	.26103	.23193	.12421	.14922
S 265	65 7.2590	10252.	23342.	27377.	28105.	.12994E+06
		1.3682	.52294	.71918	.38860	.55045
S 266	66 14.698	17780.	29916.	19025.	18804.	62598.
		1.1725	.33109	.128696	.12868	.13184
G I 308	86.700	.10920E+06	.28742E+06	.14824E+06	.14835E+06	.50886E+06
		1.2202	.38896	.32605	.17174	.18049
F F-30	27 JUNE 7830	JUNE 78	1900.5	520.92	577.50	302.17
	27 JUNE 7830	JUNE 78	1900.5	520.92	577.50	302.17
Subject	Assay	Exp. date	Io-125	Ce-141	Cr-51	Sr-85
						Sc-46

Code: S= individual assay
G= group assay
F= cardiac output data

REFERENCES

1. Domenach, R.J., J.I.E. Hoffman, M.I.M. Noble, K.B. Saunders, J.R. Henson, and S. Subijanto, "Total and regional coronary blood flow measured by radioactive microspheres in conscious and anesthetized dogs," Circulation Research 25:581-596, 1969.
2. Greenlees, K. J., C.M. Oloff, W.J. Buehring, and K.C. Smith, "A technique for the injection of radioactive tracer microspheres during acceleration stress," (in press), Aviation, Space, and Environmental Medicine. (AMRL-TR-78-114)
3. Heymann, M.A., B.D. Payne, J. Hoffman, and A.M. Rudolph, "Blood flow measurement with radio-nuclide labelled particles," 1978 (in press).
4. Lederer, C.M., J.M. Hollander, and I. Perlman, Table of Isotopes, 6th ed., John Wiley and Sons Inc., New York, 1967.
5. Oloff, C.M., and W.L. Finch, Subhuman Primate Restraint System, AMRL-TR-78-88, Aerospace Medical Research Laboratory, Wright-Patterson AFB, Ohio, December 1978.
6. Rudolph, A.M., and M.A. Heymann, "The circulation of the fetus in utero: Methods for studying distribution of blood flow, cardiac output, and organ blood flow." Circulation Research 21:163-184, 1967.
7. Sostre, S., J.A. Kennealy, J.S. Kirkland, C.M. Oloff, A.A. Karl, and M.A. Franey, "Cerebral blood flow in baboons under positive acceleration." Aerospace Medical Assn., Annual Sci. Mtgs., 75-76, 1977.