MCR-68-97 (Vol II) Copy No. 2

JPL Contract 952028

A Study Program on the Development of a Mathematical Model(s) for Microbial Burden Prediction

Final Report

Volume II

User's Manual for the Microbial Burden Prediction Model

DECEMBER 1968

MARTIN MARIETTA CORPORATION

MCR-68-97

A STUDY PROGRAM ON THE DEVELOPMENT OF MATHEMATICAL MODEL(S) FOR MICROBIAL BURDEN PREDICTION

JPL Contract 952028

Final Report

Volume II

User's Manual
for the
Microbial Burden Prediction Program

By

P. B. McKowen

Prepared for:

Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, California 91103

APPROVED:

J. C. Curlander
Manager
Systems Engineering

G. E. Fosdick
Chief
Operations Analysis

Martin Marietta Corporation
P. O. Box 179
Denver, Colorado 80201

FOREWORD

This document represents the final technical report on JPL Contract 952028, A Study Program on the Development of Mathematical Model(s) for Microbial Burden Prediction. This report was prepared in accordance with the requirements established by the subject contract. The final report is submitted in three (3) volumes:

Volume I	Technical Report
Volume II	User's Manual for the Microbial Burden Prediction Model
Volume III	Appendices

CONTENTS

Several changes were made in the computer program as a result of work done in Phases IV and V of this contract. These changes are discussed in Volume IV, Chapter VI.

ILLUSTRATIONS

Figure | **Page**

Tables

Listing

I. SUMMARY

This report describes the capabilities, logical approach, data inputs, and deck arrangements for the Microbial Burden Prediction Model Computer Program.

The Microbial Burden Prediction Program, written in FORTRAN IV for the IBM 7094, is a highly versatile program for simulating the gain and loss of biota on the various surfaces of a spacecraft system during the steps of assembly, test, transport, decontamination, etc., associated with its preparation. Three accumulation mechanisms have been developed, any combination of which can be used at any step by specifying appropriate input values for the comprehensive set of coefficients, modifiers, and histogram points. Effective use of the program depends on an understanding of the program logic, the mathematical models involved, and of the significance of the input quantities. It is the purpose of this report to provide this understanding.

Section II, Introduction, defines the basic terminology used in the program such as the program organization levels Run, Stage, Task, Subtask, and Operation, and the significance of Surfaces, Parts, and Zones.

Section III, The Program Macrologic, describes the logical structure of the program (i.e., the sequencing of inputs, computations, outputs, and logical branchings). Flow diagrams are provided for the Control Program and the Microbial Buildup Subroutine which comprise the main part of the program and describes the contributions of the remaining subroutines.

Because the factors that determine the gain and loss of biota are generally unpredictable and can be given valid descriptions only in terms of probability distributions, it is essential to deal with distributions rather than with numerical quantities. For example, such unpredictable quantities are the initial burdens on parts, the microbial concentrations associated with workers, tools, and environments, and the time required to perform each operation. Each of these quantities is best described as a random variable with a defined probability distribution. The equations describing the mechanisms of environmental fallout, operational fallout, contact contamination, and decontamination are relations between random variables. The approach used here is to express the distributions of such variables in histogram form and to use a linear interpolation technique in determining the distributions of functions of these variables. This approach and technique are described in Section IV, Arithmetic Treatment of Histograms.

The mathematical models representing the mechanisms of environmental and operational fallout, contact, and decontamination are described in Section V, Microbial Burden Modification Models. These models are more fully described in Volume I of this report. Section V restates the equations and relates the equation quantities to the corresponding program variables.

Section VI, Input Data, lists the necessary input data and describes their arrangement, format, and significance. It is worthwhile to consider this section in conjunction with a study of the program macrologic (Section III).

Section VII, Outputs, describes the information that is printed in the Output and gives some representative Output Listings.

Finally, Section VIII, Job Control, Deck Arrangement, and Tape Setup describes the job order, arrangements of program and data decks, tape usage, and the associated job processor control cards for regular and special jobs.

II. INTRODUCTION

The Microbial Burden Prediction Program simulates the gain and loss of viable organisms by the various surfaces of a spacecraft assembly throughout the various assembly, test, transport, storage, and decontamination operations used in its preparation for launch. Determining factors in this simulation are the time intervals, personnel, tools, surface characteristics, and environments involved in the operations. The microbial burden on any surface at any step is given in terms of a histogram describing the probability distribution defined over the burden range. Time intervals and the concentrations of microbes at the sources of contamination are similarly given in terms of probability distributions in histogram form so that the resulting burdens reflect the inherent lack of certainty due to the natural randomness of the processes involved.

The organization of the assembly and test sequence procedures is by RUN, STAGE, TASK, SUBTASK, and OPERATION. A RUN represents a complete computation requiring a complete set of input data. No information can be transferred from one run to another except by input data cards. A STAGE (first level) is a grouping of tasks at the completion of which is printed a STAGE SUMMARY giving the total mean burden and variance after each Task of the Stage. A TASK (second level) is a grouping of subtasks that depend on a common set of inputs and which can be treated by one computer pass, i.e., each task is preceded by a control card and any necessary changes in the environments, operations, parts, or distribution inputs. A task is followed by a TASK SUMMARY giving the microbial burden for each surface and part.

At the SUBTASK level (third level), changes can be made in the part and surface designations (representing steps in assembly or disassembly), or in the environment or retention characteristics of the parts. Each subtask can consist of several OPERATIONS in which the actual gain or loss of microbial burden on the affected parts and surfaces takes place.

The gain and loss of microbial burden is described in terms of four mechanisms, each represented in the program by an appropriate model. These are:

1. ENVIRONMENTAL FALLOUT - the long term approach of the microbial burden to some asymptotic value determined by the environment, i.e., without considering personnel, tools, etc.

2. OPERATIONAL FALLOUT - the gain or loss of microbial burden during an operation due to the presence of sources of contamination but without actual contact.

3. CONTACT - the transfer of microbial burden by touching or brushing the surface with tool, hand, or other material.

4. DECONTAMINATION - the loss of microbial burden by killing some fraction of the burden determined by the agent and the accessibility of the surface.

These models are described more completely in Section V.

The microbial burdens are defined for the surfaces of parts. Each PART can have four SURFACES - top (1), exterior (2), mated (3), and occluded (4). This classification is made on the basis of accessibility of the surface to microbes from accretion, fallout and contact, and to decontamination agents. TOP surfaces are exterior surfaces on which falling microbes lie without falling off, although they may be blown off. EXTERIOR surfaces are the designation for outer surfaces other than top surfaces. MATED surfaces are in contact and have a very low accessibility. OCCLUDED surfaces are shielded from fallout to a degree specified by the appropriate input coefficients. In practice, the accessibility of each surface to sources of contamination or decontamination is specified in terms of input coefficients and the only inherent difference is that only top and exterior surfaces are subject to contamination by contact.

The classification by parts is to be made so that the microbial burden can be assumed to be effectively uniform over each surface of the part. In other words, a part need not be a physical unit but may be a group of items that are treated similarly in the operations and can be assumed to have the same burden distributions (e.g., the screws in a particular sub-assembly).

In order to determine the effect of heat sterilization techniques, a separate classification in terms of thermal zones can be made. Each ZONE is defined in terms of the fractions of surfaces that comprise it and the microbial burden is determined from the burdens of the constituent surfaces. The separate classification by zones is desirable because, while parts are assumed to have uniform burden distributions, thermal zones can be assumed to represent regions that respond similarly to heating.

A basic print-out of input data, operation times, and microbial burdens after fallout and contact is part of the program. Burden distributions for each part can be printed by use of an appropriate control card (see Section VI). Furthermore, the gain or loss of microbial burden called the Burden Difference between specified task (not necessarily part of the same stage) can also be obtained with the use of a suitable control card.

In addition, all generated distributions for time or burden are stored on a special tape so that a more detailed printout of data can be obtained by the use of a special program which re-reads this tape and prints the desired output.

Two additional special programs are the DATA CHECK program (which determines that the data cards are in the proper order, that all indices are in the required range, and that all prerequisite subtasks or tasks have indeed been considered prior to the subtasks or tasks for which they are prerequisites) and the RESTART subroutine (which determines the necessary inputs at the restart point from the special tape mentioned above and allows a continuation of the Microbial Burden Prediction Program from that point. These programs are described in more detail in Section VII.

III. THE PROGRAM MACROLOGIC

The Microbial Burden Prediction Program is a 4-part program:

1. MICROBIAL BURDEN DATA CHECK PROGRAM (MBDC1-MBDC2)

Program MBDC which consists of the data check program and an error counting subroutine, reads the input data from cards, performs certain checks (such as determining that the data is in recognizable order, that all indices are within the required ranges, that all prerequisite distributions and quantities have been defined, etc.), and records the data on a tape to be used as the input tape for subsequent programs. Any errors are tabulated by type of error and location. If any errors occur, the data must be corrected and again checked and recorded by Program MBDC.

2. MICROBIAL BURDEN SIMULATION PROGRAM (MBP1-MBP12)

This program which consists of 12 program decks, uses the input tape generated by MBDC and simulates the microbial buildup. A standard output is printed while a very detailed output is recorded on tape for later use by Program MBDR.

3. MICROBIAL BURDEN DATA RECOVERY PROGRAM (MBDR)

Program MBDR is designed to read the output tape generated by Program MBP. Additional instructions are added to the basic reading program to calculate any desired quantities and to print them out in any desired format.

4. MICROBIAL BURDEN SPECIAL WRITE-OUT PROGRAM (MBSW)

Program MBSW was written to provide a special listing of those inputs describing Run, Stage, Task, Subtask, Subtask Operations, and Parts Affected.

Each of these levels is appropriately indented and the listing provides a complete outline of the simulation. These inputs are obtained from the input data tape generated by Program MBDC.

The Microbial Burden Simulation Program consists of a Control or Main program (MBP1 and 11) subroutines. These subroutines are (see Table 4):

HCS Histogram Combining Subroutine (MBP2)

This subroutine determines the probability distributions for the random functions $z = x+y$, $z = x - y$, $z = x \cdot y$, $z = x/y$, and $z = \max(x,y)$ (see p 24 for discussion) where x and y are random variables with probability distributions described in histogram form (see Section IV).

HES Histogram Equating Subroutine (MBP3)

This subroutine sets the probability and range values of one histogram equal to those of another.

HMS Histogram Multiplying Subroutine (MBP4)

It is convenient to input one histogram to describe several distributions that have the same shape but possibly different ranges. To use the histogram for a particular application, it is only necessary to specify the desired mean value which is then used to modify the given histogram through the subroutine HMS.

HNS Histogram Numbering Subroutine (MBP5)

Certain histograms such as those describing time distributions, biota concentrations, and the microbial burdens present on the various surfaces of the parts of the space vehicle must be saved for later reference while other histograms are used only while making a computation and can thus be discarded. Histograms with indices from 1 to ND are saved while histograms with indices of the form I1 = ND + 1, I2 = ND + 2 may be discarded and the array members used for other temporary histograms. The quantity ND is set equal to the highest index of the input histograms and is then increased by 1 for each newly generated histogram that is to be saved by calling subroutine HNS. Calling this subroutine also sets the index of the new histogram equal to the new value of ND.

HWS Histogram Writing Subroutine (MBP6)

The procedure for writing histogram values is required so often that it is convenient to have a special subroutine to write these values in a given format.

MAS Microbial Accretion by Environmental Fallout Subroutine (MBP7)

This subroutine determines the current microbial burden on a given part due to environmental fallout since time AAT(I), the last update time for part I. The environmental fallout model is described in Section V.

MAVS Mean and Variance Subroutine (MBP8)

This subroutine determines and writes in the prescribed format the mean value and variance of any given histogram.

MBRS Microbial Burden Restart Subroutine (MBP9)

If an error in the data causes the program to abort before completion, the program can be used with a control card specifying restart after the last correctly computed stage and task. This subroutine causes the record tape to rewind and then reads the stored data until it comes to the restart point at which time the regular computation, output writing, and recording on tape are resumed.

MBS Microbial Buildup Subroutine (MBP10)

This is the subroutine that reads the input data for task, subtasks, and operations and computes the microbial burden on each affected surface as a result of decontamination, operational fallout, and contact (see Sec. V.).

MVS Mean Value Subroutine (MBP11)

This subroutine determines and stores the mean value of each generated histogram.

ZF Z - Function (MBP12)

This function subroutine determines specific values for the function $z = x+y$, $z = x-y$, $z = x \cdot y$, $z = x/y$, or $z = \max(x,y)$, for specific values of the arguments x and y.

Of these subroutines, only HCS, MAS, and MBS have any significant logical structure, the rest being relatively short and straightforward. HCS is described in Section IV and MAS in Section V. MBS, the longest and most complicated of the subroutines is described, along with the Control Program, in the remainder of this section.

The macrologic of the Control Program and Subroutine MBS is shown in the accompanying flow diagrams. Because subroutines MBRS and MBS are important parts of the logical structure, the places where they are called in the program are indicated. Other subroutines are used in the indicated computations or write instructions and are not referred to by name.

As can be seen in Figure 1, the quantities KK and IK appearing on each CONTROL CARD* (see Table 1 and the paragraph describing the CONTROL CARD in Section VI) determine the logic path and the succeeding input data to be read. KK = 1 starts a RUN and calls for reading a RUN DESCRIPTION CARD as well as STAGE andTASK DESCRIPTION CARDS. KK = 2 starts a STAGE and calls for reading STAGE and TASK DESCRIPTION CARDS. KK = 3 starts a TASK and calls for reading a TASK DESCRIPTION CARD. KK = 4 calls for a Burden Difference calculation and write-out. KK = 5 calls for a determination of the microbial burden by thermal zones. KK = 6 is used to restart the program at some point where the previous calculations are valid but prior to data changes.

For KK = 1, 2, 4, 5, the previous pass may have completed a Stage. If so, a Stage Summary is first printed before proceeding to this pass. For KK = 3, the previous pass cannot have completed a Stage and, for KK = 6, there was no previous pass since KK = 6 is equivalent to starting a Run but not duplicating any valid computations.

The significance of IK depends on the value of KK. For KK = 1, 2, 3, the Control Program will call Subroutine MBS where IK determines the data to be read (see Figure 2). For KK = 4, 6, IK is left blank and for KK = 5, IK determines whether the zones are to be defined or to be the same as parts.

Subroutine MBS (Figure 2) is called to determine the microbial buildup during a Task. Before starting this calculation, additional data can be read by giving IK on the CONTROL CARD an appropriate value as shown in the branches for IK in Figure 2. A blank card or one with 0 in the first field must follow each set of inputs to move to the next set of inputs. After completing the

* Although the Control Program and Subroutine MBS actually read a tape for input data, the data must be prepared in card form and recorded on tape by Program MBDC so will be referred to here in terms of CARD input.

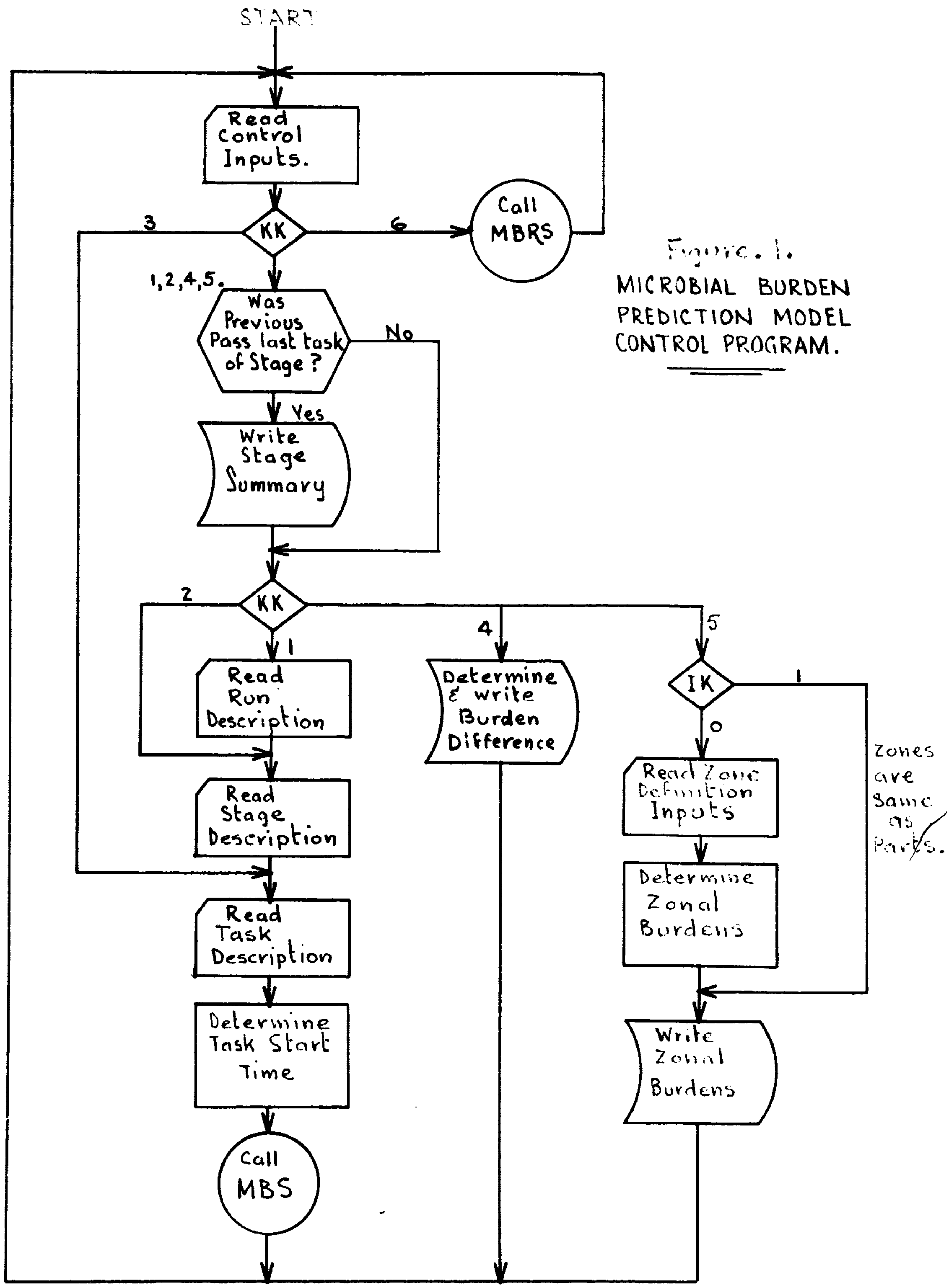


Figure. 1.

MICROBIAL BURDEN PREDICTION MODEL CONTROL PROGRAM.

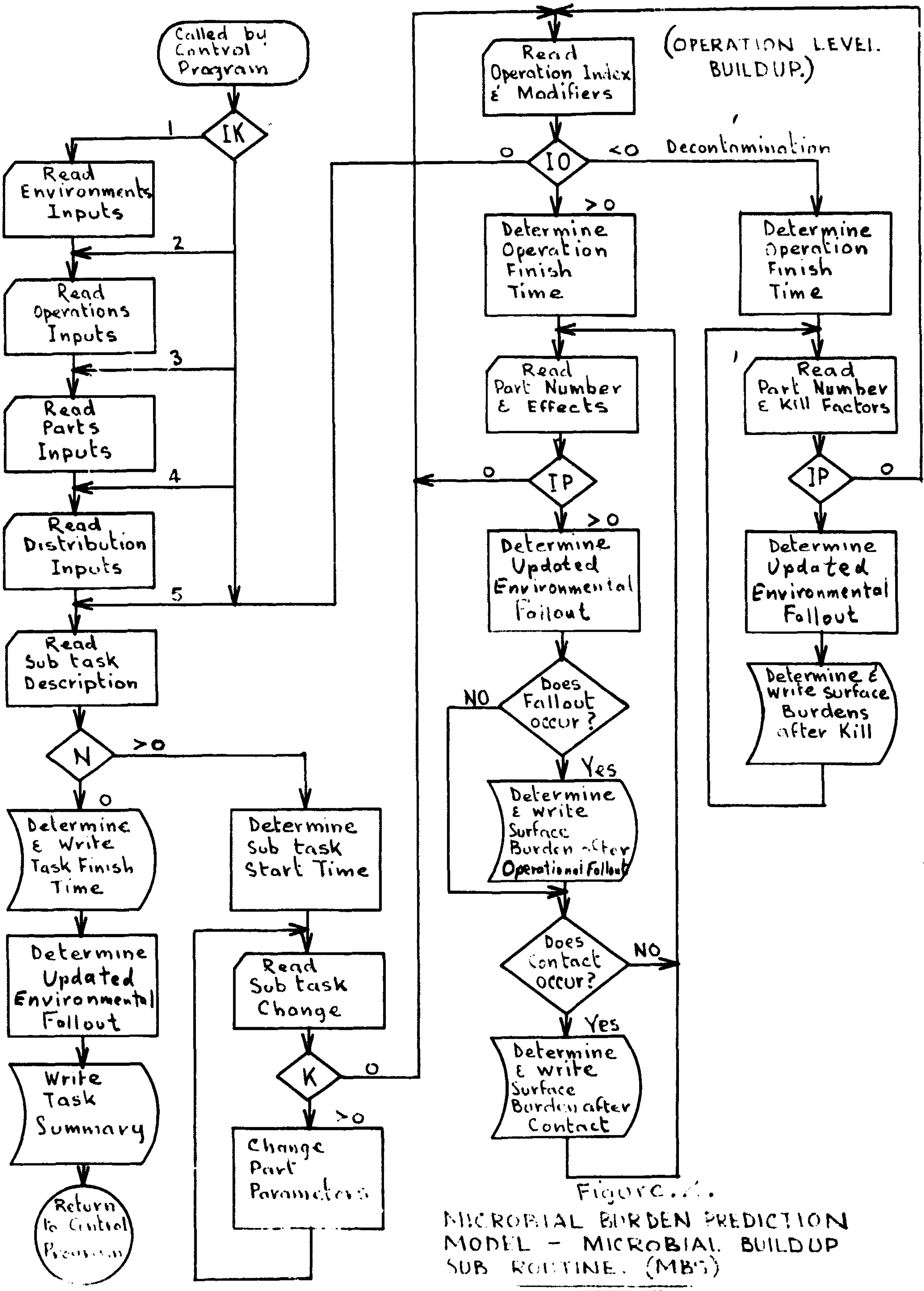


Figure 4.
MICROBIAL BURDEN PREDICTION MODEL - MICROBIAL BUILDUP SUB ROUTINE. (MBS)

Environments, Operations, Parts, and Distributions inputs , the program reads Subtask, Operation, and Part Effect card (see Table 1, Section VI) using blank cards (N=0, K=0, IO=0, and IP=0). This is described more fully in Section VI. When the last Subtask has been considered, indicated by a blank SUBTASK DESCRIPTION CARD, the subroutine ascertains that all burdens include accretion until the finish Time of the Task and prints these burdens as the Task Summary.

The Task and Subtask start times may depend on the finish times of the specified prerequisite Tasks and Subtasks. If there is one prerequisite, the start time is equal to the finish time of the prerequisite, but if there are two prerequisites, the start time is based on the function z = max (x,y), i.e., the probability for any time z is the probability that both prerequisites have been finished by that time. The prerequisites for a Subtask can only be earlier Subtasks within a Task while the prerequisites for a Task can be any earlier Task, in the same Stage or not. The finish times for such Tasks, however, must be saved by specifying an appropriate value for L3 on the TASK DESCRIPTION CARD (which is seen in Section VI).

Operations within a Subtask, however, are assumed to follow each other in sequence so that the start time of one is equal to the finish time of its predecessor. The start time of the first Operation of a Subtask is equal to the start time of the Subtask and finish time of the last Operation determines the finish time of the Subtask. Since Subtasks do not necessarily follow in sequence, the finish time of a Task is equal to the latest Subtask finish time.

Other computations are straightforward or are described in Section V.

IV. ARITHMETIC TREATMENT OF HISTOGRAMS

Because the operation times, the number and distances of the personnel from the parts whose contamination is to be considered, and the microbial concentrations in the air and on the personnel and tools must be represented by random quantities, the microbial burden buildup is also a random quantity. A possible approach would be to use Monte Carlo trials and determine a mean and variance from the results but this would involve a question of confidence based on the number of trials as well as the confidence of the original data inputs. In this program, random variables are represented by probability distributions defined in terms of histograms. This is equivalent to using a series of straight line segments to approximate the cumulative probability curve. This provides a straightforward approach to determining the functions: $z = x+y$, $z = x-y$, $z = x(y)$, $z = x/y$ and $z = \max(x,y)$ where x and y are independent random variables.

It must be pointed out that this approach is an approximation. In the first place, if x and y are random variables whose probability distributions are accurately represented by histograms, then the random function z for any of the above operation may not have a distribution accurately represented by a histogram and that only in the limit as every value of x or y has its own defined probability will the probability distribution for z be a histogram. Furthermore, the number of points on the cumulative distribution curve for z must be equal to the product of the points for x and for y (minus the number that coincide) or accuracy is lost by neglecting points that are significant. If, in addition, the contributions to the ranges of z are determined by linear interpolation rather than by a more accurate representation of the distributions, considerable error may be introduced unless each histogram for x and y has many intervals. In spite of these disadvantages, the histogram approach provides a procedure for determining the random functions and the inaccuracies can be reduced by increasing the number of points considered.

To illustrate the procedure used, consider the random variables x and y and their sum z = x+y. Let x and y have the probability distribution

$$P\ (1 \leq x < 5) = .4$$
$$P\ (5 \leq x < 9) = .6$$

$$P\ (1 \leq y < 3) = .4$$
$$P\ (3 \leq y < 6) = .6$$

Then the probability that z will lie in a certain range interval is equal to the combined probability that a point represented by a pair x, y lies in the region for which z = x+y takes on values in that interval. This is shown in Figure 3.

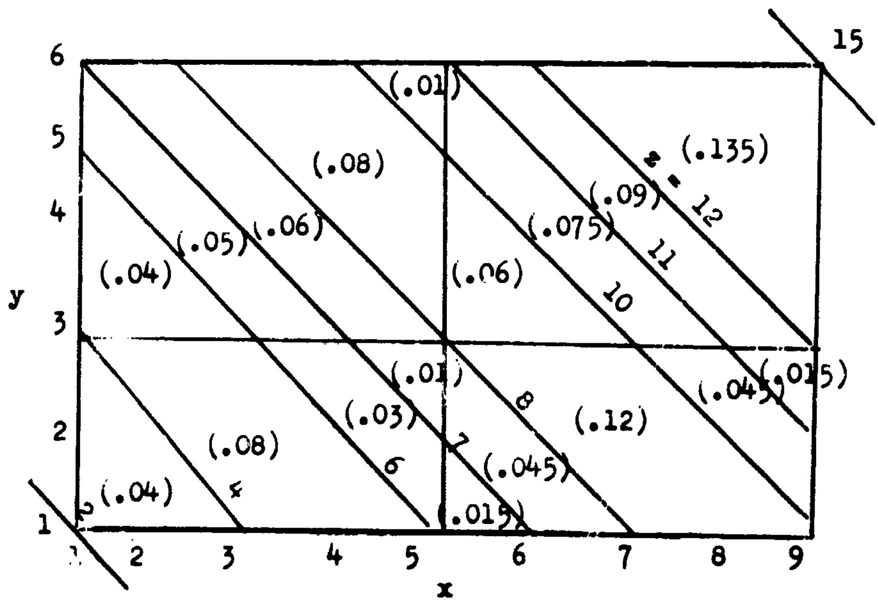


Figure 3.

A Typical Arithmetic Operation
Using the Interval Concept

Note that the z lines are drawn through the intersections of the x and y lines since these intersections represent abrupt changes in the probability levels. The probabilities that the point for x and y lie in each region is given in parentheses. The resulting probabilities for the z intervals are

$$P(2 \le z < 4) = .04$$
$$P(4 \le z < 6) = .12$$
$$P(6 \le z < 7) = .095$$
$$P(7 \le z < 8) = .115$$
$$P(8 \le z < 10) = .26$$
$$P(10 \le z < 11) = .13$$
$$P(11 \le z < 12) = .105$$
$$P(12 \le z < 15) = .135$$

These probabilities are exact if it is assumed that the imput probabilities are exact. However, it is easily seen that the distribution over each interval is not linear and that a histogram representation is not exact. Also, the number of points must be reduced or, after a few such computations, it will become prohibitively high.

To keep the number of points within reason, the program calculates as many z values as the maximum of the number of x or y values. These z values are chosen at the intersections most likely to represent the most abrupt changes in probability level. In particular, the range limits are always retained. In the above example, the z values kept would be 2, 8, 15 and the resulting histogram would be as shown in Figure 4.

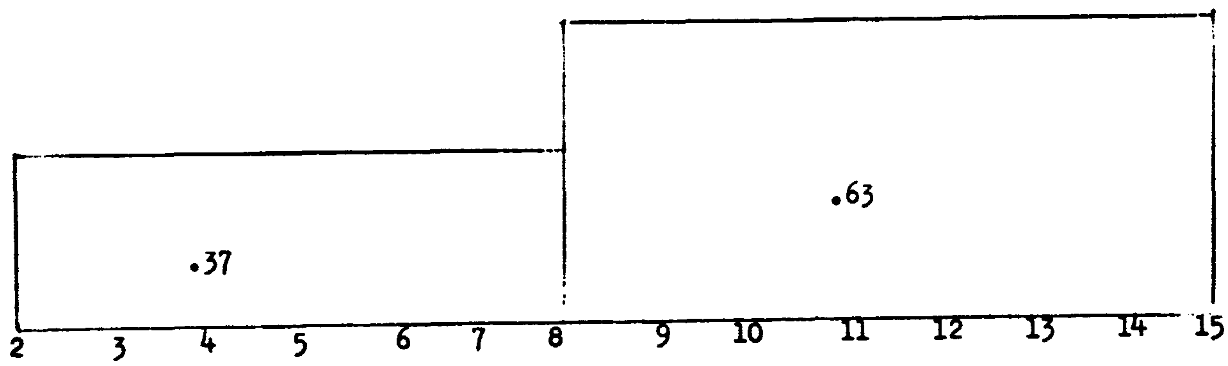
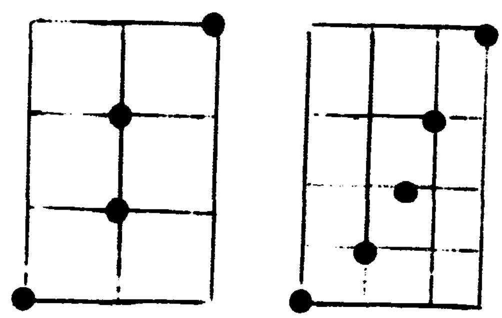


Figure 4.

A Typical Result from Combining Two Histograms

The procedure for selecting z values is to choose the z range limits (2 and 15) then to choose the x,y intersections diagonally in from these curves. If the number of x points and y points is not the same, the intersections are chosen along a central line as shown in Figure 5.



If there is no central line, then a half-way point is chosen as shown. However, this can result in histograms with narrow intervals and low probabilities so that it is worthwhile to input histograms with the same number of points.

Figure 5.
Choosing Intersections for z Values

The probability contribution of each x,y region to each z range is assumed to be proportional to the parts of the z range in the region. Thus the contributions in the example given above would be as shown in Figure 6.

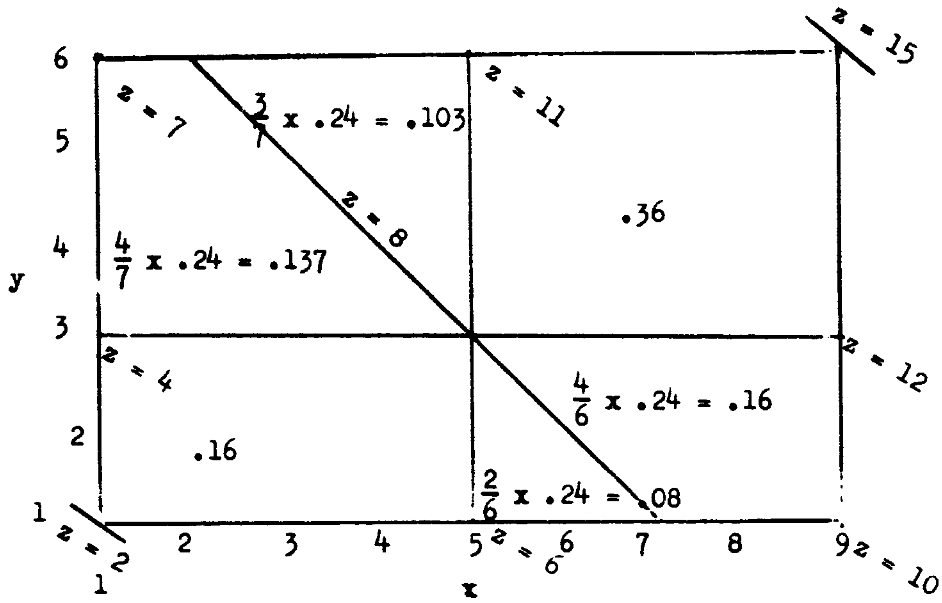


Figure 6.
Contributions by Proportion for Addition

The z probabilities are then

$$P\ (2 \leq z < 8) = .137 + .16 + .08 = .377$$

$$P\ (8 \leq z < 15) = .103 + .36 + .16 = .623$$

which is reasonably close to the distribution determined by summing over z ranges where z values are computed for each intersection. However, linear interpolation can lead to a greater error in the case of other operations. For example, consider the function $z = x \cdot y$ where x and y are defined as above. The range diagram with the proportional probability distributions is shown in Figure 7.

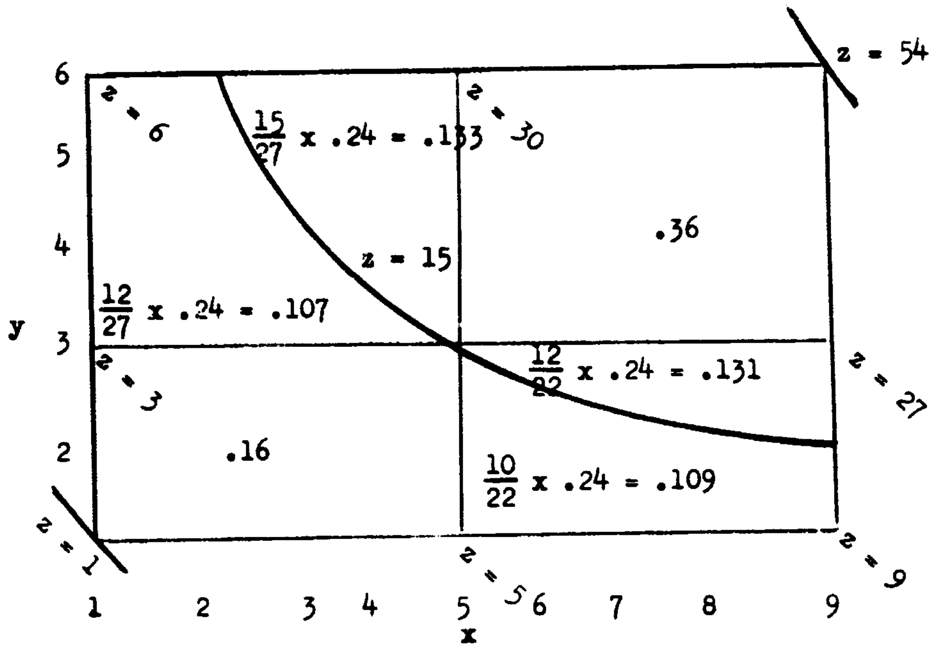


Figure 7.

Contributions by Proportion for Multiplication

These contributions (e.g., .107 to the range $1 \leq z < 15$ and .133 to the range $15 \leq z < 54$ for the upper left hand region) represent the linear proportions. However, the contributions should be proportional to the areas in each range as shown in Figure 8.

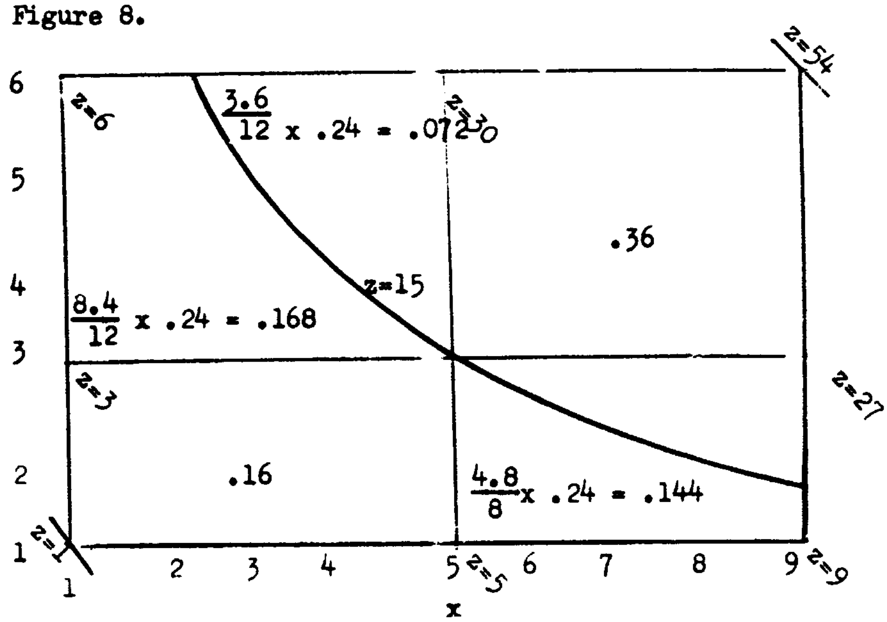


Figure 8.

Contributions by Area for Multiplication

The linear approximation and accurate values for the z probability are:

	Linear	Accurate
$P(2 \leq z < 15)$	.376	.472
$P(15 \leq z < 54)$	.624	.528

Three point histograms can be expected to introduce considerable error and the importance of using histograms with more points is easily seen.

An additional source of error is division by zero and it is imperative that histograms used as divisors not have ranges that include zero. These histograms however, represent quantities that logically, should never be zero so that careful selection of input quantities will avoid this problem.

The functions $z = x+y$, $z = x \cdot y$, and $z = x/y$ where x and y are independent random variables are computed by procedures similar to that described above, differing only in the equations used for determining the z values at the x,y intersections. The functions $z = x - y$ and $z = \max(x,y)$ (i.e., z is equal to the maximum of any pair of values x and y), however, require a different approach.

The function $z = x - y$, as used in this program, always involves dependent rather than independent variables, i.e., the distribution sought is not simply the distribution of the difference between independent random variables but is instead the distribution of a quantity z which, when added to y (which has some known distribution) will give the value x with a given distribution. To take an extreme example, suppose x and y have identical ranges and probability distributions. If x and y are assumed to be independent, z could take on values ranging from $x_{min} - y_{max}$ to $x_{max} - y_{min}$. However, adding such a quantity z to y will not produce x. Instead, the random variable sought is equal to 0 with probability 1 since adding this distribution to y will produce x.

The procedure used is best demonstrated by using the following diagram as an example:

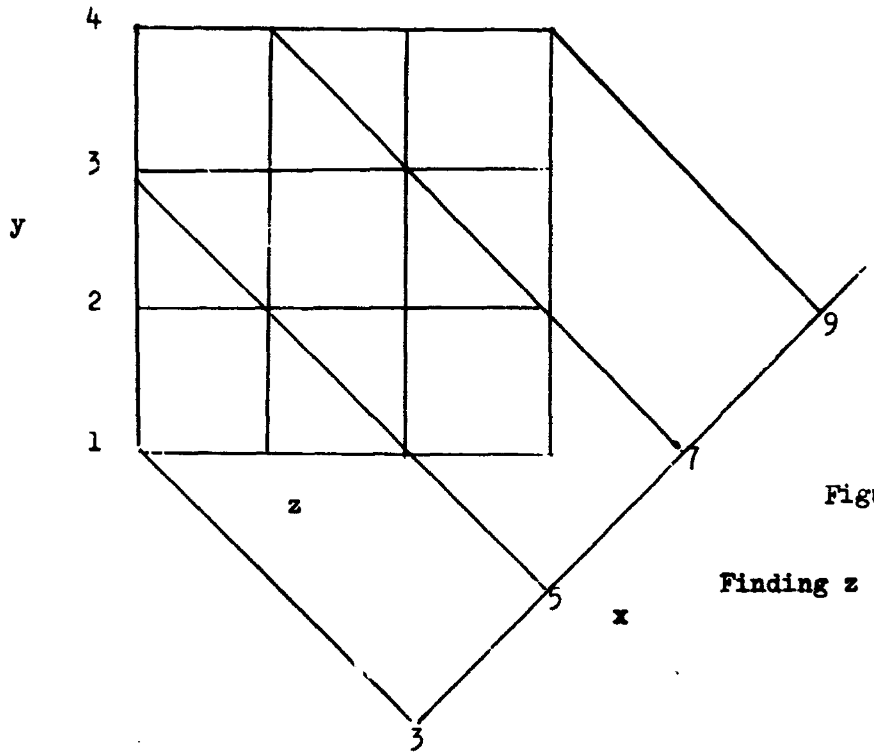


Figure 9

Finding z = x - y

Note the resemblence to the diagram for x = z + y. But x and y are the variables with the known distributions. Suppose:

$$P(3 \leq x < 5) = .2$$
$$P(5 \leq x < 7) = .6$$
$$P(7 \leq x < 9) = .2$$

$$P(1 \leq y < 2) = .2$$
$$P(2 \leq y < 3) = .6$$
$$P(3 \leq y < 4) = .2$$

The problem is to find ranges of z and calculate the probabilities. The ranges are not difficult and are seen to have the values

$$z_1 = x_1 - y_1 = 2$$
$$z_2 = x_2 - y_2 = 3$$
$$z_3 = x_3 - y_3 = 4$$
$$z_4 = x_4 - y_4 = 5$$

For the case where x and y have different numbers of values, an adjustment is made similar to that in determining $x = z + y$. Also, it should be noted, z need not have as many values as x or y. For example, consider the case (suggested earlier) where x and y are equal.

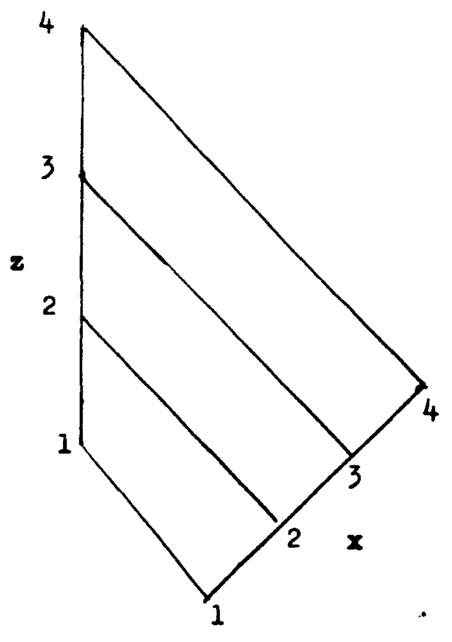


Figure 10

The Increment Which Leaves A Distribution Unchanged

Obviously, the intersections are along the single line $z = 0$. It is also possible that no acceptable distribution for z (intervals between monotonic increasing values of z having positive probabilities which sum to unity for the entire z - range) will fit the requirements of being the increment which produces the x distribution from the y distribution ($x = y + z$). In such cases, the values of z and the probabilities must be adjusted to obtain that acceptable distribution most likely to be the increment sought.

The probablities are then determined by forming a set of simultaneous equations. For the example cited, this would be:

$$\left(.2 + \frac{.6}{2}\right)P_1 + \frac{.2}{2}P_2 = .2$$

$$\left(\frac{.6}{2} + .2\right)P_1 + \left(\frac{.2}{2} + .6 + \frac{.2}{2}\right)P_2 + \left(.2 + \frac{.6}{2}\right)P_3 = .6$$

$$\frac{.2}{2}\ P_2 + \left(\frac{.6}{2} + \ 2\right)P_3 = .2$$

The solution is seen to be

$$P_1 = P(2 \leq z < 3) = 1/3$$

$$P_2 = P(3 \leq z < 4) = 1/3$$

$$P_3 = P(4 \leq z < 5) = 1/3$$

If z should have fewer values than x or y, the set of equations is redundant and an adjustment is necessary to obtain the most reasonable distribution for z.

The function $z = \max(x,y)$ is used to find the start time probability distribution for a task or subtask that depends on the finish times of two prerequisite tasks or subtasks, i.e., the probability distribution for z is to be the probability that both prerequisites have been completed. For any pair of values x and y the cumulative probability for $z = \max(x, y)$ is the product

$$P_z(z) = P_x(z) \cdot P_y(z)$$

where $P_z(z)$ is the probability that both prerequisites will have been completed by time z, $P_x(z)$ is the probability that prerequisite x will have been completed by time z and $P_y(z)$ is the probability that prerequisite y will have been completed by time z. The procedure is to transform the x and y histograms into cumulative distributions (they will consist of straight line segments). Then the combined probability is found at each point z. Those points with z less than the highest value of z for which $P_z(z) = 0$ and those points with z greater than the lowest value of z for which $P(z) = 1$ are dropped. If the remaining number of values is greater than the maximum of the numbers of x points and y points, additional z values are dropped, the dropped values being chosen as the least significant to the resulting distribution. This is shown in Figure 11. In this example, since the x and y distributions have 5 points, only the five circled points of the z distribution are retained.

Three arrays are used in the program to store the distribution data, NX, DR, and XR. For histogram I, NX(I) is the number of points for the histogram, DR(I,J) is the probability that x lie in the range XR(I,J-1) to XR(I,J). Since there is one less probability value than the number of points, DR(I,1) is used as a coefficient to store the surface area or a mean value, depending on whether the histogram represents a biota distribution or a time or concentration.

In operation, Subroutine HCS has the arguments IA, IB, IC, K. IA and IB are the indices for the x and y distributions, IC is the index for the resulting z distribution. K indicates the operation:

K = 1, $z = x+y$
2, $z = x-y$
3, $z = x \cdot y$
4, $z = x/y$
5, $z = \max(x,y)$

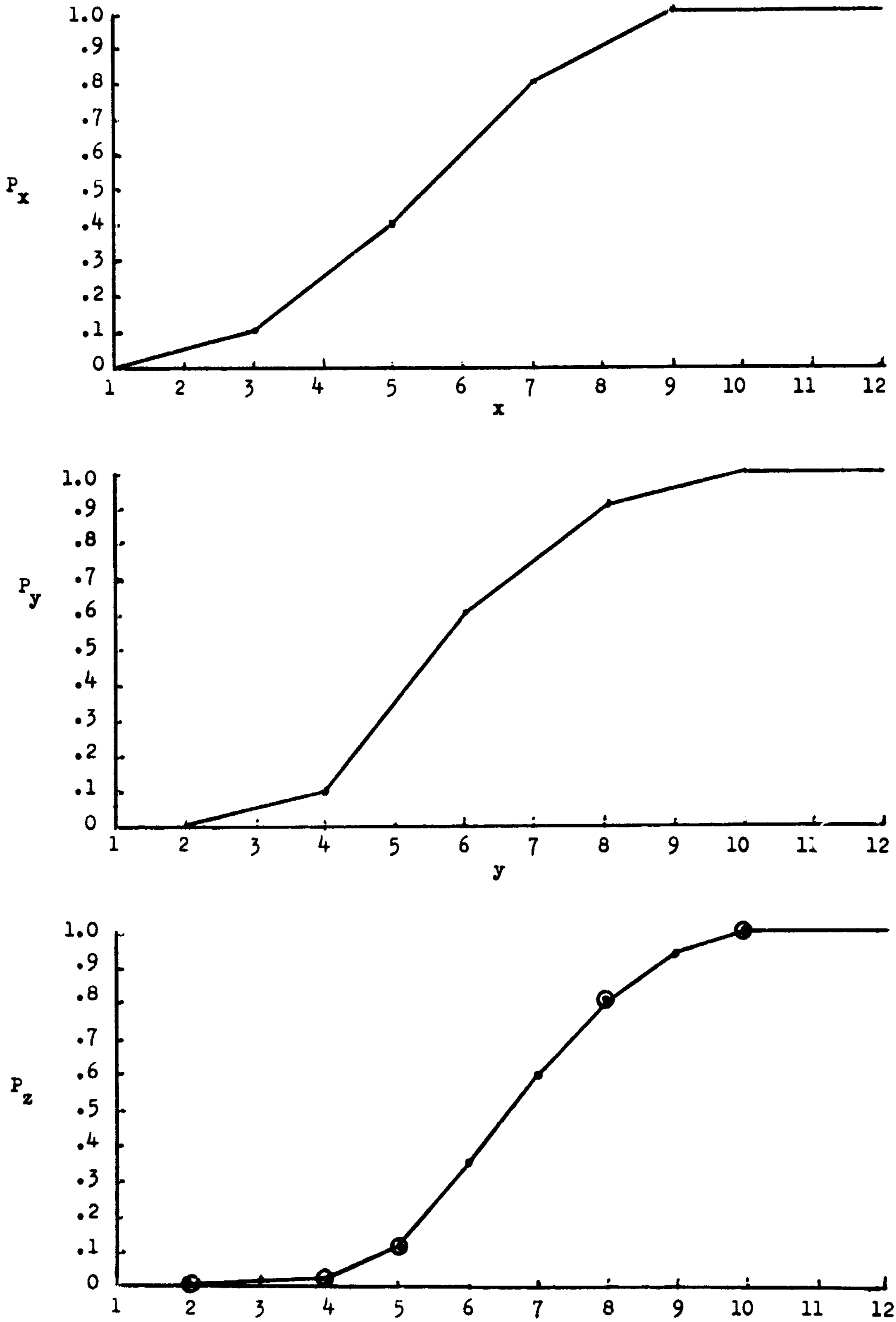


Figure 11. Determining z = max (x,y)

For example, calling Subroutine HCS (2,3,4,1) will add distributions 2 and 3 and store the sum as distribution 4.

V. MICROBIAL BURDEN MODIFICATION MODELS

The microbial burden on each surface is assumed to change by four mechanisms, environmental and operational fallout, contact, and decontamination. Of these, operational fallout, contact, and decontamination are computed at the operational level. Environmental fallout is due to the environment that is present when personnel, tools, and other such sources of fallout or contact contamination are absent. The theory underlying these models is explained in Volume I of this Final Report and is not repeated here. The equations are:

Environmental Fallout:

$$B' = B\, e^{-t/v} + A\, v\, R(1 - E^{-t/v})$$

where $R = f_j\, gc$

Operational Fallout:

$$B' = B\, e^{-t/v} + A\, v\, R(1 - e^{-t/v})$$

where $R = f_j\, g(c + Q\, e^{-\lambda d})$

Contact:

$$B' = B - \frac{B\, a\, s_2}{2A} + \frac{b_t\, a\, s_1}{2}$$

Decotamination:

$$B' = B(1 - k_j)$$

where B is the prior burden and B' the new burden. The histogram with index IB = IAB(I,J), where I is the part number and J the surface, describes the distribution of B prior to the change and the distribution of B' afterward. e is the natural logarithm base.

Other equation quantities and their relations to program quantities are as follows:

A = DR(IB,1) is the surface area in square feet.

t is the time interval involved, either the time since the last update for environmental fallout or the operation interval for operational fallout. The first of these is determined as a single quantity (mean time) by the program from the current time and time of last update for the part (AAT(N) while the second is a distribution that depends on the operation time distribution IOT(N) of the operation N and the modifier AKT specified on the SUBTASK OPERATION CARD.

v = IET(IE) • (AET(IE) • AES(J)/DR(IET,1)) is the "average lifetime" of the organisms in hours. IET(IE) is the lifetime distribution index for an organisms in environment IE, AET(IE) is the mean lifetime, AES(J) is the mean value of histogram IET. All three letter quantities with E as the central letter are specified with environmental inputs, i.e., on EM or EQ cards (see Section VI). Since IET is the index of a histogram, the multiplication indicated by • is not of the index but of the histogram indicated (see Section IV). It should also be noted that for each specified histogram index, the histogram values for that histogram must be given on DQ type cards. The environment IE for environmental fallout is kept in the IAB(I,J) array (see card type PQ, Section VI) for each part I,

$$IE = IAB(I,5)$$

while the environment for operational fallout is specified at the task or operational level,

$$IE = IKE \quad \text{if} \quad IKE > 0$$

$$= ITE \quad \text{if} \quad IKE = 0$$

where IKE is given on card type KO, ITE on a CC type card.

f_j = AEF(IE,J) if the fallout velocity in feet per hour is an input on an EQ card.

g = IG ^{o}(AAG(N)/DR(IG,1)) is the surface retention distribution for fallout. IG = IAB(N,6) is the retention curve index for part N where IAB(N,J) is given on the PQ card.

c = IEC(IE) ^{o}(AEC(IE)/DR(IEC,1)) is the airborne concentration distribution for environment IE. AEC is the concentration modifier. Both quantities are specified on the EQ card.

Q = IOQ(IO) ^{o} (AKQ/DR(IOQ,1)) is the personnel "dirtiness" distribution where IOQ(IO) is the dirtiness curve index given on card OQ, AKQ is the dirtiness modifier given on card KO.

λ = AED(IE), the distance coefficient for operational fallout, is a function of the environment and is given on a EQ card.

d = APD is the effective distance of the contaminating personnel from the affected part. APD is given on a PE card.

a = APA(J) is the area of contact for contact contamination. APA(J) is the given on a PE card.

s_1 = IG ^{o} (AAS(IP)/DR(IG,1)) is the surface retention distribution. IG, the retention distribution, as IAB(IP,6) and AAS(IP), the retention modifier for contact, are given on the PQ card.

s_2 = LS ^{o} (APS/DR(LS,1)) is the tool retention distribution.
LS and APS are both given on the PE card.

b_t = IOC(IO) $^{\circ}$ (APC/DR(IC,1)) is the contamination density distribution on the tool. IOC(IO) for operation IO is given on card OQ, APC is given on card PE.

k_j = LK$^{\circ}$ (AR(J)/DR(LK,1)) is the kill distribution, i.e., the fraction of organisms removed by the decontamination operation. LK and AR(J) are given on the PE card for decontamination.

VI. INPUT DATA

Table 1 lists all types of input data cards and shows their arrangement. Alternate branches are indicated by the flow lines on the left with the index values that cause the program to take that branch. The brackets on the right indicate cards or groups of cards that are repeated until a blank card occurs. Certain control cards do not call for data and are followed by another control card. These are control cards with KK=4 (calling for a printout of a Burden Difference), KK=5, IK=1 (calling for a zone printout without zone definitions), or KK=6 (for restarting the program at some specified stage and task for which prerequisite data has previously been stored on the special record tape).

The data contained on each card, their format, and their significance or use in the program are described next. A short listing follows the Data Check Program in Listing 1. It should be noted that all integer constants must be next to the right hand side of their field (right adjusted).

CC CONTROL CARD

KK, IK, I1, I2, I3, I4, I5, I6

FORMAT (8I5)

A CONTROL CARD is used as the first card of the input data deck, after the input data for a Task to prescribe the next step, and after the CONTROL CARDS mentioned above that are not associated with a data set. The significance of the data is as follows:

KK=0 causes the program to terminate without reading any additional data.

KK=1 indicates that a RUN is to be initiated and calls for RUN, STAGE, and TASK DESCRIPTION CARDS and TASK data inputs.

TABLE 1.

Input Data Deck

	CC	CONTROL CARD
KK=1	RD	RUN DESCRIPTION CARD
KK=2	SD	STAGE DESCRIPTION CARD
KK=3	TD	TASK DESCRIPTION CARD
IK=1	EM	ENVIRONMENTAL SURFACE LIFETIME MODIFIERS
	ED	ENVIRONMENTAL DESCRIPTION CARD
	EQ	ENVIRONMENTAL QUANTITIES CARD
IK=2	OD	OPERATION DESCRIPTION CARD
	OQ	OPERATION QUANTITIES CARD
IK=3	PD	PART DESCRIPTION CARD
	PQ	PART QUANTITIES CARD
IK=4	DD	DISTRIBUTION DESCRIPTION CARD
	DQ	DISTRIBUTION QUANTITIES CARD (1 or 2)
IK=5	KD	SUBTASK DESCRIPTION CARD
	KC	SUBTASK CHANGE CARD
	KØ	SUBTASK OPERATION CARD
	PE	PART EFFECT CARD
KK=5, IK=0	ZD	ZONE DESCRIPTION CARD
	ZC	ZONE COMPOSITION CARD

KK=3 indicates that a TASK is to be initiated and calls for a TASK DESCRIPTION CARD and TASK data inputs.

For KK=1, 2, 3, IK determines the point at which TASK inputs start. These inputs and the values of IK for which they are read are:

IK=1	ENVIRONMENTS (EM and all subsequent data)
IK=2	OPERATIONS (OD and all subsequent data)
IK=3	PARTS (PD and all subseque t data)
IK=4	DISTRIBUTIONS (DD and all subsequent data)
IK=5	SUBTASKS (SD and all subsequent data)

I1, I2, I3, I4, I5, and I6 are used as follows:

IF I1 > 0,	NE=I1	(number of environments
IF I2 > 0,	NO=I2	(number of operations)
IF I3 > 0,	NP=I3	(number of parts)
IF I4 > 0,	ND=I4	(number of distributions)
IF I5 > 0,	NS=I5	(number of subtasks)
IF I6 > 0,	ITE=I6	(specified task environment)

Resetting these values may help to determine whether the numbers NE, NO, NP, ND, or NS are increased during some portion of a run. Generally these values will be left blank except for I6 which must be specified if any fallout is to be determined. However, it need only be specified on the first input card and whenever it is to be changed.

KK=4 calls for determining and printing the Burden Difference, i.e., the difference between the total microbial burden at the finish of Stage I1, Task I2, which is saved as special distribution I3 and the burden at the finish of Stage I4, Task I5, which is saved as special distribution I6. The stage and task numbers are used only in the print-out. The distribution indices I3 and I6 identify the distributions whose difference is to be found and must have the same values as L4 specified on the appropriate TASK DESCRIPTION CARDS. For example, suppose in Stage 1 we have the TASK DESCRIPTION CARD

2 ALPHAMERIC DESCRIPTION OF TASK 2 0 0 0 11

and in Stage 4 we have the TASK DESCRIPTION CARD

62 ALPHAMERIC DESCRIPTION OF TASK 62 0 0 0 12

A subsequent CONTROL CARD calling for the Burden Difference would be

4 1 2 11 4 62 12

This Burden Difference is the increment which, added to distribution 11, yields distribution 12 where 11 and 12 are the indices where the total burden at the end of these tasks are stored.

A CONTROL CARD with KK=4 is followed by another CONTROL CARD

KK=5 calls for the printing of areas and microbial burdens by zones.

IK=0 indicates that the zones are to be defined by a deck of ZONE DEFINITION CARDS.

IK=1 indicates that the zones are the same as parts and that no ZONE DEFINITION CARDS are necessary. This card is to be followed by another CONTROL CARD.

KK=6 calls for a restart following Stage I1, Task I2. It is assumed that computations for this and all previous stages have been successfully performed and the data recorded on the permanent record tape. This card is to be followed by the CONTROL CARD initiating the next Stage or Task and all succeeding data cards in their normal order.

RD RUN DESCRIPTION CARD

KR, RUN

FORMAT (I5, 1X, 7A6)

A RUN DESCRIPTION CARD is the first card to follow the CONTROL CARD initiating a Run. KR is the Run Number and RUN is a 42 alphameric character description of the run. There is no limit to the allowed number of runs.

SD STAGE DESCRIPTION CARD

KS, STG

FORMAT (I5, 1X, 7A6)

A STAGE DESCRIPTION CARD follows a RUN DESCRIPTION CARD or a CONTROL CARD initiating a Stage. KS, which must be in the range 1 to 20 is the Stage Number and STG is a 42 alphameric character description of the Stage.

TD TASK DESCRIPTION CARD

KT, TSK, L1, L2, L3, L4

FORMAT(I5, 1X, 7A6, 2X, 4I5)

A TASK DESCRIPTION CARD follows a STAGE DESCRIPTION CARD or a CONTROL CARD initiating a task. KT, which must be in the range 1 to 100, is the Task Number and TSK a 42 alphameric character description of the task. L1 and L2 indicate the finish time distributions of prerequisite tasks which are to determine the start time of this task. If only one prerequisite (L1) is specified, the new start time is set equal to the distribution for L1. If no prerequisite is specified, the start time is set equal to zero. L3 is the index of the finish time distribution if this task is to be prerequisite for any later task. L4, if specified, performs two tasks. First, it causes all burden distributions to be written by part and surface in the task summary. Second, it causes the total microbial burden at the end of the task to be stored as distribution L4 and to be available for later computation of a Burden Difference. IK on the preceding CONTROL CARD indicates the card type to follow.

EM ENVIRONMENTAL SURFACE LIFETIME MODIFIERS CARD

AES(1), AES(2), AES(3), AES(4)

FORMAT (4E10.3)

The ENVIRONMENTAL SURFACE LIFETIME MODIFIERS appear on the first card of the Environments Inputs. AES(J) is the factor which modifies the quantity v for Surface J (See Section V.)

ED ENVIRONMENTAL DESCRIPTION CARD

N, DSC

FORMAT (I5, 1X, 4A6)

Each of the (up to 10) environments requires two input cards. The first of these is the ENVIRONMENT DESCRIPTION CARD which gives the index N and the 24 alphameric character description DSC. The last such description card is to be blank to indicate that all environment inputs have been read. All but the last blank ENVIRONMENT DESCRIPTION CARD are to be followed by an ENVIRONMENT QUANTITIES CARD.

EQ ENVIRONMENT QUANTITIES CARD

IEC(N), IET(N), AEC(N), AET(N), AED(N), AEF(N,1), AEF(N,2), AEF(N,3), AEF(N,4)

FORMAT (2I5, 7E10.3)

In order to reduce the number of required histograms, a given histogram is used primarily to define the shape of the distribution and can have a different mean value prescribed for each application. For each application, then, a histogram and a mean value are prescribed, e.g., IEC(N), AEC(N).

IEC(N), AEC(N) describe the microbial concentration associated with environment N. IEC is a distribution index and AEC is the mean of the distribution for this application.

IET(N), AET(N) describe the reference environmental fallout lifetime distribution.

AED(N) is the distance coefficient (see the description of the Operational Fallout Model).

AEF(N) is the velocity at which oranisms settle.

OD OPERATION DESCRIPTION CARD

N, DSC

FORMAT (I5, 1X, 4A6)

Each of the (up to 20) cataloged operations requires two input cards. The first of these is the OPERATION DESCRIPTION CARD which gives the index N and the 24 alphameric character description DSC. The last of these cards is to be blank to indicate that all operations inputs have been read. All but the last blank card are to be followed by an OPERATION QUANTITIES CARD.

OQ: OPERATION QUANTITIES CARD

IOT(N), FOQ(N), IOC(N)

FORMAT (3I5)

IOT(N) is the index for the distribution describing the time required to perform operation N. Modification of this time is made when the operation is used under a subtask.

IOQ(N) is the biota concentration distribution associated with the source of contamination, personnel, etc) when this operation is used (see the description of the Operational Fallout Model, Section V). This distribution can be modified for each particular situation in which the operation is performed.

IOC(N) is the biota concentration on hands, tools, clothing, etc., for the case of contamination by contact.

PD: PART DESCRIPTION CARD

N, DSC

FORMAT (I5, IX, 6A6)

Each of the (up to 120) initial parts requires two input cards. The first is the PART DESCRIPTION CARD giving the part number N and the 36 alphameric character description DSC. The last 12 characters (in columns 31 through 42) are retained as a permanent serial description of the part. The last of these cards is to be blank to indicate that there are no more parts inputs. All but the blank card are followed by a PART QUANTITIES CARD.

PQ: PART QUANTITIES CARD

IAB(I,1), IAB(F,2), IAB(I,3), IAB(I,4), IAB(J,5) IAB(I,6),

AAG(N), AAS(N)

FORMAT (6I5,2E10.3)

IAB(1) is the index of the distribution of microbial burden on the top surface of the part. The area of this and the other surfaces is given in the coefficient associated with the distribution (see the explanation for DQ below).

IAB(2) is the index of the distribution for area and burden of all other exterior surfaces.

IAB(3) is the index of the distribution for area and burden of the mated surface.

IAB(4) is the index of the distribution for area and burden of the occluded surface.

IAB(5) is the environment of the part used in determining microbial accretion.

IAB(6) is the part retention distribution index.

AAG is the surface retention factor for fallout contamination.

AAS is the mean surface retention factor for contact contamination.

DD: DISTRIBUTION DESCRIPTION CARD

N, DSC, L, M

FORMAT (I5, 1X, 4A6, 2I5)

Arrays are defined for storing up 551 histograms, each with a maximum of 11 absisssa values and their associated probabilities. However, the last 51 of these are reserved for special uses and only 500 places are available for general use. Since the program automatically indexes new histograms that it must store, only a fraction of these 500 should be specified at input time. The DISTRIBUTION DESCRIPTION CARD gives the distribution number N and 24 alphameric character description DSC. The last of these cards is to be blank to indicate that there are no more distributions inputs.

The index L indicates whether the range values and associated probabilities are to be read directly from the DISTRIBUTION QUANTITIES CARDS or calculated from the range minimum, mode and maximum values.

M is the number of values (to be in the range 1 to 11).

DQ: DISTRIBUTION QUANTITIES CARDS

For L = 1: DR(N,J), J=1,M

XR(N,J), J=1,M

For L = 2: DR(N,1),X1,X2,X3

FORMAT (8E10.3)

For L = 1, there are two sets of data DR(N,J) and XR(N,J). DR(N,1) is equal to the area for distributions describing the microbial burden on particular surfaces. For other distributions it is left blank and the program automatically supplies the mean of the distribution. DR(2) to DR(M) are the probabilities associated with the intervals XR(1) to XR(2) to XR(3), ..., XR(M-1) to XR(M). If M is larger than 8, two cards are required for each set of array values.

For L = 2, DR(1) has the same significance as for L=1. The range intervals of the distribution and the associate probabilities are determined by the program to describe, with a 5 value histogram, a triangular distribution with the range X1 to X3 and mode (probability peak) at X2.

The program automatically generates and indexes (index = KO) a zero value histogram.

KD: SUBTASK DESCRIPTION CARD

N, DSC, N1, N2

FORMAT (I5, 1X, 4A6, 2I5)

The input data for each of the (up to 20 subtasks per task) is headed with a SUBTASK DESCRIPTION CARD that gives the subtasks number N, the 24 alphameric character description DSC, and the indices of two prerequisite subtasks, N1 and N2. If N1 and N2 are blank, the subtask start time is taken to be the same as the task start time. If there is only one prerequisite subtask, it is to be indicated by N1 since, if N1 is blank, N2 is not checked. Both subtasks are, of course, to have been previously considered by the program and to have had their finish times computed.

KC: SUBTASK CHANGE CARD

K, IR(1), IR(2), IR(3), IR(4), IR(5), AR(1), AR(2), AR(3)

FORMAT (6I5, 3E10.3)

The SUBTASK CHANGE CARD is used to define any changes in area, environment, or retention factor values for a part.

Assembly or disassembly may cause the creation of new parts of surfaces, removing area from one change in the type of surfaces as when a certain amount of exterior area is covered and becomes an occluded or mated surface. The amount of area AR(1) is subtracted from part IR(3), surface IR(4), and added to part IR(1), surface IR(2).

If IR(5) is a positive integer, it is the new environment index assigned to part IR(1).

If AR(2) is a non-zero number, it is the new retention factor for fallout for part IR(1).

If AR(3) is a non-zero number, it is the new retention factor for contact for part IR(1).

K is the change index and is to be a positive integer except where it is to indicate there are not more changes and is given the value 0 or is left blank. The particular positive value assigned to K is arbitrary and can be used to designate certain types of changes or changes of a temporary nature for which some special index is useful in checking whether the reverse changes have been made.

KO: SUBTASK OPERATION CARD

IO, IKE AKT, AKQ

FORMAT (2I5, 2E10.3)

Each subtask consists of an arbitrary number of operations. These operations are partially described in the catalog of operations (see inputs OD and OQ described above) but have additional parameters which describe the resulting microbial burden changes for each particular performance of the operation. In addition, further parameters that depend on the part are specified on the PART EFFECT CARD to be described next.

IO is the operation index and indicates the particular operation involved. If IO is zero or blank, there are no more operations and the program is to proceed to a consideration of the next subtask. If IO is negative, a decontamination is indicated for which the input quantities have a different significance.

For decontamination (IO<0):

IKE is not significant and can be left blank.

AKT is the operation time.

For other operations (IO > 0):

IKE is the operation environment unless it is left blank in which case the operation environment is the same as for the previous operation. If IKE is blank for all operations in a subtask, the operation environments are all the same as the environment ITE given in card CC.

AKT is the operation time modifier and is multiplied by the distribution IOT(IO) (see card OQ) to determine the operation time.

AKQ is the modifier for themicrobial concentration associated with the source of contamination and is multiplied by the distribution IOQ(IO) (see card OQ) for the concentration distribution.

PE: PART EFFECT CARD

IP, LS, APD, APC, APS, APA(1), APA(2) or IP, LK, AR(1), AR(2), AR(3), AR(4)

FORMAT (2I5, 5E10.3)

Each part whose microbial burdens are affected by an operation is specified along with appropriate effect quantities by a PART EFFECT CARD.

IP is the index of the part affected. If IP is blank or zero, it indicates that all affected parts have been considered and the program is to proceed to a consideration of the next operation.

LS is the tool retention distribution for contact.

APD is the distance of part IP from the source of contamination (see Section IV for the effect of APD on fallout).

APC is the tool microbial concentration modifier which is multiplied by IOC(IO) (see card OQ) to determine the microbial concentration of the contamination tool (or other object) for determining microbial buildup by contact (see Section IV).

APS is the retention factor for the tool in determining contamination by contact. AI and APS are specified with the PART EFFECT inputs to allow modifying the quantities given with the operation IO in this particular application.

APA(1) is the contact area for the top surface, APA(2) the contact area for other exterior surfaces (see Section IV.). It is assumed that occluded and mated surfaces are not contacted.

The second set of quantities is for use in describing decontamination efforts following a SUBTASK OPERATION CARD for which IO is negative.

LK is the kill distribution index and describes the distribution of the fraction of organisms removed by a decontamination operation.

AR(J) is the modifier of the kill distribution for surface J.

ZD: ZONE DESCRIPTION CARD

IZ, DSC

FORMAT(I5, 1X, 4A6)

Definition of each thermal zone in terms of the fractions of surfaces that comprise it requires a ZONE DEFINITION CARD for the zone and one ZONE COMPOSITION CARD for each part involved. The zone is described in terms of a number, IZ, and a 24 alphameric character description DSC. If IZ is blank or zero it indicates that all zones have been defined and that the program is to proceed to the next CONTROL CARD.

ZC: ZONE COMPOSITION CARD

IP, FP(1), FP(2), FP(3), FP(4)

FORMAT (I5, 5X, 4F10.7)

A separate ZONE COMPOSITION CARD is used for each part IP that is wholly or partially included in zone IZ. The quantities FP(J) indicate the decimal fractions of surface J (top, exterior, mated, and interior) to be included in zone IZ.

VII. OUTPUTS

Typical program outputs are given in Listing 5 at the end of this report. It should be noted that a heading identifying the Run, Stage, and Task appears at the start of each set of Task inputs (Environments, Operations, Parts, and Distributions), each Subtask, and the Task Summary concluding each Task. Each Stage ends with a Stage Summary that has a heading that identifies the Run and Stage.

Burden Difference and Zonal Burden printouts must be specially called for by appropriate control cards. The Burden Difference heading identifies the Run and the two sets of Stage and Task between which the burden buildup is to be determined and printed. The Zonal Burden heading identifies the Run and Stage and the Task given is the Task that precedes the Zonal Burden determination.

Listings 5-a through 5-d simply list data that has been read as inputs. Table 5-e shows the results of Subtask and Operation level computations. It should be noted that preprequisites are not required and are not printed when not specified. Similarly, only the Environment/Area/Retention Factor changes specified are printed. There is no limit to the number of Operations in a Subtask or the number of Parts affected by an Operation.

Burden buildup by environmental fallout is not printed nor is the associated environment, but this buildup is always considered before considering the buildup by operational fallout or contact. The Environment index given for each operation is the operational environment. The start and finish time distributions are written on the record tape but only the mean time is printed in this output.

Under Parts affected by Operation, the Part, Surface, and the Distribution index are given along with the Source of contamination (Fallout or Contact) and the Burden on that surface following this particular instance of contamination.

Listing 5-f gives a typical Task Summary. The start and finish time distributions are given as well as the mean times. The Areas and Burdens are given for all parts that have been involved in any Operations or had IU(I) input as some positive value. Composite distributions along with mean and variance are given for the sums of Top Surfaces, Exterior Surfaces, Mated Surfaces, and Occluded Surfaces, and finally for the sum of all Surfaces. If L4 on the control card preceding the task is assigned some integer in the range $1 \leq L4 \leq 20$, the Task Summary will include the burden distribution on each surface of each part for which there is a prescribed area and which has been affected by any operation.

Listing 5-g gives a typical Stage Summary which is simply a convenient statement of the mean and variance of the distribution for Total Microbial Burden at the end of each Task of the Stage.

VIII. JOB CONTROL AND DECK ARRANGEMENT

A detailed description of job control cards and system options is given in two JPL reports*. This section describes a working arrangement of job control cards and program decks.

The job control cards are located at the start of Listings 1, 2, 5, and 6 (at the end of the report). These cards and the information furnished are described as follows:

$JOB, used at the beginning of each job

Columns	Contents
1 - 4	$JOB
5 - 6	blank
7 - 9	Programer's 3 initials
10	Comma
11 - 17	7-digit program identification
18	comma
19 - 25	[illegible] order number (format: XXXXX-X)
26	comma
27 - 31	5-digit requestor number
32	comma
33	Destination code
34 - 35	blank
36	System (I for IBJOB)

---

*The JPL Direct-Couple Operating System User's Reference Guide, JPL Report EPD-476, March 6, 1967; The FORTRAN IV User's Guide to the IBM 7090/94 IBJOB System, JPL Report EPD 478, March, 1967.

Columns	Contents
37	Time usage (C for code check, P for production)
38 - 39	blank
40	Estimates of running time in minutes and output in lines, separated by a comma but no blanks

$SETUP, used to indicate need for special record tape

Columns	Contents
1 - 6	$SETUP
8 -	Unit (UT5)
16 -	Tape number

$ASSIGN, used to assign the unit setup by the $SETUP card and any other units normally left unassigned by the DC-IBSYS.

Columns	Contents
1 - 7	$ASSIGN
16 -	SYSUT5

$ATEND, used to indicate the actions to be taken when a job segment is terminated.

Columns	Contents
1 - 6	$ATEND
16 -	Lower limit, upper limit, format code separated by commas, no blanks.

$IBJOB, specifies the options that describe how a job segment is to be processed.

Columns	Contents
1 - 6	$IBJOB
16 -	Options (GO, LOGIC, MAP, SOURCE)

$IBFTC, used immediately before a source deck.

Columns	Contents
1 - 6	$IBFTC
8 - 13	Deck name (6 or fewer alphameric characters)
16	Options (FULIST, REF, M94, XR7, DD, DECK)

$IBLDR, used immediately before a binary deck.

Columns	Contents
1 - 6	$IBLDR
8 - 13	Deck name (6 or fewer alphameric characters)

$DATA, used between the package of program decks and the data deck.

Columns	Contents
1 - 6	$DATA

EOF, the last card of the job

Columns	Contents
1	1, 7, 8 multipunched
2 - 4	EOF

There are 4 types of jobs associated with the Microbial Burden Prediction Model. They are

1. The Data Check Program (Table 2,3).
2. The Burden Prediction Program (Table 4,5).

3. The Data Recovery Program (Table 6,7).

4. The Special Write Program (Table 8,9).

Tables 2, 4, 6, and 8 show the arrangement of control cards, source decks, and, for the Data Check Program, data. Whenever the source programs are used and compiled, binary decks are generated which should be used in subsequent runs to speed the program. The arrangements for control cards, binary decks, and data are shown in Tables 3, 5, 7, and 9. Whenever a program consists of more than one deck, source and binary decks can be mixed providing all subprograms are included and that each source deck is preceded by the proper $IBFTC card and each binary deck by the proper $IBLDR card.

The Data Check Program (Tables 2,3 and Listing 1) is designed to read the input data from cards, write the data on tape 9 and in the output listing, and perform certain checks on the data such as determining that the data cards are in order, that all indices are within the required ranges, that prerequisite distributions and quantities have been defined, etc.

TABLE 2. THE MICROBIAL BURDEN DATA CHECK PROGRAM, SOURCE DECKS

```
$JOB
$SETUP CK1    TAPE #(9)
$ASSIGN       SYSCK1
$UNITS
$IBJOB        GO, LOGIC, MAP,SOURCE
$IBFTC MBDC1 FULIST,REF,M94,XR7,DD,DECK
(Deck MBDC1, Data Check Main Program)
$IBFTC MBDC2 FULIST, REF, M94, XR7, DD, DECK
(Deck MBDC2), Error Counting Subroutine)
$DATA
(Data Deck)
‡EOF
```

‡ Indicates 1,7,8 multipunched

TABLE 3. THE MICROBIAL BURDEN DATA CHECK PROGRAM, BINARY DECKS

```
$JOB
$SETUP    CKI           TAPE #(9)
$ASSIGN                 SYSCK1
$UNITS
$IBJOB                  GO,LOGIC,MAP,SOURCE
$IBLOR    MBDC1
(Binary Deck MBDC1)
$IBLDR    MBDC2
(Binary Deck MBDC2)
$DATA
(Data Deck)
‡EOF
```

‡ Indicates 1,7,8, multipunched

If there are no data errors, the Data Check Program writes a message to that effect and the data tape specified in the $SETUP card is ready to be used witn either the Microbial Burden Prediction Program or the Special Write Program.

The Microbial Burden Prediction Program (Tables 4,5 and Listing 2) reads inputs from the previously recorded tape (tape 9) and writes outputs on a second tape (tape 11) as well as in regular listing form.

TABLE 4. THE MICROBIAL BURDEN PREDICTION PROGRAM SOURCE DECKS

```
$JOB
$SETUP    CK1           TAPE #9
$SETUP    UT5           TAPE #11
$ASSIGN                 SYSCK1
$ASSIGN                 SYSUT5
$ASSIGN                 SYSUT6
$ASSIGN                 SYSUT8
$UNITS
$IBJOB                  GO,LOGIC,MAP,SOURCE
$IBFTC    MBP1          FULIST,REF,M94,XR7,DD,DECK
 (Deck MBP1, Predicition Program, Main Program)
$IBFTC    MBP2          FULIST,REF,M94,XR7,DD,DECK
 (Deck MBP2, Subroutine HCS)
$IBFTC    MBP3          FULIST,REF,M94,XR7,DD,DECK
 (Deck MBP3, Subroutine HES)
```

TABLE 4 (Continued)

```
$IBFTC   MBP4          FULIST,REF,M94,XR7,DD,DECK
 (Deck MBP4, Subroutine HMS)
$IBFTC   MBP5          FULIST,REF,M94,XR7,DD,DECK
 (Deck MBP5, Subroutine HNS)
$IBFTC   MBP6          FULIST,REF,M94,XR7,DD,DECK
 (Deck MBP6, Subroutine HWS)
$IBFTC   MBP7          FULIST,REF,M94,XR7,DD,DECK
 (Deck MBP7, Subroutine MAS)
$IBFTC   MBP8          FULIST,REF,M94,XR7,DD,DECK
 (Deck MBP8, Subroutine MAVS)
$IBFTC   MBP9          FULIST,REF,M94,XR7,DD,DECK
 (Deck MBP9, Subroutine MBRS)
$IBFTC   MBP10         FULIST,REF,M94,XR7,DD,DECK
 (Deck MBP10, Subroutine MBS)
$IBFTC   MBP11         FULIST,REF,M94,XR7,DD,DECK
 (Deck MBP11, Subroutine MVS)
$IBFTC   MBP12         FULIST,REF,M94,XR7,DD,DECK
 (Deck MBP12, Function ZF)
$DATA
‡EOF
```

Note that a $DATA card follows the last source deck but without being followed by a Data Deck. The Data is read from Tape 9. This is also true of the following programs.

TABLE 5. THE MICROBIAL BURDEN PREDICTION MODEL, BINARY DECKS

```
$JOB
$SETUP   CK1          TAPE #9
$SETUP   UT5          TAPE #11
$ASSIGN               SYSCK1
$ASSIGN               SYSUT5
$ASSIGN               SYSUT6
$ASSIGN               SYSUT8
$UNITS
$IBJOB                GO,LOGIC,MAP,SOURCE
$IBLPR   MBP1
 (Binary Deck MBP1)
$IBLDR   MBP2
 (Binary Deck MBP2)
$IBLDR   MBP3
 (Binary Deck MBP3)
$IBLDR   MBP4
 (Binary Deck MBP4)
$IBLDR   MBP5
 (Binary Deck MBP4)
$IBLDR   MBP6
 (Binary Deck MBP6
$IBLDR   MBP7
 (Binary Deck MBP7)
$IBLDR   MBP8
 (Binary Deck MBP8)
```

TABLE 5. (Continued)

```
$IBLDR    MBP9
 (Binary Deck MBP9)
$IBLDR    MBP10
 (Binary Deck MBP10)
$IBLDR    MBP11
 (Binary Deck MBP11)
$IBLDR    MBP12
 (Binary Deck MBP12)
$DATA
+EOF
```

The Data Recover Program (Tables 6, 7, and Listing 3) is not a fixed program but any modification of the basic program given in Listing 6. It is designed to read the data recorded on Tape 11 by The Microbial Burden Prediction Program. Instructions are to be added to this basic program for computing and writing any special outputs desired. These added instructions must not change the order in which data are read from Tape 11 because the basic program is designed to read this data in the same order as recorded by the Prediction Program. Any required subroutines should be added as indicated.

TABLE 6. THE MICROBIAL BURDEN DATA RECOVERY PROGRAM, SOURCE DECK

```
$JOB
$SETUP    UT5         TAPE #(11)
$ASSIGN               SYSUT5
$ASSIGN               SYSUT6
$ASSIGN               SYSUT8
$UNITS
$IBJOB                GO,LOGIC,MAP,SOURCE
$IBFTC    MBDR        FULIST,REF,M94,XR7,DD,DECK
 (Deck MBDR, Data Recovery Program)
$IBFTC    MBP2        FULIST,REF,M94,XR7,DD,DECK
 (Deck MBP2, Subroutine HCS, needed only if Histograms
  are to be added, subtracted, etc.)
$IBFTS    MBP6        FULIST,REF,M94,XR7,DD,DECK
 (Deck MBP6, Subroutine HWS, needed only if Histograms
  are to be written).
$DATA
+EOF
```

TABLE 7. THE MICROBIAL BURDEN DATA RECOVERY PROGRAM, BINARY DECK

```
$JOB
$SETUP    UT5         TAPE #(11)
$ASSIGN               SYSUT5
$ASSIGN               SYSUT6
$ASSIGN               SYSUT8
$UNITS
$IBJOB                GO,LOGIC,MAP,SOURCE
$IBLDR    MBDR
 (Binary Deck MBDR)
 (Any required subroutine decks)
$DATA
+EOF
+
```

The Special Write Program (Tables 8,9 and Listing 4) is designed to read the input tape (Tape 9) and list those cards that describe the Run, Stage, Task, Subtask, Operation, and Part Affected. Each card type is indented according to its level so that the result is a convenient outline of the run.

TABLE 8. THE MICROBIAL BURDEN SPECIAL WRITE PROGRAM, SOURCE DECK

```
$JOB
$SETUP    CK1         TAPE #(9)
$ASSIGN               SYSCK1
$UNITS
$IBJOB                GO,LOGIC,MAP,SOURCE
$IBFTC    MBSW        FULIST,REF,M94,XR7,DD,DECK
 (Deck MBSW, Special Write Program)
$DATA
+EOF
+
```

TABLE 9. THE MICROBIAL BURDEN SPECIAL WRITE PROGRAM, BINARY DECK

```
$JOB
$SETUP    CK1          TAPE #(9)
$ASSIGN                SYSCK1
$UNITS
$IBJOB                 GO,LOGIC,MAP,SOURCE
$IBLDR    MBSW
 (Binary Deck MBSW)
$DATA
+EOF
+
```

A special use of the Microbial Burden Prediction Program, that of Restart, deserves a special description. If an input data set is to be rerun with modifications that are not near the beginning of the set and if the output tape (Tape 11) has been saved, running the program from the beginning can be avoided by using the restart capability of the program.

To use the Restart capability, a Restart CONTROL CARD (i.e., KK = 6, IK = 0, I1,I2 = the last stage and task for which the data has not been changed) is put in front of the CONTROL CARD initiating the first task that is changed. This data is then recorded on Tape 9 by using the Data Check Program. This Tape 9 and the Tape 11 with the saved output data are then used, in the standard way, with the Burden Prediction Program which reads Tape 11 until the specified last stage and task are finished at which time input data is read from Tape 9 and the computation proceeds in the customary way. The output listing, in this case, will commence at the Task where the data has been changed and, when the run is complete, Tape 11 will hold outputs from the start of the original Run but, after the specified restart point, will have outputs based on the new input data.

```
$JOB  GMM,584800,06904-0,78563,A   IC   8,19000          GMM  BOX 84              84
$SETUP CK1      4451
$ASSIGN         SYSCK1
$UNITS
$IBJOB          GO,LOGIC,MAP,SOURCE
$IBFTC MBDC1    FULIST,REF,M94,XR7,DD,DECK
C
C     MICROBIOLOGICAL BURDEN PREDICTION PROGRAM DATA CHECK PROGRAM
C     THIS PROGRAM WRITES A DATA TAPE AS IT CHECKS THE DATA
C
C     ERROR TYPES-
C        1 INDEX OUT OF RANGE               5 INDEX USED PREVIOUSLY
C        2 PREREQUISITE MISSING             6 DATA OUT OF RANGE
C        3 NECESSARY DATA MISSING           7 REMAINING AREA NEGATIVE
C        4 TOO MANY CARDS IN SET            8 AFFECTED PART AREA ZERO
C                                           9 CONTACT AREA GT SURFACE AREA
C
C     CARD TYPES-
C        CC CONTROL CARD                    PD PART DESCRIPTION
C        RD RUN DESCRIPTION                 PQ PART QUANTITIES
C        SD STAGE DESCRIPTION               DD DISTRIBUTION DESCRIPTION
C        TD TASK DESCRIPTION                DQ DISTRIBUTION QUANTITIES
C        EM ENVIRONMENTAL MODIFIERS         KD SUBTASK DESCRIPTION
C        ED ENVIRONMENT DESCRIPTION         KC SUBTASK CHANGE
C        EQ ENVIRONMENT QUANTITIES          KO SUBTASK OPERATION
C        OD OPERATION DESCRIPTION           PE PART EFFECTS
C        OQ OPERATION QUANTITIES            ZD ZONE DESCRIPTION
C                                           ZC ZONE COMPOSITION
C
      DIMENSION RUN(7),STG(7),TSK(7),DSC(6),AES(4),IEC(10),IET(10),
     .AEC(10),AET(10),AED(10),AEF(10,4),IOT(20),IOQ(20),IOC(20),
     .IAB(120,6),AAG(120),AAS(120),AAT(120),DR(500,11),XR(500,11),
     .KKT(20),IR(5),AR(4),APA(2),LH(20)
      COMMON NL,CT(19),KR,KS,KT,JS,JO,KA(20,5),KL(9)
      DATA (AAG(I),I=1,19)/2HCC,2HRD,2HSD,2HTD,2HEM,2HED,2HEQ,
     .2HOD,2HOQ,2HPD,2HPQ,2HDD,2HDQ,2HKD,2HKC,2HKO,2HPE,2HZD,2HZC/
    1 FORMAT(34H1MICROBIAL BURDEN PREDICTION MODEL/12H DATA CHECK-//)
    2 FORMAT(8I5)
    3 FORMAT(//35H RESTART CALLED FOR FOLLOWING STAGE,I3,6H, TASK,I3//)
    4 FORMAT(I5,1X,7A6,2X,4I5,30X,4I5)
    5 FORMAT(I5,1X,4A6,2I5)
    6 FORMAT(I5,5X,4F10.7)
    7 FORMAT(8E10.3)
    8 FORMAT(2I5,9E10.3)
    9 FORMAT(6I5,3E10.3,10X,I5,E10.3,I5,E10.3)
   11 FORMAT(12H ERROR LIST-//30H    N   KR   KS   KT   JS   JO/)
   12 FORMAT(6I5)
   13 FORMAT(/19H NO ERRORS THIS RUN)
   14 FORMAT(78X,17HKR   KS   KT   NL/)
   15 FORMAT(27H NUMBERS OF ERRORS BY TYPE-//9I5)
      REWIND 9
      ND=0
      NL=0
      KR=0
      KS=0
      KT=0
      DO 50 I=1,19
   50 CT(I)=AAG(I)
      DO 55 J=1,9
   55 KL(J)=0
```

Listing 1. The Microbial Burden Data Check Program, Main Program

```
      DO 60 I=1,120
      DO 60 J=1,6
   60 IAB(I,J)=0
      DO 70 I=1,500
   70 DR(I,1)=0.
      WRITE(6,1)
C
C     CONTROL CARD
  100 READ (5,2)KK,IK,I1,I2,I3,I4,I5,I6
      WRITE(6,2)KK,IK,I1,I2,I3,I4,I5,I6
      WRITE(9)  KK,IK,I1,I2,I3,I4,I5,I6
      JS=0
      JO=0
      IF(KK)102,300,104
  102 CALL ECS(1,1)
      GO TO 300
  104 IF(KK.GT.6)GO TO 102
      IF(KK.LT.6)GO TO 108
      WRITE(6,3)I1,I2
      KS=I1
      KT=I2
      DO 106 I=1,20
  106 LH(I)=1
      GO TO 100
  108 GO TO (109,109,109,140,150),KK
  109 IF(IK.LE.-1)CALL ECS(1,1)
      IF(IK.GT.5)CALL ECS(1,1)
      GO TO (110,120,130),KK
C
C     RUN DESCRIPTION
  110 READ (5,4)KR,(RUN(J),J=1,7)
      WRITE(6,4)KR,(RUN(J),J=1,7)
      WRITE (9) KR,(RUN(J),J=1,7)
      DO 112 I=1,20
  112 LH(I)=0
C
C     STAGE DESCRIPTION
  120 READ (5,4)KS,(STG(J),J=1,7)
      WRITE(6,4)KS,(STG(J),J=1,7)
      WRITE (9) KS,(STG(J),J=1,7)
      IF(KS.LE.0)GO TO 125
      IF(KS.LE.20)GO TO 130
  125 CALL ECS(2,1)
      KS=1
C
C     TASK DESCRIPTION
  130 READ (5,4)KT,(TSK(J),J=1,7),L1,L2,L3,L4
      WRITE(6,4)KT,(TSK(J),J=1,7),L1,L2,L3,L4,KR,KS,KT,NL
      WRITE (9) KT,(TSK(J),J=1,7),L1,L2,L3,L4
      IF(KS.LE.0)CALL ECS(4,3)
      DO 132 I=1,20
  132 KKT(I)=0
      IF(KT.LE.0)GO TO 135
      IF(KT.LE.100)GO TO 170
  135 CALL ECS(3,1)
      KT=1
      GO TO 170
  140 IF(I1.LE.0)CALL ECS(1,1)
      IF(I1.GT.20)CALL ECS(1,1)
      IF(I2.LE.0)CALL ECS(1,1)
      IF(I2.GT.100)CALL ECS(1,1)
```

Listing 1. The Microbial Burden Data Check Progam, Main Pr

```
      GO TO 100
 150  IF(IK.EQ.1)GO TO 100
 152  READ (5,5)IZ,(DSC(J),J=1,4)
      WRITE(6,5)IZ,(DSC(J),J=1,4)
      WRITE (9) IZ,(DSC(J),J=1,4)
      IF(IZ)158,100,160
 158  CALL ECS(18,1)
 160  READ (5,6)IP,(AR(J),J=1,4)
      WRITE(6,6)IP,(AR(J),J=1,4)
      WRITE (9) IP,(AR(J),J=1,4)
      IF(IP)162,152,160
 162  CALL ECS(19,1)
      GO TO 160
 170  IF(L1)172,190,174
 172  CALL ECS(4,1)
      L1=-L1
 174  IF(L1.GT.20)GO TO 182
      IF(LH(L1).EQ.0)CALL ECS(4,2)
 176  IF(L2)178,190,180
 178  CALL ECS(4,1)
 180  IF(L2.GT.20)GO TO 182
      IF(LH(L2).EQ.0)CALL ECS(4,2)
      GO TO 190
 182  CALL ECS(4,1)
 190  IF(L3)194,196,192
 192  IF(L3.GT.20)GO TO 194
      LH(L3)=1
      GO TO 196
 194  CALL ECS(4,1)
 196  IF(L4.LE.-1)CALL ECS(4,1)
      IF(L4.GT.20)CALL ECS(4,1)
      IF(I4.GT.0)ND=I4
      GO TO (210,220,230,240,250),IK
C
C     ENVIRONMENTS INPUTS-
 210  READ (5,7)(AES(J),J=1,4)
      WRITE(6,7)(AES(J),J=1,4)
      WRITE (9) (AES(J),J=1,4)
      DO 216 I=1,10
      READ (5,5)N,(DSC(J),J=1,4)
      WRITE(6,5)N,(DSC(J),J=1,4)
      WRITE (9) N,(DSC(J),J=1,4)
      IF(N)213,220,212
 212  IF(N.LE.10)GO TO 214
 213  CALL ECS(6,1)
      N=10
 214  READ (5,8)IEC(N),IET(N),AEC(N),AET(N),AED(N),(AEF(N,J),J=1,4)
      WRITE(6,8)IEC(N),IET(N),AEC(N),AET(N),AED(N),(AEF(N,J),J=1,4)
 216  WRITE (9) IEC(N),IET(N),AEC(N),AET(N),AED(N),(AEF(N,J),J=1,4)
      READ (5,5)N
      WRITE(6,5)N
      WRITE (9) N
      IF(N.GT.0)CALL ECS(6,4)
C
C     OPERATIONS INPUTS-
 220  DO 226 K=1,20
      READ (5,5)N,(DSC(J),J=1,4)
      WRITE(6,5)N,(DSC(J),J=1,4)
      WRITE (9) N,(DSC(J),J=1,4)
      IF(N)223,230,222
 222  IF(N.LE.20)GO TO 224
```

Listing 1. The Microbial Burden Data Check Program, Main Program

```
 223  CALL ECS(8,1)
      N=20
 224  READ (5,9)IOT(N),IOO(N),IOC(N)
      WRITE(6,9)IOT(N),IOO(N),IOC(N)
 226  WRITE (9) IOT(N),IOO(N),IOC(N)
      READ (5,5)N
      WRITE(6,5)N
      WRITE (9) N
      IF(N.GT.0)CALL ECS(8,4)
C
C     PARTS INPUTS-
 230  DO 236 I=1,120
      READ (5,4)N,(DSC(J),J=1,6)
      WRITE(6,4)N,(DSC(J),J=1,6)
      WRITE (9) N,(DSC(J),J=1,6)
      IF(N)233,240,232
 232  IF(N.LE.120)GO TO 234
 233  CALL ECS(10,1)
      N=120
 234  READ (5,9)(IAB(N,J),J=1,6),AAG(N),AAS(N)
      WRITE(6,9)(IAB(N,J),J=1,6),AAG(N),AAS(N)
 236  WRITE (9) (IAB(N,J),J=1,6),AAG(N),AAS(N)
      WRITE(6,4)N
      WRITE (9) N
      IF(N.GT.0)CALL ECS(10,4)
C
C     DISTRIBUTIONS INPUTS-
 240  DO 248 I=1,500
      READ (5,5)N,(DSC(J),J=1,4),K,M
      WRITE(6,5)N,(DSC(J),J=1,4),K,M
      WRITE (9) N,(DSC(J),J=1,4),K,M
      IF(N)243,249,242
 242  IF(N.LE.500)GO TO 244
 243  CALL ECS (12,1)
      N=500
 244  IF(K.LE.0)CALL ECS(12,1)
      IF(K.GT.2)CALL ECS(12,1)
      IF(K.EQ.1)GO TO 245
      READ (5,7)DR(N,1),X1,X2,X3
      WRITE(6,7)DR(N,1),X1,X2,X3
      WRITE(9)  DR(N,1),X1,X2,X3
      V=.25-U
      DR(N,2)=U
      DR(N,5)=V
      XR(N,1)=X1
      XR(N,3)=X2
      XR(N,4)=.5*(X2+X3)
      NX(N)=5
      GO TO 248
 245  IF(M.LE.0)GO TO 246
      IF(M.LE.11)GO TO 247
 246  CALL ECS(12,1)
      M=11
 247  READ (5,7)(DR(N,J),J=1,M)
      WRITE(6,7)(DR(N,J),J=1,M)
      WRITE (9) (DR(N,J),J=1,M)
      READ (5,7)(XR(N,J),J=1,M)
      WRITE(6,7)(XR(N,J),J=1,M)
      WRITE (9) (XR(N,J),J=1,M)
 248  IF(NO.LT.N)NO=N
      READ (5,5)N
```

Listing 1. The Microbial Burden Data Check Program, Main Program

```
      WRITE(6,5)N
      WRITE (9) N
      IF(N.GT.0)CALL ECS(12,4)
  249 ND=ND+1
  250 DO 296 I=1,20
C
C     SUBTASK DESCRIPTION
      READ (5,5)N,(DSC(J),J=1,4),N1,N2
      WRITE(6,5)N,(DSC(J),J=1,4),N1,N2
      WRITE (9) N,(DSC(J),J=1,4),N1,N2
      JS=N
      IF(N)253,100,252
  252 IF(N.LE.20)GO TO 254
  253 CALL ECS(14,1)
      N=20
  254 IF(KKT(N).EQ.1)CALL ECS(14,5)
      IF(N1)262,256,258
  256 IF(N2)262,264,262
  258 IF(N1.GT.20)GO TO 262
      IF(KKT(N1).EQ.0)CALL ECS(14,2)
      IF(N2)262,264,260
  260 IF(N2.GT.20)GO TO 262
      IF(KKT(N2).EQ.0)CALL ECS(14,2)
      GO TO 264
  262 CALL ECS(14,1)
  264 KKT(N)=1
C
C     SUBTASK LEVEL CHANGE
  270 READ (5,9)K,(IR(J),J=1,5),(AR(J),J=1,3)
      WRITE (9) K,(IR(J),J=1,5),(AR(J),J=1,3)
      IF(K.LE.0) GO TO 280
      IF(AR(1).GT.0.)GO TO 272
      WRITE(6,9)K,(IR(J),J=1,5),(AR(J),J=1,3)
      IF(IR(1).LE.0)CALL ECS(15,1)
      GO TO 270
  272 I1=IR(1)
      I2=IR(2)
      IF(I1.LE.0)GO TO 276
      IF(I1.GT.120)GO TO 276
      IF(I2.LE.0)GO TO 276
      IF(I2.GT.4)GO TO 276
      IB=IAB(I1,I2)
      I3=IR(3)
      I4=IR(4)
      IF(I3.LE.0)GO TO 276
      IF(I3.GT.120)GO TO 276
      IF(I4.LE.0)GO TO 276
      IF(I4.GT.4)GO TO 276
      IA=IAB(I3,I4)
      IF(IA.GT.0)GO TO 273
      WRITE(6,9)K,(IR(J),J=1,5),(AR(J),J=1,3)
      CALL ECS(15,3)
      GO TO 270
  273 IF(IB.GT.0)GO TO 274
      ND=ND+1
      IB=ND
      IAB(I1,I2)=IB
      DR(IB,1)=0.
  274 DR(IA,1)=DR(IA,1)-AR(1)
      DR(IB,1)=DR(IB,1)+AR(1)
      WRITE(6,9)K,(IR(J),J=1,5),(AR(J),J=1,3),IA,DR(IA,1),IB,DR(IB,1)
```

Listing 1. The Microbial Burden Data Check Program. Main Program

```
      IF(DR(IA,1).LT.0.)CALL ECS(15,7)
      IF(DR(IA,1).LE.0.)IAB(I3,I4)=0
      GO TO 270
  276 WRITE(6,9)K,(IR(J),J=1,5),(AR(J),J=1,3)
      CALL ECS(15,1)
      GO TO 270
C
C     SUBTASK OPERATION
  280 WRITE(6,9)K,(IR(J),J=1,5),(AR(J),J=1,3)
      READ (5,8)IO IKE,AKT,AKQ
      WRITE(6,8)IO,IKE,AKT,AKQ
      WRITE (9) IO,IKE,AKT,AKQ
      JO=IO
      IF(IO)284,296,288
  282 CALL ECS(17,1)
C
C     PART EFFECTS FOR DECONTAMINATION
  284 READ (5,8)IP,LK,(AR(J),J=1,4)
      WRITE(6,8)IP,LK,(AR(J),J=1,4)
      WRITE (9) IP,LK,(AR(J),J=1,4)
      IF(IP)282,280,286
  286 IF(IP.GT.120)GO TO 282
      M=NX(LK)
      X=XR(LK,M)
      DO 287 J=1,4
  287 IF(AR(J)*X.GT.1.)CALL ECS(17,6)
      GO TO 284
  288 IF(IO.GT.20)CALL ECS(16,1)
      IF(IKE.LE.-1)CALL ECS(16,1)
      IF(IKE.GT.10)CALL ECS(16,1)
C
C     PART EFFECTS
  290 READ (5,8)IP,LS,APD,APC,APS,APA(1),APA(2)
      WRITE (9) IP,LS,APD,APC,APS,APA(1),APA(2)
      IF(IP.GT.120)GO TO 294
      IF(IP.LE.0)GO TO 294
      AA=0.
      DO 292 J=1,4
      AR(J)=0.
      IB=IAB(IP,J)
      IF(IB.LE.0)GO TO 292
      AR(J)=DR(IB,1)
      AA=AA+AR(J)
  292 CONTINUE
      WRITE(6,8)IP,LS,APD,APC,APS,APA(1),APA(2),(AR(J),J=1,4)
      IF(AA.LE.0.)CALL ECS(17,8)
      DO 293 J=1,2
  293 IF(APA(J).GT.AR(J))CALL ECS(17,9)
  294 WRITE(6,8)IP,LS,APD,APC,APS,APA(1),APA(2)
      IF(IP)295,280,295
  295 CALL ECS(17,1)
      GO TO 290
  296 CONTINUE
      READ (5,5)N
      WRITE(6,5)N
      WRITE (9) N
      IF(N.GT.0)CALL ECS(14,4)
      GO TO 100
  300 WRITE(6,1)
      IF(NL.EQ.0)GO TO 310
      WRITE(6,11)
```

Listing 1. The Microbial Burden Data Check Program, Main Program

```
      M=20
      IF(NL.LT.20)M=NL
      WRITE(6,12)(I,(KA(I,J),J=1,5),I=1,M)
      WRITE(6,15)(KL(J),J=1,9)
      CALL EXIT
  310 WRITE(6,13)
      CALL EXIT
      GO TO 100
      END
```

Listing 1. The Microbial Burden Data Check Program, Main Program

```
$IBFTC MBDC2    FULIST,REF,M94,XR7,DD,DECK
      SUBROUTINE ECS(LC,LE)
C
C     ERROR COUNT SUBROUTINE
C     LC IS THE DATA CARD TYPE, LE THE ERROR TYPE
      COMMON NL,CT(19),KR,KS,KT,JS,JO,KA(20,5),KL(9)
    1 FORMAT(//12H ** ** **ERROR,I3,6H, TYPE,I3,1X,A2,6H* ** ** *//)
      NL=NL+1
      WRITE(6,1)NL,LE,CT(LC)
      IF(NL.GT.20)GO TO 10
      KA(NL,1)=KR
      KA(NL,2)=KS
      KA(NL,3)=KT
      KA(NL,4)=JS
      KA(NL,5)=JO
   10 KL(LE)=KL(LE)+1
      RETURN
      END
```

Listing 1. The Microbial Burden Data Check Program, Error Count Subroutine

```
$DATA
    1    1                                  1
    1 VOYAGER TEST CASE
    1 A   VERNIER MODULE ASSEMBLY + TEST
    1 1   SUBSYSTEM POSITIONING                                  1
    1.         1.        10.        10.
    1 LAMINAR FLOW TENT
    1     4    0.7      100.        2.3        4.2        1.4
    2 CLASS 100K CLEAN ROOM
    1     4    3.       100.         .46      4.8         1.6
    3 TEST FACILITIES
    1     4    7.       100.         .46       4.8        1.6
      END OF ENVIRONMENTS
    1 MOVE ASSEMBLY MANUALLY
    1     3    3
    2 POSITION OVERHEAD CRANE
    2     3
    3 ATTACH CRANE HOOKS
    2     3    3
    4 HOIST WITH CRANE
    2     3
    5 MOVE WITH CRANE
    3     3
    6 LOWER WITH CRANE
    2     3
    7 DETACH CRANE HOOKS
    2     3
    8 VISUAL INSPECTION
    3     2
    9 LIFT WITH MSPF
    2     2    2
   10 PLACE IN HANDLING CONTANR
    1     3    2
   11 MOVE IN HANDLING CONTANR
    1     2    1
   12 CONNECT CABLES,HOSES,ETC
    3     3    3
   13 DISCONNECT CABLES,ETC.
    1     3    3
   14 FUNCTIONAL TESTING
    3     1
   15 INSERT SCREW,BOLT,ETC.
    2     2    1
   16 TIGHTEN SCREW,BOLT,ETC.
    1     2    1
   17 LOOSEN SCREW,BOLT,ETC.
    1     2    1
   18 REMOVE SCREW,BOLT,ETC.
    2     2    1
   19 REMOVE FRM HNDLNG CONTNR
    1     3    2
      END OF OPERATIONS
    1 VERNIER STRUCTURE
    7     8    0    0    0    5    .4        .4
    2 LANDER LEGS
    9    10                   5    .4        .4
    3 DEORBIT STRUCTURE
   11    12                   5    .4        .4
    5 AEROSHELL STRUCTURE
   13    14                   5    .6        .9
```

Listing 1. The Microbial Burden Data Check Program, Start of Input Data

```
$JOB  GMM,5848000,06904-0,78563,A   IC   55,300000        GMM  BOX 84               84
$SETUP CK1       4451
$SETUP UT5       5740
$ASSIGN          SYSCK1
$ASSIGN          SYSUT5
$ASSIGN          SYSUT6
$ASSIGN          SYSUT8
$UNITS
$IBJOB           GO,LOGIC,MAP,SOURCE
$IBFTC MBP1      FULIST,REF,M94,XR7,DD,DECK
C
C      MICROBIAL BURDEN PREDICTION PROGRAM
C      THIS PROGRAM READS FROM TAPE 9
       COMMON/BASE/K1,K2,K3,K4,K5,KK,IK,
      ,KR,RUN(7),KS,STG(7),KT,TSK(7),DSC(6),
      ,NE,AES(4),IEC(10),IET(10),AEC(10),AET(10),AED(10),AEF(10,4),
      ,NO,IOT(20),IOQ(20),IOC(20),
      ,NP,DAC(120,2),IU(120),IAB(120,6),AAG(120),AAS(120),AAT(120),
      ,ND,NX(551),DR(551,11),XR(551,11),KO,
      ,NS,KKT(20),IR(5),AR(4),APA(2),
      ,NT,JT(100),XMT(100),XVT(100),L1,L2,L3,L4,KTS,ITE
    1  FORMAT(8I5)
    2  FORMAT(I5,1X,7A6,2X,4I5)
    3  FORMAT(34H1MICROBIAL BURDEN PREDICTION MODEL/4H RUN,I5,2H, ,7A6)
    4  FORMAT(6H STAGE,I3,2H, ,7A6/15H STAGE SUMMARY-//
      ,6X,27HTASK  MEAN BURDEN  VARIANCE/)
    5  FORMAT(I10,1X,1P2E11.3)
    6  FORMAT(29H CUMULATIVE STATISTICS, STAGE,I3,6H, TASK,I3,
      ,9H TO STAGE,I3,6H, TASK,I3//52X,4HFROM,7X,2HTO,10X,5HPROB./)
    7  FORMAT(43H MICROBIAL BURDEN BY ZONES (FOLLOWING STAGE,I3,6H, TASK,
      ,I3,2H)-//6X,4HZONE,24X,4HAREA,13X,4HFROM,8X,2HTO,10X,5HPROB./)
    8  FORMAT(I5,5X,4F10.7)
    9  FORMAT(I5,1X,4A6,1PE11.3)
   10  FORMAT(I5,1X,2A6,12X,1PE11.3)
       K1=1
       K2=2
       K3=3
       K4=4
       K5=5
       NE=0
       NO=0
       NP=0
       ND=0
       NS=0
       NT=0
       DO 80 I=1,120
       IU(I)=0
       AAG(I)=0.
       AAS(I)=0.
       AAT(I)=0.
       DO 80 J=1,5
   80  IAB(I,J)=0
       REWIND 9
       REWIND 11
  100  READ  (9) KK,IK,I1,I2,I3,I4,I5,I6
C      KK=1 INDICATES A NEW RUN, STAGE, AND TASK
C      KK=2 INDICATES A NEW STAGE AND TASK
C      KK=3 INDICATES A NEW TASK
C      KK=4 CALLS FOR A BURDEN DIFFERENCE DETERMINATION
```

Listing 2. The Microbial Burden Prediction Program, Main Program (MBP1)

```
C      KK=5 CALLS FOR ZONE BURDEN WRITEOUT
C         IK=0 CALLS FOR ZONE DEFINITION DATA INPUTS
C         IK=1 INDICATES ZONES CORRESPOND TO PARTS
C      KK=6 CALLS FOR RESTART AT CONTROL CARD AFTER STAGE I1, TASK I2
       IF(KK.LE.3)WRITE(11)KK,IK,I1,I2,I3,I4,I5,I6
       IF(KK.NE.6)GO TO 102
       CALL MBRS(I1,I2,I3,I4,I5,I6)
       GO TO 100
  102  IF(KK.EQ.3)GO TO 130
       IF(NT.EQ.0)GO TO 108
C
C      WRITE STAGE SUMMARY FOR PRECEEDING STAGE
       WRITE(6,3)KR,(RUN(J),J=1,7)
       WRITE(6,4)KS,(STG(J),J=1,7)
       DO 104 I=1,NT
  104  WRITE(6,5)JT(I),XMT(I),XVT(I)
       NT=0
  108  IF(KK.LE.0)CAL' EXIT
       GO TO (110,120,130,140,150),KK
C
C      READ RUN NUMBER AND DESCRIPTION
  110  READ  (9) KR,(RUN(J),J=1,7)
C
C      READ STAGE NUMBER AND DESCRIPTION
  120  READ  (9) KS,(STG(J),J=1,7)
C
C      READ TASK NUMBER AND DESCRIPTION
  130  READ  (9) KT,(TSK(J),J=1,7),L1,L2,L3,L4
C      L1 AND L2 ARE PREREQUISITE TIME DISTRIBUTIONS
C      L3 IS THE FINISH TIME DISTRIBUTION IF NEEDED
C         AS PREREQUISITE FOR ANOTHER TASK
C      L4 IS THE INDEX FOR SAVING THE TOTAL MICROBIAL BURDEN
C         IF NEEDED FOR USE IN DETERMINING A BURDEN DIFFERENCE
C         AND FOR PRINTING EACH SURFACE BURDEN IN THE TASK SUMMARY
C
C      DETERMINE TASK START TIME
       KTS=0
       IF(L1.LE.0)GO TO 134
       L1=L1+530
       IF(L2.GT.0)GO TO 132
       KTS=L1
       GO TO 134
  132  L2=L2+530
       KTS=551
       CALL HCS(L1,L2,KTS,5)
  134  IF(I1.GT.0)NE=I1
       IF(I2.GT.0)NO=I2
       IF(I3.GT.0)NP=I3
       IF(I4.GT.0)ND=I4
       IF(I5.GT.0)NS=I5
       IF(I6.GT.0)ITE=I6
       WRITE(11)KR,KS,KT,(RUN(J),STG(J),TSK(J),J=1,7),L1,L2,L3,L4,KTS,ITE
       NT=NT+1
       CALL MBS
       GO TO 100
C
C      DETERMINE AND WRITE BURDEN DIFFERENCE
  140  IF(I3*I6.LE.0)GO TO 100
       WRITE(6,3)KR,(RUN(J),J=1,7)
       WRITE(6,6)I1,I2,I4,I5
       L1=I3+530
```

Listing 2. The Microbial Burden Prediction Program, Main Program (MBP_1)

```
      L2=I6+530
      L3=551
      CALL HCS(L2,L1,L3,2)
      CALL HWS(L3)
      CALL MAVS(L3,XM,XV)
      GO TO 100
C
C     DETERMINE AND WRITE ZONE BURDEN DISTRIBUTIONS
 150  WRITE(6,3)KR,(RUN(J),J=1,7)
      WRITE(6,7)KS,KT
      IF(IK.EQ.1)GO TO 180
C
C     USE ZONE DEFINITION INPUTS
 155  READ  (9) IZ,(DSC(J),J=1,4)
      IF(IZ.LE.0)GO TO 100
      CALL HES(510,KO)
 160  READ  (9) IP,(AR(J),J=1,4)
C     IP IS THE PART
C     AR(J) THE FRACTION OF SURFACE J OF PART IP BELONGING TO ZONE IZ
      IF(IP.LE.0)GO TO 170
      DO 165 J=1,4
      IF(AR(J).EQ.0.)GO TO 165
      IF(IAB(IP,J).EQ.0)GO TO 165
      F=AR(J)
      IB=IAB(IP,J)
      CALL HMS(IB,F,509)
      CALL HCS(510,509,510,1)
 165  CONTINUE
      GO TO 160
 170  WRITE(6,9)IZ,(DSC(J),J=1,4),DR(510,1)
      CALL HWS(510)
      GO TO 155
C
C     USE PARTS AS ZONES
 180  DO 190 I=1,NP
      IF(IU(I).EQ.0)GO TO 190
      CALL HES(510,KO)
      DO 185 J=1,4
      IB=IAB(I,J)
      IF(IB.GT.0)CALL HCS(510,IB,510,1)
 185  CONTINUE
      WRITE(6,10)I,DAC(I,1),DAC(I,2),DR(510,1)
      CALL HWS(510)
 190  CONTINUE
      GO TO 100
      END
```

Listing 2. The Microbial Burden Prediction Program, Main Program (MBP1)

```
$IBFTC MBP2     FULIST,REF,M94,XR7,DD,DECK
      SUBROUTINE HCS(IA,IB,IC,K)
C
C     HISTOGRAM COMBINING SUBROUTINE
      DIMENSION CA(11),CB(11),CR(21),ZR(21),QR(11,11),NQ(11)
      COMMON/BASE/K1,K2,K3,K4,K5,KK,IK,
     .KR,RUN(7),KS,STG(7),KT,TSK(7),DSC(6),
     .NE,AES(4),IEC(10),IET(10),AEC(10),AET(10),AED(10),AEF(10,4),
     .NO,IOT(20),IOQ(20),IOC(20),
     .NP,DAC(120,2),IU(120),IAB(120,6),AAG(120),AAS(120),AAT(120),
     .ND,NX(551),DR(551,11),XR(551,11),KO,
     .NS,KKT(20),IR(5),AR(4),APA(2),
     .NT,JT(100),XMT(100),XVT(100),L1,L2,L3,L4,KTS,ITE
      NA=NX(IA)
      NB=NX(IB)
      A=DR(IA,1)
      B=DR(IB,1)
      X=XR(IA,1)
      Y=XR(IB,1)
      CR(1)=ZF(A,B,K)
      DR(IC,1)=CR(1)
      ZR(1)=ZF(X,Y,K)
      IF(NA.GT.1)GO TO 50
      IF(X.NE.0.)GO TO 30
      IF((K-3)*(K-4).EQ.0)GO TO 35
   30 IF(NB.GT.1)GO TO 40
   35 XR(IC,1)=ZR(1)
      NX(IC)=1
      RETURN
   40 NC=NB
      DO 45 J=2,NC
      Y=XR(IB,J)
      CR(J)=DR(IB,J)
   45 ZR(J)=ZF(X,Y,K)
      GO TO 60
   50 IF(NB.GT.1)GO TO 90
      NC=NA
      DO 55 J=2,NC
      X=XR(IA,J)
      CR(J)=DR(IA,J)
   55 ZR(J)=ZF(X,Y,K)
   60 NX(IC)=NC
      IF(ZR(NC).GT.ZR(1))GO TO 70
      XR(IC,1)=ZR(NC)
      JC=NC
      DO 65 J=2,NC
      DR(IC,J)=CR(JC)
      JC=JC-1
   65 XR(IC,J)=ZR(JC)
      GO TO 80
   70 XR(IC,1)=ZR(1)
      DO 75 J=2,NC
      DR(IC,J)=CR(J)
   75 XR(IC,J)=ZR(J)
   80 IF(K.LE.4)RETURN
C     ELIMINATION OF DUPLICATE X VALUES FOR K=5
      CR(1)=0.
      CR(2)=0.
      JC=1
```

Listing 2. The Microbial Burden Prediction Program, Subroutine HCS (MBP2)

```
      ZR(1)=XR(IC,1)
      DO 85 J=2,NC
      CR(J)=CR(J)+DR(IC,J)
      IF(XR(IC,J).LE.XR(IC,J-1))GO TO 85
      JC=JC+1
      CR(JC)=CR(J)
      ZR(JC)=XR(IC,J)
 85   CR(J+1)=CR(J)
      NC=JC
      GO TO 300
 90   IF(K.EQ.5)GO TO 200
      JA=0
      JB=0
      JC=0
      KA=NA+1
      KB=NB+1
      IF(NA.GE.NB)GO TO 94
      NC=NB
      KC=KB
      GO TO 100
 94   NC=NA
      KC=KA
100   JC=JC+1
      KC=KC-1
      IF(JC.GT.KC)GO TO 140
      IF(JA+1.GE.KA)GO TO 124
      JA=JA+1
      KA=KA-1
      X1=XR(IA,JA)
      X2=XR(IA,KA)
      IF(JB+1.GE.KB)GO TO 120
      JB=JB+1
      KB=KB-1
      Y1=XR(IB,JB)
      Y2=XR(IB,KB)
      IF(K.EQ.2)GO TO 110
      Z1=ZF(X1,Y1,K)
      Z2=ZF(X1,Y2,K)
      IF(Z2.GT.Z1)GO TO 102
      ZR(JC)=Z2
      GO TO 104
102   ZR(JC)=Z1
104   Z1=ZF(X2,Y1,K)
      Z2=ZF(X2,Y2,K)
      IF(Z2.GT.Z1)GO TO 106
      ZR(KC)=Z1
      GO TO 100
106   ZR(KC)=Z2
      GO TO 100
110   ZR(JC)=ZF(X1,Y1,K)
      ZR(KC)=ZF(X2,Y2,K)
      IF(ZR(JC)-ZR(KC))100,112,112
112   ZR(JC)=.5*(ZR(JC)+ZR(KC))
114   KC=KC+1
      IF(KC.GT.NC)GO TO 116
      JC=JC+1
      ZR(JC)=ZR(KC)
      GO TO 114
116   NC=JC
      GO TO 140
120   Y1=.5*(XR(IB,JB)+XR(IB,KB))
```

Listing 2. The Microbial Burden Prediction Program. Subroutine HCS (MBP2)

```
      ZR(JC)=ZF(X1,Y1,K)
      ZR(KC)=ZF(X2,Y1,K)
      GO TO 100
124   X1=.5*(XR(IA,JA)+XR(IA,KA))
      IF(JB+1.GE.KB)GO TO 128
      JB=JB+1
      KB=KB-1
      Y1=XR(IB,JB)
      Y2=XR(IB,KB)
      IF(K.EQ.2)GO TO 130
      Z1=ZF(X1,Y1,K)
      Z2=ZF(X1,Y2,K)
      IF(Z2.GT.Z1)GO TO 126
      ZR(JC)=Z2
      ZR(KC)=Z1
      GO TO 100
126   ZR(JC)=Z1
      ZR(KC)=Z2
      GO TO 100
128   Y1=.5*(XR(IB,JB)+XR(IB,KB))
      ZR(JC)=ZF(X1,Y1,K)
      ZR(KC)=ZR(JC)
      GO TO 100
130   ZR(JC)=ZF(X1,Y2,K)
      ZR(KC)=ZF(X1,Y1,K)
      GO TO 100
140   NX(IC)=NC
      IF(NC.EQ.1)GO TO 35
      IF(K.EQ.2)GO TO 171
      DO 150 J=1,NC
150   CR(J)=0.
      DO 170 JA=2,NA
      DO 170 JB=2,NB
      P=DR(IA,JA)*DR(IB,JB)
      X1=XR(IA,JA-1)
      X2=XR(IA,JA)
      Y1=XR(IB,JB-1)
      Y2=XR(IB,JB)
      Z1=ZF(X1,Y1,K)
      Z2=ZF(X1,Y2,K)
      IF(Z2.GT.Z1)GO TO 154
      ZA=Z2
      GO TO 155
154   ZA=Z1
155   Z1=ZF(X2,Y1,K)
      Z2=ZF(X2,Y2,K)
      IF(Z2.GT.Z1)GO TO 158
      ZB=Z1
      GO TO 160
158   ZB=Z2
160   DO 170 JC=2,NC
      IF(ZR(JC).LE.ZA)GO TO 170
      IF(ZR(JC).GE.ZB)GO TO 166
      CR(JC)=CR(JC)+P*(ZR(JC)-ZA)/(ZB-ZA)
      GO TO 170
166   CR(JC)=CR(JC)+P
170   CONTINUE
      GO TO 300
171   DO 172 JA=2,NA
172   CA(JA)=DR(IA,JA)
      DO 180 JC=2,NC
```

Listing 2. The Microbial Burden Prediction Program, Subroutine HCS (MBP2)

```
      Z1=ZR(JC-1)
      Z2=ZR(JC)
      DO 174 JA=1,NA
174   QR(JA,JC)=0.
      DO 178 JB=2,NB
      Y1=XR(IB,JB-1)
      Y2=XR(IB,JB)
      XA=ZF(Z1,Y1,1)
      XB=ZF(Z2,Y2,1)
      DO 178 JA=2,NA
      IF(XR(IA,JA).LE.XA)GO TO 178
      IF(XR(IA,JA).GE.XB)GO TO 176
      QR(JA,JC)=QR(JA,JC)+DR(IB,JB)*(XR(IA,JA)-XA)/(XB-XA)
      GO TO 178
176   QR(JA,JC)=QR(JA,JC)+DR(IB,JB)
178   CONTINUE
      KA=NA
      DO 180 JA=2,NA
      QR(KA,JC)=QR(KA,JC)-QR(KA-1,JC)
      KA=KA-1
180   CONTINUE
      DO 190 JA=2,NC
      KC=0
      QRX=0.
      DO 182 JC=2,NC
      IF(QR(JA,JC).LE.QRX)GO TO 182
      KC=JC
      NQ(JA)=KC
      QRX=QR(JA,JC)
182   CONTINUE
      DO 184 JC=2,NC
184   QR(JA,JC)=QR(JA,JC)/QRX
      CA(JA)=CA(JA)/QRX
      DO 188 J=2,NA
      IF(J.EQ.JA)GO TO 188
      QX=QR(J,KC)
      DO 186 JC=2,NC
      QR(J,JC)=QR(J,JC)-QX*QR(JA,JC)
186   CONTINUE
      CA(J)=CA(J)-QX*CA(JA)
188   CONTINUE
190   CONTINUE
      DO 192 JA=2,NC
      KC=NQ(JA)
192   CR(KC)=CA(JA)
      CR(1)=0.
      DO 194 JA=2,NC
      IF(CR(JA).LT.0.)CR(JA)=0.
194   CR(JA)=CR(JA)+CR(JA-1)
      GO TO 300
200   JA=1
      JB=1
      JC=0
      KA=0
      KB=0
      CA(1)=0.
      CB(1)=0.
      NM=NA
      IF(NM.LT.NB)NM=NB
205   IF(XR(IA,JA)-XR(IB,JB))210,235,255
210   IF(KA.GT.0)GO TO 260
```

L ing 2. The Microbial Burden Prediction Program, Subroutine HCS (MBP2)

```
      IF(XR(IA,JA).LT.XR(IB,1))GO TO 225
      DB=CB(JB-1)+(CB(JB)-CB(JB-1))*(XR(IA,JA)-XR(IB,JB-1))/
     .(XR(IB,JB)-XR(IB,JB-1))
      GO TO 220
  215 DB=CB(NB)
  220 JC=JC+1
      CR(JC)=CA(JA)*DB
      ZR(JC)=XR(IA,JA)
  225 IF(JA.GE.NA)GO TO 230
      JA=JA+1
      CA(JA)=CA(JA-1)+DR(IA,JA)
      GO TO 205
  230 IF(KB.GT.0)GO TO 280
      KA=1
      GO TO 205
  235 JC=JC+1
      CR(JC)=CA(JA)*CB(JB)
      ZR(JC)=XR(IA,JA)
      IF(JA.GE.NA)GO TO 245
      JA=JA+1
      CA(JA)=CA(JA-1)+DR(IA,JA)
  240 IF(JB.GE.NB)GO TO 250
      JB=JB+1
      CB(JB)=CB(JB-1)+DR(IB,JB)
      GO TO 205
  245 IF(KB.GT.0)GO TO 280
      KA=1
      GO TO 240
  250 IF(KA.GT.0)GO TO 280
      KB=1
      GO TO 205
  255 IF(KB.GT.0)GO TO 215
      IF(XR(IB,JB).LT.XR(IA,1))GO TO 270
      DA=CA(JA-1)+(CA(JA)-CA(JA-1))*(XR(IB,JB)-XR(IA,JA-1))/
     .(XR(IA,JA)-XR(IA,JA-1))
      GO TO 265
  260 DA=CA(NA)
  265 JC=JC+1
      CR(JC)=CB(JB)*DA
      ZR(JC)=XR(IB,JB)
  270 IF(JB.GE.NB)GO TO 275
      JB=JB+1
      CB(JB)=CB(JB-1)+DR(IB,JB)
      GO TO 205
  275 IF(KA.GT.0)GO TO 280
      KB=1
      GO TO 205
  280 NC=JC
  290 IF(NC.LE.NM)GO TO 300
      DJ=1.
      JJ=2
      NC=NC-1
      DO 294 J=2,NC
      DS=(CR(J-1)+(CR(J+1)-CR(J-1))*(ZR(J)-ZR(J-1))/(ZR(J+1)-ZR(J-1))
     .-CR(J))**2
      IF(DS.GE.DJ)GO TO 294
      DJ=DS
      JJ=J
      IF(DJ.LT..0001)GO TO 296
  294 CONTINUE
  296 DO 298 J=JJ,NC
```

Listing 2. The Microbial Burden Prediction Program, Subroutine HCS (MBP2)

```
      CR(J)=CR(J+1)
  298 ZR(J)=ZR(J+1)
      GO TO 290
  300 F=1./CR(NC)
      DO 310 J=2,NC
      DR(IC,J)=F*(CR(J)-CR(J-1))
  310 XR(IC,J)=ZR(J)
      XR(IC,1)=ZR(1)
      NX(IC)=NC
      RETURN
      END
```

Listing 2. The Microbial Burden Prediction Program, Subroutine HCS (MBP2)

```
$IBFTC MBP3     FULIST,REF,M94,XR7,DD,DECK
      SUBROUTINE HES(J,K)
C
C     HISTOGRAM EQUATING SUBROUTINE
C     SET J=K.
      COMMON/BASE/K1,K2,K3,K4,K5,KK,IK,
     .KR,RUN(7),KS,STG(7),KT,TSK(7),DSC(6),
     .NE,AES(4),IEC(10),IET(10),AEC(10),AET(10),AED(10),AEF(10,4),
     .NO,IOT(20),IOQ(20),IOC(20),
     .NP,DAC(120,2),IU(120),IAB(120,6),AAG(120),AAS(120),AAT(120),
     .ND,NX(551),DR(551,11),XR(551,11),KO,
     .NS,KKT(20),IR(5),AR(4),APA(2),
     .NT,JT(100),XMT(100),XVT(100),L1,L2,L3,L4,KTS,ITE
      M=NX(K)
      NX(J)=M
      DO 10 I=1,M
      DR(J,I)=DR(K,I)
   10 XR(J,I)=XR(K,I)
      RETURN
      END
```

Listing 2. The Microbial Burden Prediction Program, Subroutine HES (MBP3)

```
$IBFTC MBP4     FULIST,REF,M94,XR7,DD,DECK
      SUBROUTINE HMS(IA,C,IC)
C
C     HISTOGRAM MULTIPLYING SUBROUTINE
C     HISTOGRAM IC EQUALS IA MULTIPLIED BY CONSTANT C
      COMMON/BASE/K1,K2,K3,K4,K5,KK,IK,
     .KR,RUN(7),KS,STG(7),KT,TSK(7),DSC(6),
     .NE,AES(4),IEC(10),IET(10),AEC(10),AET(10),AED(10),AEF(10,4),
     .NO,IOT(20),IOQ(20),IOC(20),
     .NP,DAC(120,2),IU(120),IAB(120,6),AAG(120),AAS(120),AAT(120),
     .ND,NX(551),DR(551,11),XR(551,11),KO,
     .NS,KKT(20),IR(5),AR(4),APA(2),
     .NT,JT(100),XMT(100),XVT(100),L1,L2,L3,L4,KTS,ITE
      IF(C.EQ.0.)GO TO 20
      M=NX(IA)
      NX(IC)=M
      DO 10 J=1,M
      DR(IC,J)=DR(IA,J)
   10 XR(IC,J)=XR(IA,J)*C
      DR(IC,1)=DR(IA,1)*C
      RETURN
   20 NX(IC)=1
      DR(IC,1)=0.
      XR(IC,1)=0.
      RETURN
      END
```

Listing 2. The Microbial Burden Prediction Program, Subroutine HMS (MBP4)

```
$IBFTC MBP5     FULIST,REF,M94,XR7,DD,DECK
      SUBROUTINE HNS(I)
C
C     HISTOGRAM NUMBERING SUBROUTINE
C     I IS THE NEW DISTRIBUTION INDEX.
      COMMON/BASE/K1,K2,K3,K4,K5,KK,IK,
     .KR,RUN(7),KS,STG(7),KT,TSK(7),DSC(6),
     .NE,AES(4),IEC(10),IET(10),AEC(10),AET(10),AED(10),AEF(10,4),
     .NO,IOT(20),IOQ(20),IOC(20),
     .NP,DAC(120,2),IU(120),IAB(120,6),AAG(120),AAS(120),AAT(120),
     .ND,NX(551),DR(551,11),XR(551,11),KO,
     .NS,KKT(20),IR(5),AR(4),APA(2),
     .NT,JT(100),XMT(100),XVT(100),L1,L2,L3,L4,KTS,ITE
    2 FORMAT(16H ....ND AT LIMIT)
      IF(ND.GE.500)GO TO 20
      ND=ND+1
      I=ND
      RETURN
   20 WRITE(6,2)
      CALL EXIT
      RETURN
      END
```

Listing 2. The Microbial Burden Prediction Program, Subroutine HNS (MBP5)

```
$IBFTC MBP6     FULIST,REF,M94,XR7,DD,DECK
      SUBROUTINE HWS(I)
C     HISTOGRAM WRITING SUBROUTINE
      COMMON/BASE/K1,K2,K3,K4,K5,KK,IK,
     .KR,RUN(7),KS,STG(7),KT,TSK(7),DSC(6),
     .NE,AES(4),IEC(10),IET(10),AEC(10),AET(10),AED(10),AEF(10,4),
     .NO,IOT(20),IOQ(20),IOC(20),
     .NP,DAC(120,2),IU(120),IAB(120,6),AAG(120),AAS(120),AAT(120),
     .ND,NX(551),DR(551,11),XR(551,11),KO,
     .NS,KKT(20),IR(5),AR(4),APA(2),
     .NT,JT(100),XMT(100),XVT(100),L1,L2,L3,L4,KTS,ITE
    1 FORMAT(48X,1P2E11.3,0PF10.4)
      M=NX(I)
      IF(M.GT.1)GO TO 10
      P=1.
      WRITE(6,1)XR(I,1),XR(I,1),P
      RETURN
   10 DO 20 J=2,M
   20 WRITE(6,1)XR(I,J-1),XR(I,J),DR(I,J)
      RETURN
      END
```

Listing 2. The Microbial Burden Prediction Program, Subroutine HWS (MBP6)

```
$IBFTC MBP7     FULIST,REF,M94,XR7,DD,DECK
      SUBROUTINE MAS(I,T)
C
C     MICROBIAL ACCRETION SUBROUTINE
C     I IS THE PART AFFECTED, T THE TIME OF UPDATE
      COMMON/BASE/K1,K2,K3,K4,K5,KK,IK,
     .KR,RUN(7),KS,STG(7),KT,TSK(7),DSC(6),
     .NE,AES(4),IEC(10),IET(10),AEC(10),AET(10),AED(10),AEF(10,4),
     .NO,IOT(20),IOQ(20),IOC(20),
     .NP,DAC(120,2),IU(120),IAB(120,6),AAG(120),AAS(120),AAT(120),
     .ND,NX(551),DR(551,11),XR(551,11),KO,
     .NS,KKT(20),IR(5),AR(4),APA(2),
     .NT,JT(100),XMT(100),XVT(100),L1,L2,L3,L4,KTS,ITE
      DT=T-AAT(I)
      IF(DT.LE.0.)RETURN
      AAT(I)=T
      IE=IAB(I,5)
      IF(IE.LE.0)RETURN
      IC=IEC(IE)
      I1=ND+1
      I2=ND+2
      I3=ND+3
      I4=ND+4
      C=AAG(I)*AEC(IE)/DR(IC,1)
      DO 40 J=1,4
      IF(IAB(I,J).EQ.0)GO TO 40
      IF(AEF(IE,J).EQ.0.)GO TO 40
      IB=IAB(I,J)
      A=DR(IB,1)
      F=C*AEF(IE,J)*A
      CALL HMS(IC,F,I1)
      V=AET(IE)*AES(J)
      F=EXP(-DT/V)
      CALL HMS(IB,F,I2)
      F=V*(1.-F)
      CALL HMS(I1,F,I3)
      CALL HCS(I2,I3,IB,1)
      DR(IB,1)=A
   40 CONTINUE
      RETURN
      END
```

Listing 2. The Microbial Burden Prediction Program, Subroutine MAS (MBP7)

```
$IBFTC MBP8     FULIST,REF,M94,XR7,DD,DECK
      SUBROUTINE MAVS(I,XM,XV)
C
C     MEAN VALUE AND VARIANCE DETERMINING SUBROUTINE
C     XM IS THE DISTRIBUTION MEAN, XV IS THE VARIANCE.
      COMMON/BASE/K1,K2,K3,K4,K5,KK,IK,
     .KR,RUN(7),KS,STG(7),KT,TSK(7),DSC(6),
     .NE,AES(4),IEC(10),IET(10),AEC(10),AET(10),AED(10),AEF(10,4),
     .NO,IOT(20),IOQ(20),IOC(20),
     .NP,DAC(120,2),IU(120),IAB(120,6),AAG(120),AAS(120),AAT(120),
     .ND,NX(551),DR(551,11),XR(551,11),KO,
     .NS,KKT(20),IR(5),AR(4),APA(2),
     .NT,JT(100),XMT(100),XVT(100),L1,L2,L3,L4,KTS,ITE
    1 FORMAT(/17X,11HMEAN VALUE=,1PE11.3/17X,11HVARIANCE=   ,E11.3)
      IF(NX(I).GT.1)GO TO 10
      XM=XR(I,1)
      XV=0.
      GO TO 30
   10 M=NX(I)
      XM=0.
      XV=0.
      DO 20 J=2,M
      X=.5*(XR(I,J-1)+XR(I,J))
      XS=X**2
      XM=XM+DR(I,J)*X
   20 XV=XV+DR(I,J)*XS
      XV=XV-XM**2
   30 WRITE(6,1)XM,XV
      RETURN
      END
```

Listing 2. The Microbial Burden Prediction Program, Subroutine MAVS (MBP8)

```
$IBFTC MBP9     FULIST,REF,M94,XR7,DD,DECK
      SUBROUTINE MBRS(I1,I2,I3,I4,I5,I6)
C
C     MICROBIAL BURDEN RESTART SUBROUTINE
      COMMON/BASE/K1,K2,K3,K4,K5,KK,IK,
     .KR,RUN(7),KS,STG(7),KT,TSK(7),DSC(6),
     .NE,AES(4),IEC(10),IET(10),AEC(10),AET(10),AED(10),AEF(10,4),
     .NO,IOT(20),IOQ(20),IOC(20),
     .NP,DAC(120,2),IU(120),IAB(120,6),AAG(120),AAS(120),AAT(120),
     .ND,NX(551),DR(551,11),XR(551,11),KO,
     .NS,KKT(20),IR(5),AR(4),APA(2),
     .NT,JT(100),XMT(100),XVT(100),L1,L2,L3,L4,KTS,ITE
    1 FORMAT(34H1MICROBIAL BURDEN PREDICTION MODEL/
     .19H RESTART SUBROUTINE/)
    2 FORMAT(2I7)
      REWIND 11
      WRITE(6,1)
      JSR=1000*I1+I2
   90 READ (11)KK,IK,I1,I2,I3,I4,I5,I6
      READ (11)KR,KS,KT,(RUN(J),STG(J),TSK(J),J=1,7),L1,L2,L3,L4,KTS,ITE
      GO TO (100,120,140,160,200),IK
  100 READ (11)(AES(J),J=1,4)
      DO 110 I=1,10
      READ (11)N,(DSC(J),J=1,4)
      IF(N.LE.0)GO TO 120
  110 READ (11)IEC(N),IET(N),AEC(N),AET(N),AED(N),(AEF(N,J),J=1,4)
  120 DO 130 I=1,20
      READ (11)N,(DSC(J),J=1,4)
      IF(N.LE.0)GO TO 140
  130 READ (11)IOT(N),IOQ(N),IOC(N)
  140 DO 150 I=1,120
      READ (11)N,(DSC(J),J=1,4)
      IF(N.LE.0)GO TO 160
  150 READ (11) DAC(N,1),DAC(N,2),IU(N),(IAB(N,J),J=1,6),
     .AAG(N),AAS(N),AAT(N)
  160 DO 180 I=1,500
      READ (11)N,(DSC(J),J=1,4)
      IF(N.LE.0)GO TO 200
      READ (11)L,M,(DR(N,J),XR(N,J),J=1,M)
  180 NX(N)=M
  200 READ (11)NE,NO,NP,ND,NS,NT,ITE,KO,KTS
      DO 295 I=1,20
      READ (11)N,(DSC(J),J=1,4),N1,N2
      IF(N.LE.0)GO TO 300
  220 READ (11)K,(IR(J),J=1,5),(AR(J),J=1,3)
      IF(K.EQ.0)GO TO 240
      IF(AR(1).EQ.0.)GO TO 232
      READ (11)I3,I4,IA,MA,(DR(IA,JJ),XR(IA,JJ),JJ=1,MA),IAB(I3,I4),
     .I1,I2,IB,MB,(DR(IB,JJ),XR(IB,JJ),JJ=1,MB),IAB(I1,I2)
      NX(IA)=MA
      NX(IB)=MB
  232 READ (11)I1,IAB(I1,5),AAG(I1),AAS(I1)
      GO TO 220
  240 READ (11)ITT,IT,M,(DR(ITT,J),XR(ITT,J),DR(IT,J),XR(IT,J),J=1,M)
      NX(ITT)=M
      NX(IT)=M
      KKT(N)=IT
  250 READ (11)IO,AKT,AKQ
      IF(IO)252,290,258
```

Listing 2. The Microbial Burden Prediction Program, Subroutine MBRS (MBP9)

```
252   READ (11)IT,M,(DR(IT,J),XR(IT,J),J=1,M)
      NX(IT)=M
254   READ (11) IP,LK,(AR(J),J=1,4)
      IF(IP.EQ.0)GO TO 250
      DO 256 J=1,4
      IF(AR(J).EQ.0.)GO TO 256
      IF(IAB(IP,J).EQ.0)GO TO 256
      IB=IAB(IP,J)
      READ (11)IB,M,(DR(IB,JJ),XR(IB,JJ),JJ=1,M)
      NX(IB)=M
256   CONTINUE
      GO TO 254
258   READ (11)IT,M,(DR(IT,J),XR(IT,J),J=1,M),IE,ITK,IC,AQ
      NX(IT)=M
260   READ (11) IP,LS,APD,APC,APS,APA(1),APA(2)
      IF(IP.EQ.0)GO TO 250
      READ (11)IP,(IAB(IP,J),J=1,6),IE,(AEF(IE,J),J=1,4),Q,B
      IF(AQ.EQ.0.)GO TO 275
      IF(IE.EQ.0)GO TO 275
      IF(ITK.EQ.0)GO TO 275
      DO 270 J=1,4
      READ (11)IB,F
      IF(IB.LE.0)GO TO 270
      IF(F.EQ.0.)GO TO 270
      IF(DR(IB,1).LE.0.)GO TO 270
      READ (11)IB,MM,(DR(IB,JJ),XR(IB,JJ),JJ=1,MM)
      NX(IB)=MM
270   CONTINUE
275   IF(IC.EQ.0)GO TO 260
      IF(B.EQ.0.)GO TO 260
      DO 280 J=1,2
      READ (11)IB,APA(J)
      IF(IB.LE.0)GO TO 280
      IF(DR(IB,1).LE.0.)GO TO 280
      IF(APA(J).LE.0.)GO TO 280
      IF(DR(IB,1).LE.0.)GO TO 280
      READ (11)IB,M,(DR(IB,JJ),XR(IB,JJ),JJ=1,M)
      NX(IB)=M
280   CONTINUE
      GO TO 260
290   CONTINUE
295   CONTINUE
300   IF(L3.LE.0)GO TO 302
      READ (11)L3,M,(DR(L3,J),XR(L3,J),J=1,M)
      NX(L3)=M
302   READ (11)LL,M,(DR(LL,J),XR(LL,J),J=1,M),NP,(IU(I),I=1,NP),NT
      NX(LL)=M
      DO 310 I=1,NP
      IF(IU(I).EQ.0)GO TO 310
      DO 305 J=1,4
      IB=IAB(I,J)
      IF(IB.LE.0)GO TO 305
      READ (11)IB,M,(DR(IB,JJ),XR(IB,JJ),JJ=1,M)
      NX(IB)=M
305   CONTINUE
310   CONTINUE
      READ (11)M,(DR(505,J),XR(505,J),J=1,M),NT,JT(NT),XMT(NT),XVT(NT)
      NX(505)=M
      IF(L4.LE.0)GO TO 320
      READ (11)L4,M,(DR(L4,J),XR(L4,J),J=1,M)
      NX(L4)=M
```

Listing 2. The Microbial Burden Prediction Program, Subroutine MBRS (MBP9)

```
320  KSR=1000*KS+KT
     WRITE(6,2)JSR,KSR
     IF(KSR.NE.JSR)GO TO 90
     RETURN
     END
```

Listing 2. The Microbial Burden Prediction Program, Subroutine MBRS (MBP9)

```
$IBFTC MBP10    FULIST,REF,M94,XR7,DD,DECK
      SUBROUTINE MBS
C
C     MICROBIAL BUILDUP SUBROUTINE
      COMMON/BASE/K1,K2,K3,K4,K5,KK,IK,
     .KR,RUN(7),KS,STG(7),KT,TSK(7),DSC(6),
     .NE,AES(4),IEC(10),IET(10),AEC(10),AET(10),AED(10),AEF(10,4),
     .NO,IOT(20),IOO(20),IOC(20),
     .NP,DAC(120,2),IU(120),IAB(120,6),AAG(120),AAS(120),AAT(120),
     .ND,NX(551),DR(551,11),XR(551,11),KO,
     .NS,KKT(20),IR(5),AR(4),APA(2),
     .NT,JT(100),XMT(100),XVT(100),L1,L2,L3,L4,KTS,ITE
    7 FORMAT(34H1MICROBIAL BURDEN PREDICTION MODEL/4H RUN,I5,2H, ,7A6/
     .6H STAGE,I3,2H, ,7A6/5H TASK,I4,2H, ,7A6)
    8 FORMAT(21H ENVIRONMENTS INPUTS-//6X,26HSURFACE LIFETIME MODIFIERS/
     .10X,41HSURF (1)    SURF (2)    SURF (3)    SURF (4)/8X,1P4E11.3//
     .6X,11HENVIRONMENT,15X,40HCONCENTRATION   MEAN LIFETIME   DISTANCE,
     .13X,21HSURFACE ARRIVAL RATES/32X,84HDSTR  MODIFIER  DSTR  MODIFIER
     .   COEFF.    SURF (1)    SURF (2)    SURF (3)    SURF (4)/)
    9 FORMAT(19H OPERATIONS INPUTS-//6X,9HOPERATION,25X,
     .20HDISTRIBUTION INDICES/38X,26HTIME    DIRTINESS    CONTACT/36X,
     .28HINTERVAL    FACTOR    CONCEN./)
   10 FORMAT(14H PARTS INPUTS-//6X,4HPART,39X,
     .55HSURFACE AREA/BURDEN   ENV.    SURFACE RETENTION FACTORS/49X,
     .56HDISTRIBUTION INDICES          DSTR    FALLOUT    CONTACT/49X,
     .54HTOP  EXT  MAT  OCC                    MEAN       MEAN/49X,
     .18H(1)  (2)  (3)  (4)/)
   11 FORMAT(22H DISTRIBUTIONS INPUTS-//6X,12HDISTRIBUTION,20X,
     .6HCOEFF.,8X,4HFROM,7X,2HTO,10X,5HPROB./)
   12 FORMAT(8H SUBTASK,I3,2H, ,4A6/)
   13 FORMAT(/6X,21HPREREQUISITE SUBTASK=,I3)
   14 FORMAT(/6X,22HPREREQUISITE SUBTASKS=,I3,1H,,I3)
   15 FORMAT(/6X,31HSUBTASK START TIME DISTRIBUTION/
     .14X,10HMEAN TIME=,F11.2,1H,,16X,4HFROM,7X,2HTO,10X,5HPROB./)
   16 FORMAT(/6X,42HENVIRONMENT/AREA/RETENTION FACTOR CHANGES-/)
   17 FORMAT(/6X,44HMICROBIAL BURDEN BUILDUP (OPERATIONAL LEVEL))
   18 FORMAT(/6X,28HPARTS AFFECTED BY OPERATION-//14X,
     .42HPART SURF DSTR     AREA     SOURCE        FROM,
     .7X,2HTO,10X,5HPROB./)
   19 FORMAT(/6X,43HAREA/BURDEN DISTRIBUTIONS AT FINISH OF TASK/
     .18X,23HPART SURF DSTR      AREA,11X,4HFROM,7X,2HTO,10X,5HPROB./)
   20 FORMAT(I5,1X,4A6,I5,1PE11.3,I5,6E11.3)
   21 FORMAT(I5,1X,4A6,1X,3I10)
   22 FORMAT(I5,1X,6A6,I9,3I5,2I8,2X,1P2E11.3)
   23 FORMAT(I5,1X,4A6,4X,1PE11.3)
   24 FORMAT(10X,6HCHANGE,I4,6H, PART,I4,6H, SURF,I3,1H,,10X,
     .17HADDITIONAL AREA  ,1PE11.3,10H FROM PART,I4,6H, SURF,I3)
   25 FORMAT(50X,15HNEW ENVIRONMENT,I5)
   26 FORMAT(50X,17HFALLOUT RETENTION,1PE11.3)
   27 FORMAT(50X,17HCONTACT RETENTION,1PE11.3)
   28 FORMAT(/10X,6HCHANGE,I4,6H, PART,I4,1H,)
   29 FORMAT(/10X,9HOPERATION,I3/14X,11HENVIRONMENT,I3/
     .14X,19HTIME IN HOURS, FROM,F11.2,4H  TO,F11.2)
   30 FORMAT(12X,3I5,2X,1PE11.3,8H FALLOUT)
   31 FORMAT(12X,3I5,2X,1PE11.3,8H CONTACT)
   32 FORMAT(14H TASK SUMMARY-//6X,24HSTART TIME DISTRIBUTION-/
     .14X,10HMEAN TIME=,F11.2,1H,,16X,4HFROM,7X,2HTO,10X,5HPROB./)
   33 FORMAT(/6X,25HFINISH TIME DISTRIBUTION-/
     .14X,10HMEAN TIME=,F11.2,1H,,16X,4HFROM,7X,2HTO,10X,5HPROB./)
```

Listing 2. The Microbial Burden Prediction Program, Subroutine MBS (MBP10)

```
   34  FORMAT(/10X,26HTOTAL AREA/BURDEN, SURFACE,I2,1H-/
      .14X,11HTOTAL AREA=,1PE11.3,1H,,15X,4HFROM,7X,2HTO,10X,5HPROB./)
   35  FORMAT(/10X,32HTOTAL AREA/BURDEN, ALL SURFACES-/
      .14X,11HTOTAL AREA=,1PE11.3,1H,,15X,4HFROM,7X,2HTO,10X,5HPROB./)
   36  FORMAT(16X,3I5,2X,1PE11.3)
   37  FORMAT(/10X,9HOPERATION,I3,15H, STERILIZATION/
      .14X,19HTIME IN HOURS, FROM,F11.2,4H  TO,F11.2)
   38  FORMAT(/14X,17HSTERILIZED PARTS-//14X,23HPART SURF DSTR     AREA,
      .15X,4HFROM,7X,2HTO,10X,5HPROB./)
   39  FORMAT(12X,3I5,2X,1PE11.3)
       GO TO (100,120,140,160,200),IK
C
C      ENVIRONMENTS INPUTS-
  100  WRITE(6,7)KR,(RUN(J),J=1,7),KS,(STG(J),J=1,7),KT,(TSK(J),J=1,7)
       READ  (9) (AES(J),J=1,4)
       WRITE (11)(AES(J),J=1,4)
       WRITE(6,8)(AES(J),J=1,4)
C      AES(J) IS THE SURFACE LIFETIME MODIFIER FOR SURFACE J
       DO 110 I=1,10
       READ  (9) N,(DSC(J),J=1,4)
       WRITE (11)N,(DSC(J),J=1,4)
       IF(N.LE.0)GO TO 120
       READ  (9) IEC(N),IET(N),AEC(N),AET(N),AED(N),(AEF(N,J),J=1,4)
       WRITE (11)IEC(N),IET(N),AEC(N),AET(N),AED(N),(AEF(N,J),J=1,4)
C      IEC,AEC DESCRIBE THE ENVIRONMENTS BIOTA CONCENTRATION
C      IET,AET DESCRIBE THE REFERENCE ACCRETION TIME
C      AED IS THE DISTANCE COEFFICIENT
C      AEF IS THE RATE AT WHICH, FOR ALL OTHER FACTORS STANDARD,
C         BIOTA REACH EACH SURFACE
       WRITE(6,20)N,(DSC(J),J=1,4),IEC(N),AEC(N),IET(N),AET(N),AED(N),
      .(AEF(N,J),J=1,4)
  110  IF(NE.LT.N)NE=N
       READ  (9) N
C
C      OPERATIONS INPUTS-
  120  WRITE(6,7)KR,(RUN(J),J=1,7),KS,(STG(J),J=1,7),KT,(TSK(J),J=1,7)
       WRITE(6,9)
       DO 130 I=1,20
       READ  (9) N,(DSC(J),J=1,4)
       WRITE (11)N,(DSC(J),J=1,4)
       IF(N.LE.0)GO TO 140
       READ  (9) IOT(N),IOQ(N),IOC(N)
       WRITE (11)IOT(N),IOQ(N),IOC(N)
       WRITE(6,21)N,(DSC(J),J=1,4),IOT(N),IOQ(N),IOC(N)
C      IOT IS THE OPERATION TIME INTERVAL
C      IOQ IS THE DIRTINESS FACTOR
C      IOC IS THE BIOTA CONCENTRATION FOR CONTACT CONTAMINATION
  130  IF(NO.LT.N)NO=N
       READ  (9) N
C
C      PARTS INPUTS-
  140  WRITE(6,7)KR,(RUN(J),J=1,7),KS,(STG(J),J=1,7),KT,(TSK(J),J=1,7)
       WRITE(6,10)
       DO 150 I=1,120
       READ  (9) N,(DSC(J),J=1,6)
       WRITE (11)N,(DSC(J),J=1,4)
       IF(N.LE.0)GO TO 160
       DAC(N,1)=DSC(5)
       DAC(N,2)=DSC(6)
       IU(N)=0
       AAT(N)=0.
```

Listing 2. The Microbial Burden Prediction Program, Subroutine MBS (MBP10)

```
      IF(KTS.GT.0)AAT(N)=DR(KTS,1)
      READ (9) (IAB(N,J),J=1,6),AAG(N),AAS(N)
      IF(IAB(N,6).GT.0)GO TO 142
      AAG(N)=0.
      AAS(N)=0.
 142  WRITE(11) DAC(M,1),DAC(N,2),IU(N),(IAB(N,J),J=1,6),
     .AAG(N),AAS(N),AAT(N)
C     DAC IS THE PERMANENT ALPHAMERIC DESCRIPTION OF PART N
C     IAB(J) INDICATES THE DISTRIBUTION FOR AREA/BURDEN FOR
C        J=1, TOP SURFACE
C        J=2, OTHER EXTERIOR SURFACE
C        J=3, MATED SURFACE
C        J=4, OCCLUDED SURFACE
C     IAB(5) IS THE ENVIRONMENT INDEX (MAY BE LEFT BLANK)
C     IAB(6) IS THE RETENTION DISTRIBUTION FOR CONTACT
C     AAG IS THE RETENTION DISTRIBUTION MEAN FOR FALLOUT
C     AAS IS THE RETENTION DISTRIBUTION MEAN FOR CONTACT
C     AAT IS THE LAST TIME OF ACCRETION UPDATE
      WRITE(6,22)N,(DSC(J),J=1,6),(IAB(N,J),J=1,6),AAG(N),AAS(N)
 150  IF(NP.LT.N)NP=N
      READ (9) N
C
C     DISTRIBUTIONS INPUTS -
 160  WRITE(6,7)KR,(RUN(J),J=1,7),KS,(STG(J),J=1,7),KT,(TSK(J),J=1,7)
      WRITE(6,11)
      DO 180 I=1,500
      READ (9) N,(DSC(J),J=1,4),L,M
      WRITE (11)N,(DSC(J),J=1,4)
C     L=1 INDICATES A HISTOGRAM WITH M-1 INTERVALS
C     L=2 INDICATES A HISTOGRAM WITH 4 INTERVALS BASED ON 3 INPUT VALUES
C     M=1 INDICATES THE DISTRIBUTION IS A CONSTANT
      IF(N.LE.0)GO TO 190
      IF(L.EQ.2)GO TO 170
 165  READ (9) (DR(N,J),J=1,M)
      READ (9) (XR(N,J),J=1,M)
      NX(N)=M
      GO TO 175
 170  READ (9) DR(N,1),X1,X2,X3
      U=.25*(X2-X1)/(X3-X1)
      V=.25-U
      DR(N,2)=U
      DR(N,3)=3.*U
      DR(N,4)=3.*V
      DR(N,5)=V
      XR(N,1)=X1
      XR(N,2)=.5*(X1+X2)
      XR(N,3)=X2
      XR(N,4)=.5*(X2+X3)
      XR(N,5)=X3
      M=5
      NX(N)=5
 175  IF(DR(N,1).LE.0.)CALL MVS(N)
      WRITE (11)L,M,(DR(N,J),XR(N,J),J=1,M)
      WRITE(6,23)N,(DSC(J),J=1,4),DR(N,1)
      CALL HWS(N)
 180  IF(ND.LT.N)ND=N
      READ (9) N
C
C     THE ZERO DISTRIBUTION
 190  CALL HNS(KO)
      NX(KO)=1
```

Listing 2. The Microbial Burden Prediction Program, Subroutine MBS (MBP10)

```
      DR(KO,1)=0.
      XR(KO,1)=0.
C
C     SUBTASK LEVEL MICROBIAL BUILDUP DETERMINATION
  200 IF(KTS.EQ.0)KTS=KO
C     KTS IS THE TASK START TIME
      LL=KTS
      NS=0
      WRITE(11)NE,NO,NP,ND,NS,NT,ITE,KO,KTS
      DO 295 I=1,20
      READ  (9) N,(DSC(J),J=1,4),N1,N2
      WRITE (11)N,(DSC(J),J=1,4),N1,N2
C     N1 AND N2 ARE PREREQUISITE SUBTASKS
      IF(N.LE.0)GO TO 300
      WRITE(6,7)KR,(RUN(J),J=1,7),KS,(STG(J),J=1,7),KT,(TSK(J),J=1,7)
      WRITE(6,12)N,(DSC(J),J=1,4)
      IF(NS.LT.N)NS=N
      IF(N1.GT.0)GO TO 205
      ITT=KTS
      GO TO 215
  205 J1=KKT(N1)
      IF(N2.GT.0)GO TO 210
      WRITE(6,13)N1
      ITT=J1
      GO TO 215
  210 WRITE(6,14)N1,N2
      J2=KKT(N2)
      ITT=510
      CALL HCS(J1,J2,ITT,5)
C     ITT IS THE SUBTASK START TIME INDEX
  215 T1=DR(ITT,1)
      WRITE(6,15)T1
      CALL HWS(ITT)
      NC=0
C     ENVIRONMENT,AREA/BIOTA BURDEN CHANGES-
  220 READ  (9) K,(IR(J),J=1,5),(AR(J),J=1,3)
      WRITE (11)K,(IR(J),J=1,5),(AR(J),J=1,3)
C     K IS THE CHANGE IDENTIFIER
C     IR(1),IR(2) ARE THE NEW PART AND SURFACE INDICES
C     IR(3),IR(4) ARE THE CONTRIBUTING PART AND SURFACE
C     IR(5) IS THE NEW ENVIRONMENT (IF ANY) FOR PART IR(1)
C     AR(1) IS THE AREA CHANGED TO IR(1),IR(2) FROM IR(3),IR(4)
C     AR(2) IS THE NEW AAG FACTOR
C     AR(3) IS THE NEW AAS FACTOR
C     IF IR(5), AR(1), AR(2), AR(3) ZERO, NO CHANGE IS MADE
      IF(K.LE.0)GO TO 240
      NC=NC+1
      IF(NC.EQ.1)WRITE(6,16)
      I1=IR(1)
      IF(NP.LT.I1)NP=I1
      CALL MAS(I1,T1)
      IF(AR(1).EQ.0.)GO TO 232
      I2=IR(2)
      I3=IR(3)
      CALL MAS(I3,T1)
      I4=IR(4)
      IB=IAB(I1,I2)
      IA=IAB(I3,I4)
      IF(IB.GT.0)GO TO 222
      CALL HNS(IB)
      IAB(I1,I2)=IB
```

Listing 2. The Microbial Burden Prediction Program, Subroutine MBS (MBP10)

```
      CALL HES(IB,KO)
  222 IC=ND+1
      WRITE(6,24)K,I1,I2,AR(1),I3,I4
      IF(IR(5).LE.0)GO TO 224
      IAB(I1,5)=IR(5)
      WRITE(6,25)IR(5)
  224 IF(AR(2).LE.0.)GO TO 226
      AAG(I1)=AR(2)
      WRITE(6,26)AAG(I1)
  226 IF(AR(3).LE.0.)GO TO 228
      AAS(I1)=AR(3)
      WRITE(6,27)AAS(I1)
  228 F=AR(1)/DR(IA,1)
      IF(F.LT..995)GO TO 230
      F=1.
      IAB(I3,I4)=0
  230 CALL HMS(IA,F,IC)
      CALL HCS(IB,IC,IB,1)
      F=1.-F
      CALL HMS(IA,F,IA)
      MA=NX(IA)
      MB=NX(IB)
      WRITE(11)I3,I4,IA,MA,(DR(IA,JJ),XR(IA,JJ),JJ=1,MA),IAB(I3,I4),
     .I1,I2,IB,MB,(DR(IB,JJ),XR(IB,JJ),JJ=1,MB),IAB(I1,I2)
      GO TO 238
  232 WRITE(6,28)K,I1
      IF(IR(5).LE.0)GO TO 234
      IAB(I1,5)=IR(5)
      WRITE(6,25)IR(5)
  234 IF(AR(2).LE.0.)GO TO 236
      AAG(I1)=AR(2)
      WRITE(6,26)AAG(I1)
  236 IF(AR(3).LE.0.)GO TO 238
      AAS(I1)=AR(3)
      WRITE(6,27)AAS(I1)
  238 WRITE(11)I1,IAB(I1,5),AAG(I1),AAS(I1)
      GO TO 220
C
C     OPERATIONAL LEVEL BUILDUP-
  240 IT=510+N
      CALL HES(IT,ITT)
C     ITT IS THE SUBTASK START TIME DISTRIBUTION INDEX
C     IT=510+N IS THE SUBTASK CURRENT TIME DISTRIBUTION INDEX
      KKT(N)=IT
      M=NX(ITT)
      WRITE(11)ITT,IT,M,(DR(ITT,J),XR(ITT,J),DR(IT,J),XR(IT,J),J=1,M)
      IC=ITE
      WRITE(6,17)
  250 READ  (9) IO,IKE,AKT,AKQ
C     IO IS THE OPERATION
C     IKE IS THE OPERATION ENVIRONMENT WHEN DIFFERENT FROM
C        THE TASK ENVIRONMENT OR THE PREVIOUS OPERATION ENVIRONMENT
C     AKT IS THE OPERATION TIME MODIFIER
C     AKQ IS THE DIRTINESS MODIFIER
C        IO LT 0 INDICATES A DECONTAMINATION OPERATION FOR WHICH
C        AKT IS THE FIXED OPERATION INTERVAL
      WRITE (11)IO,AKT,AKQ
      IF(IO)252,290,258
C
C     DECONTAMINATION
  252 IO=-IO
```

Listing 3. The Microbial Burden Prediction Program, Subroutine MBS (MBP10)

```
      T1=DR(IT,1)
      IF(AKT.LE.0.)GO TO 253
      I7=ND+7
      NX(I7)=1
      DR(I7,1)=AKT
      XR(I7,1)=AKT
      CALL HCS(IT,I7,IT,1)
  253 WRITE(6,37)IO,T1,DR(IT,1)
      M=NX(IT)
      WRITE (11)IT,M,(DR(IT,J),XR(IT,J),J=1,M)
      WRITE(6,38)
  254 READ (9) IP,LK,(AR(J),J=1,4)
      WRITE(11)IP,LK,(AR(J),J=1,4)
C     IP IS THE PART AFFECTED
C     LK IS THE CURVE DESCRIBING FRACTION OF BIOTA REMOVED
C     AR(J) IS THE MEAN FRACTION OF BIOTA REMOVED FROM SURFACE J
      IF(IP.LE.0)GO TO 250
      IU(IP)=IU(IP)+1
      CALL MAS(IP,T1)
      DO 256 J=1,4
      IF(AR(J).LE.0.)GO TO 256
      IB=IAB(IP,J)
      IF(IB.LE.0)GO TO 256
      A=DR(IB,1)
      F=AR(J)/DR(LK,1)
      IC=ND+1
      CALL HMS(LK,F,IC)
      CALL HCS(IB,IC,IC,3)
      CALL HCS(IB,IC,IB,2)
      DR(IB,1)=A
      M=NX(IB)
      WRITE (11)IB,M,(DR(IB,JJ),XR(IB,JJ),JJ=1,M)
      WRITE(6,39)IP,J,IB,A
      CALL HWS(IB)
  256 CONTINUE
      GO TO 254
C
C     OPERATION-
  258 IF(IKE.GT.0)IE=IKE
      I2=ND+2
      I3=ND+3
      I4=ND+4
      I5=ND+5
      I6=ND+6
      I7=ND+7
      IC=IOC(IO)
      IQ=IOQ(IO)
      II=IET(IE)
      JC=IEC(IE)
      ITK=IOT(IO)
      T1=DR(IT,1)
      IF(ITK.EQ.0)GO TO 259
      F=AKT/DR(ITK,1)
      CALL HMS(ITK,F,I7)
      CALL HCS(IT,I7,IT,1)
      CALL MVS(IT)
C     I7 INDICATES THE OPERATION TIME DISTRIBUTION
  259 WRITE(6,29)IO,IE,T1,DR(IT,1)
      AQ=AKQ/DR(IQ,1)
      AC=AEC(IE)/DR(JC,1)
      CALL HMS(JC,AC,I6)
```

Listing 2. The Microbial Burden Prediction Program, Subroutine MBS (MBP10)

```
C      I6 INDICATES THE AIRBORN BIOTA CONCENTRATION
       M=NX(IT)
       WRITE (11)IT,M,(DR(IT,J),XR(IT,J),J=1,M),IE,ITK,IC,AQ
       WRITE(6,19)
 260   READ (9) IP,LS,APD,APC,APS,APA(1),APA(2)
C      IP IS THE PART AFFECTED
C      LS IS THE TOOL STICKINESS DISTRIBUTION
C      APD IS THE DISTANCE FROM CONTAMINATOR
C      APC IS THE TOOL BIOTA CONCENTRATION MODIFIER
C      APS IS THE MEAN TOOL STICKINESS
C      APA IS THE CONTACT AREA FOR EACH TOUCHED SURFACE
       WRITE(11)IP,LS,APD,APC,APS,APA(1),APA(2)
       IF(IP.LE.0)GO TO 250
C
C      BIOTA UPDATING-
       IU(IP)=IU(IP)+1
       CALL MAS(IP,T1)
       IG=IAB(IP,6)
       Q=AQ*EXP(-AED(IE)*APD)
       S1=.5*AAS(IP)*APC
       S2=.5*APS/DR(LS,1)
       B=0.
       IF(LS.GT.0)B=S1*S2
       WRITE(11)IP,(IAB(IP,J),J=1,6),IE,(AEF(IE,J),J=1,4),Q,B
       IF(AQ.LE.0.)GO TO 275
       IF(IE.LE.0)GO TO 275
       IF(ITK.LE.0)GO TO 275
C
C      FALLOUT CONTAMINATION-
       CALL HMS(IQ,Q,I5)
       CALL HCS(I5,I6,I5,1)
C      I5 INDICATES THE TOTAL FALLOUT SOURCE CONCENTRATION (C+Q)
       AAT(IP)=DR(IT,1)
       DO 270 J=1,4
       IB=IAB(IP,J)
       F=AEF(IE,J)*AAG(IP)
       WRITE(11)IB,F
       IF(IB.LE.0)GO TO 270
       IF(F.LE.0.)GO TO 270
       A=DR(IB,1)
       IF(A.LE.0.)GO TO 270
       F=F*A/DR(I6,1)
       CALL HMS(I6,F,I4)
       CALL HCS(I4,I5,I4,3)
       EE=AET(IE)*AES(J)/DR(II,1)
       CALL HMS(II,EE,I2)
       CALL HCS(I4,I2,I4,3)
       T=DR(I7,1)
       V=DR(I2,1)
       CALL HCS(I7,I2,I2,4)
       M=NX(I2)
       NX(I3)=M
       DO 265 JJ=1,M
       DR(I3,JJ)=DR(I2,JJ)
       XR(I2,JJ)=EXP(XR(I2,JJ))
 265   XR(I3,JJ)=1.-1./XR(I2,JJ)
       CALL HCS(I4,I3,I4,3)
       CALL HCS(IB,I2,I2,4)
       CALL HCS(I2,I4,IB,1)
       DR(IB,1)=A
       MM=NX(IB)
```

Listing 2. The Microbial Burden Prediction Program, Subroutine MBS (MBP10)

```
      WRITE(11)IB,MM,(DR(IB,JJ),XR(IB,JJ),JJ=1,MM)
      WRITE(6,30)IP,J,IB,DR(IB,1)
      CALL HWS(IB)
  270 CONTINUE
C
C     CONTACT CONTAMINATION-
  275 IF(IC.LE.0)GO TO 260
      IF(B.EQ.0.)GO TO 260
      S1=S1/(DR(IC,1)*DR(IG,1))
      DO 280 J=1,2
      IB=IAB(IP,J)
      WRITE(11)IB,APA(J)
      IF(IB.LE.0)GO TO 280
      A=DR(IB,1)
      IF(APA(J).LE.0.)GO TO 280
      IF(A.LE.0.)GO TO 280
      F=S1*APA(J)
      CALL HMS(IC,F,I4)
      CALL HCS(I4,IG,I4,3)
      F=S2*APA(J)/A
      CALL HMS(LS,F,I5)
      CALL HCS(IB,I5,I5,3)
      CALL HCS(IB,I5,IB,2)
      CALL HCS(IB,I4,IB,1)
      DR(IB,1)=A
      M=NX(IB)
      WRITE(11)IB,M,(DR(IB,JJ),XR(IB,JJ),JJ=1,M)
      WRITE(6,31)IP,J,IB,DR(IB,1)
      CALL HWS(IB)
  280 CONTINUE
      GO TO 260
  290 IF(DR(LL,1).LT.DR(IT,1))LL=IT
  295 CONTINUE
      READ (9) N
  300 IF(L3.LE.0)GO TO 302
      L3=L3+530
      CALL HES(L3,LL)
      M=NX(L3)
      WRITE(11)L3,M,(DR(L3,J),XR(L3,J),J=1,M)
  302 M=NX(LL)
      WRITE(11)LL,M,(DR(LL,J),XR(LL,J),J=1,M),NP,(IU(I),I=1,NP),NT
      T=DR(LL,1)
      CALL HES(501,KO)
      CALL HES(502,KO)
      CALL HES(503,KO)
      CALL HES(504,KO)
      CALL HES(505,KO)
      WRITE(6,7)KR,(RUN(J),J=1,7),KS,(STG(J),J=1,7),KT,(TSK(J),J=1,7)
      WRITE(6,32)DR(KTS,1)
      CALL HWS(KTS)
      WRITE(6,33)T
      CALL HWS(LL)
      IF(L4.GT.0)WRITE(6,19)
      DO 310 I=1,NP
      IF(IU(I).EQ.0)GO TO 310
      CALL MAS(I,T)
      DO 305 J=1,4
      JJ=500+J
      IB=IAB(I,J)
      IF(IB.LE.0)GO TO 305
      M=NX(IB)
```

Listing 2. The Microbial Burden Prediction Program, Subroutine MBS (MBP10)

```
      WRITE(11) IB,M,(DR(IB,L),XR(IB,L),L=1,M)
      CALL HCS(JJ,IB,JJ,1)
      IF(L4.LE.0) GO TO 305
      WRITE(6,36) I,J,IB,DR(IB,1)
      CALL HWS(IB)
305   CONTINUE
310   CONTINUE
      DO 320 J=1,4
      JJ=500+J
      WRITE(6,34) J,DR(JJ,1)
      CALL HWS(JJ)
      CALL MAVS(JJ,XM,XV)
320   CALL HCS(505,JJ,505,1)
      WRITE(6,35) DR(505,1)
      CALL HWS(505)
      CALL MAVS(505,XM,XV)
      JT(NT)=KT
      XMT(NT)=XM
      XVT(NT)=XV
      M=NX(505)
      WRITE(11) M,(DR(505,J),XR(505,J),J=1,M),NT,JT(NT),XMT(NT),XVT(NT)
      IF(L4.LE.0) RETURN
      L4=L4+530
      CALL HES(L4,505)
      M=NX(L4)
      WRITE(11) L4,M,(DR(L4,J),XR(L4,J),J=1,M)
      RETURN
      END
```

Listing 2. The Microbial Burden Prediction Program, Subroutine MBS (MBP10)

```
$IBFTC MBP11    FULIST,REF,M94,XR7,DD,DECK
      SUBROUTINE MVS(I)
C
C     MEAN VALUE SUBROUTINE
      COMMON/BASE/K1,K2,K3,K4,K5,KK,IK,
     .KR,RUN(7),KS,STG(7),KT,TSK(7),DSC(6),
     .NE,AES(4),IEC(10),IET(10),AEC(10),AET(10),AED(10),AEF(10,4),
     .NO,IOT(20),IOQ(20),IOC(20),
     .NP,DAC(120,2),IU(120),IAB(120,6),AAG(120),AAS(120),AAT(120),
     .ND,NX(551),DR(551,11),XR(551,11),KO,
     .NS,KKT(20),IR(5),AR(4),APA(2),
     .NT,JT(100),XMT(100),XVT(100),L1,L2,L3,L4,KTS,ITE
      IF(NX(I).GT.1)GO TO 10
      DR(I,1)=XR(I,1)
      RETURN
   10 M=NX(I)
      DR(I,1)=0.
      DO 20 J=2,M
   20 DR(I,1)=DR(I,1)+.5*DR(I,J)*(XR(I,J-1)+XR(I,J))
      RETURN
      END
```

Listing 2. The Microbial Burden Prediction Program, Subroutine MVS (MBP11)

```
$IBFTC MBP12    FULIST,REF,M94,XR7,DD,DECK
      FUNCTION ZF(X,Y,K)
C
C     Z FUNCTION OF X AND Y
C     K DETERMINES THE OPERATION + - * /.
      GO TO (10,20,30,40,50),K
   10 ZF=X+Y
      RETURN
   20 ZF=X-Y
      RETURN
   30 ZF=X*Y
      RETURN
   40 ZF=X/Y
      RETURN
   50 ZF=X
      IF(Y.GT.X)ZF=Y
      RETURN
      END
$DATA
@EOF
```

Listing 2. The Microbial Burden Prediction Program, Function ZF (MBP12)

```
$JOB  GMM,5848000,06904-0,78563,A   IC   15,90000          GMM  BOX 84                84
$SETUP UT5        5740
$ASSIGN           SYSUT5
$ASSIGN           SYSUT6
$ASSIGN           SYSUT8
$UNITS
$IBJOB            GO,LOGIC,MAP,SOURCE
$IBFTC MBDR       FULIST,REF,M94,XR7,DD,DECK
C
C       MICROBIAL BURDEN DATA RECOVERY PROGRAM
        COMMON/BASE/K1,K2,K3,K4,K5,KK,IK,
       .KR,RUN(7),KS,STG(7),KT,TSK(7),DSC(6),
       .NE,AES(4),IEC(10),IET(10),AEC(10),AET(10),AED(10),AEF(10,4),
       .NO,IOT(20),IOO(20),IOC(20),
       .NP,DAC(120,2),IU(120),IAB(120,5),AAG(120),AAS(120),AAT(120),
       .ND,NX(551),DR(551,11),XR(551,11),KO,
       .NS,KKT(20),IR(5),AR(4),APA(2),
       .NT,JT(100),XMT(100),XVT(100),L1,L2,L3,L4,KTS,ITE
    1  FORMAT(34H1MICROBIAL BURDEN PREDICTION MODEL/
       .22H DATA RECOVERY PROGRAM/4H RUN,I5,2H, ,7A6/
       .6H STAGE,I3,2H, ,7A6/5H TASK,I4,2H, ,7A6)
    2  FORMAT(8H SUBTASK,I3,2H, ,4A6/)
    3  FORMAT(47H BURDEN ACCRETED BY FALLOUT DURING OPERATION 16)
    4  FORMAT(38H OPERATION 16 NOT USED IN THIS SUBTASK)
       REWIND 11
   90  READ (11)KK,IK,I1,I2,I3,I4,I5,I6
       READ (11)KR,KS,KT,(RUN(J),STG(J),TSK(J),J=1,7),L1,L2,L3,L4,KTS,ITE
       GO TO (100,120,140,160,200),IK
  100  READ (11)(AES(J),J=1,4)
       DO 110 I=1,10
       READ (11)N,(DSC(J),J=1,4)
       IF(N.LE.0)GO TO 120
  110  READ (11)IEC(N),IET(N),AEC(N),AET(N),AED(N),(AEF(N,J),J=1,4)
  120  DO 130 I=1,20
       READ (11)N,(DSC(J),J=1,4)
       IF(N.LE.0)GO TO 140
  130  READ (11)IOT(N),IOQ(N),IOC(N)
  140  DO 150 I=1,120
       READ (11)N,(DSC(J),J=1,4)
       IF(N.LE.0)GO TO 160
  150  READ (11) DAC(N,1),DAC(N,2),IU(N),(IAB(N,J),J=1,6)
       .AAG(N),AAS(N),AAT(N)
  160  DO 180 I=1,500
       READ (11)N,(DSC(J),J=1,4)
       IF(N.LE.0)GO TO 200
  180  READ (11)L,M,(DR(N,J),XR(N,J),J=1,M)
       NX(N)=M
  200  READ (11)NE,NO,NP,ND,NS,NT,ITE,KO,KTS
       DO 295 I=1,20
       READ (11)N,(DSC(J),J=1,4),N1,N2
       IF(N.LE.0)GO TO 300
       WRITE(6,1)KR,(RUN(J),J=1,7),KS,(STG(J),J=1,7),KT,(TSK(J),J=1,7)
       WRITE(6,2)N,(DSC(J),J=1,4)
       NU=0
  220  READ (11)K,(IR(J),J=1,5),(AR(J),J=1,3)
       IF(K.EQ.0)GO TO 240
       IF(AR(1).EQ.0.)GO TO 232
       READ (11)I3,I4,IA,MA,(DR(IA,JJ),XR(IA,JJ),JJ=1,MA),IAB(I3,I4),
      .I1,I2,IB,MB,(DR(IB,JJ),XR(IB,JJ),JJ=1,MB),IAB(I1,I2)
```

Listing 3. Microbial Burden Data Recovery Program

```
      NX(IA)=MA
      NX(IB)=MB
232   READ (11) I1,IAB(I1,5),AAG(I1),AAS(I1)
      GO TO 220
240   READ (11) ITT,IT,M,(DR(ITT,J),XR(ITT,J),DR(IT,J),XR(IT,J),J=1,M)
      NX(ITT)=M
      NX(IT)=M
      KKT(N)=IT
250   READ (11) IO,AKT,AKQ
      IF(IO)252,290,258
252   READ (11) IT,M,(DR(IT,J),XR(IT,J),J=1,M)
      NX(IT)=M
254   READ (11) IP,LK,(AR(J),J=1,4)
      IF(IP.EQ.0)GO TO 250
      DO 256 J=1,4
      IF(AR(J).EQ.0.)GO TO 256
      IF(IAB(IP,J).EQ.0)GO TO 256
      IB=IAB(IP,J)
      READ (11) IB,M,(DR(IB,JJ),XR(IB,JJ),JJ=1,M)
      NX(IB)=M
256   CONTINUE
      GO TO 254
258   READ (11) IT,M,(DR(IT,J),XR(IT,J),J=1,M),IE,ITK,IC,AQ
      NX(IT)=M
      IF(IO.EQ.16)CALL HES(501,KO)
260   READ (11) IP,LS,APD,APC,APS,APA(1),APA(2)
      IF(IP.EQ.0)GO TO 250
      READ (11) IP,(IAB(IP,J),J=1,6),IE,(AEF(IE,J),J=1,4),Q,B
      IF(AQ.EQ.0.)GO TO 275
      IF(IE.EQ.0)GO TO 275
      IF(ITK.EQ.0)GO TO 275
      DO 270 J=1,4
      READ (11) IB,F
      IF(IB.EQ.0)GO TO 270
      IF(F.EQ.0.)GO TO 270
      IF(DR(IB,1).EQ.0.)GO TO 270
      IF(IO.EQ.16)CALL HES(502,IB)
      READ (11) IB,MM,(DR(IB,JJ),XR(IB,JJ),JJ=1,MM)
      NX(IB)=MM
      IF(IO.NE.16)GO TO 270
      CALL HCS(IB,502,502,2)
      CALL HCS(501,502,501,1)
      NU=NU+1
270   CONTINUE
275   IF(IC.EQ.0)GO TO 260
      IF(B.EQ.0.)GO TO 260
      DO 280 J=1,2
      READ (11) IB,APA(J)
      IF(IB.EQ.0)GO TO 280
      IF(APA(J).EQ.0.)GO TO 280
      READ (11) IB,M,(DR(IB,JJ),XR(IB,JJ),JJ=1,M)
      NX(IB)=M
280   CONTINUE
      GO TO 260
290   CONTINUE
      IF(NU.EQ.0)GO TO 293
      WRITE(6,3)
      CALL HWS(501)
      GO TO 295
293   WRITE(6,4)
295   CONTINUE
```

Listing 3. Microbial Burden Data Recovery Program

```
 300  IF(L3.LE.0)GO TO 302
      READ (11)L3,M,(DR(L3,J),XR(L3,J),J=1,M)
      NX(L3)=M
 302  READ (11)LL,M,(DR(LL,J),XR(LL,J),J=1,M),NP,(IU(I),I=1,NP),NT
      NX(LL)=M
      DO 310 I=1,NP
      IF(IU(I).EQ.0)GO TO 310
      DO 305 J=1,4
      IB=IAB(I,J)
      IF(IB.LE.0)GO TO 305
      READ (11)IB,M,(DR(IB,JJ),XR(IB,JJ),JJ=1,M)
      NX(IB)=M
 305  CONTINUE
 310  CONTINUE
      READ (11)M,(DR(505,J),XR(505,J),J=1,M),NT,JT(NT),XMT(NT),XVT(NT)
      NX(505)=M
      IF(L4.LE.0)GO TO 320
      READ (11)L4,M,(DR(L4,J),XR(L4,J),J=1,M)
      NX(L4)=M
 320  GO TO 90
      END
$DATA
@EOF
```

Listing 3. Microbial Burden Data Recovery Program

```
$JOB  GMM,?8 8000,06904-0,78563,A   IC   4,9000              GMM  BOX 84              84
$SETUP CK1        4450
$ASSIGN           SYSCK1
$ASSIGN           SYSUT5
$ASSIGN           SYSUT6
$ASSIGN           SYSUT8
$UNITS
$IBJOB            GO,LOGIC,MAP,SOURCE
$IBFTC MBSW       FULIST,REF,M94,XR7,DD,DECK
C
C      MICROBIAL BURDEN SPECIAL WRITE PROGRAM
C      THIS PROGRAM READS FROM 9 AND LISTS RD, SD, TD, KD, KD, KD, PE
       DIMENSION RUN(7),STG(7),TSK(7),DSC(6),ODSC(20,4),PDSC(120,6),
      .AES(4),IEC(10),IET(10),AEC(10),AET(10),AED(10),AEF(10,4),
      .IOT(20),IOQ(20),IOC(20),IU(120),IAB(120,6),AAG(120),AAS(120),
      .AAT(120),NX(551),DR(551,11),XR(551,11),KKT(20),IR(5),AR(4),APA(2)
    1  FORMAT(8I5)
    2  FORMAT(1H1)
    3  FORMAT(I5,1X,7A6,2X,4I5)
    4  FORMAT(4X,I5,1X,7A6)
    5  FORMAT(8X,I5,1X,7A6)
    6  FORMAT(16X,I5,16H DECONTAMINATION)
    7  FORMAT(12X,I5,1X,4A6)
    8  FORMAT(7E10.3)
    9  FORMAT(I5,1X,4A6,2I5)
   10  FORMAT(2I5,7E10.3)
   11  FORMAT(6I5,3E10.3)
   12  FORMAT(I5,1X,6A6)
   13  FORMAT(I5,5X,5E10.3)
   14  FORMAT(16X,I5,1X,4A6)
   16  FORMAT(I5,5X,4F10.7)
   17  FORMAT(20X,I5,1X,6A6)
       REWIND 9
       WRITE(6,2)
  100  READ  (9) KK,IK,I1,I2,I3,I4,I5,I6
       IF(KK.LE.0)CALL EXIT
       GO TO (110,120,130,100,150,100),KK
  110  READ  (9) KR,(RUN(J),J=1,7)
       WRITE(6,3)KR,(RUN(J),J=1,7)
  120  READ  (9) KS,(STG(J),J=1,7)
       WRITE(6,4)KS,(STG(J),J=1,7)
  130  READ  (9) KT,(TSK(J),J=1,7),L1,L2,L3,L4
       WRITE(6,5)KT,(TSK(J),J=1,7)
       GO TO (210,220,230,240,250),IK
  150  IF(IK.EQ.1)GO TO 100
  155  READ  (9) IZ,(DSC(J),J=1,4)
       IF(IZ.LE.0)GO TO 100
  160  READ  (9)  IP,(AR(J),J=1,4)
       IF(IP)155,155,160
C
C      ENVIRONMENTS INPUTS-
  210  READ  (9) (AES(J),J=1,4)
       DO 216 I=1,10
       READ  (9) N,(DSC(J),J=1,4)
       IF(N.LE.0)GO TO 220
       READ  (9)  IEC(N),IET(N),AEC(N),AET(N),AED(N),(AEF(N,J),J=1,4)
  216  CONTINUE
       READ  (9) N
C
```

Listing 4. The Microbial Burden Special Write Program

```
C       OPERATIONS INPUTS-
  220   DO 226 K=1,20
        READ  (9) N,(DSC(J),J=1,4)
        IF(N.LE.0)GO TO 230
        DO 222 J=1,4
  222   ODSC(N,J)=DSC(J)
        READ  (9)  IOT(N),IOQ(N),IOC(N)
  226   CONTINUE
        READ  (9) N
C
C       PARTS INPUTS-
  230   DO 236 I=1,120
        READ  (9)  N,(DSC(J),J=1,6)
        IF(N.LE.0)GO TO 240
        DO 232 J=1,6
  232   PDSC(N,J)=DSC(J)
        READ (9) (IAB(N,J),J=1,6),AAG(N),AAS(N)
  236   CONTINUE
        READ  (9) N
C
C       DISTRIBUTIONS INPUTS-
  240   DO 248  I=1,500
        READ  (9) N,(DSC(J),J=1,4),K,M
        IF(N.LE.0)GO TO 250
        IF(K.EQ.1)GO TO 245
        READ  (9) (XR(N,J),J=1,4)
        GO TO 248
  245   READ  (9) (DR(N,J),J=1,M)
        READ  (9) (XR(N,J),J=1,M)
  248   CONTINUE
        READ  (9) N
  250   DO 296 I=1,20
        READ  (9) N,(DSC(J),J=1,4),N1,N2
        IF(N.LE.0)GO TO 100
        WRITE(6,7)N,(DSC(J),J=1,4)
  270   READ  (9)  K,(IR(J),J=1,5),(AR(J),J=1,3)
        IF(K)280,280,270
  280   READ  (9)  IO,IKE,AKT,AKQ
        IF(IO)284,296,290
  284   IO=-IO
        WRITE(6,6)IO
  286   READ (9) IP,LK,(AR(J),J=1,4)
        IF(IP)280,280,286
  290   WRITE(6,14)IO,(ODSC(IO,J),J=1,4)
  292   READ (9) IP,LS,APD,APC,APS,APA(1),APA(2)
        IF(IP.LE.0)GO TO 280
        WRITE(6,17)IP,(PDSC(IP,J),J=1,6)
        GO TO 292
  296   CONTINUE
        READ  (9) N
        GO TO 100
        END
$DATA
0EOF
```

Listing 4. The Microbial Burden Special Write Program

```
MICROBIAL BURDEN PREDICTION MODEL
RUN     1. VOYAGER TEST CASE
STAGE   1. A  VERNIER MODULE ASSEMBLY + TEST
TASK    1. 1  SUBSYSTEM POSITIONING
ENVIRONMENTS INPUTS-

     SURFACE LIFETIME MODIFIERS
        SURF (1)    SURF (2)    SURF (3)    SURF (4)
       10.000E-01 10.000E-01 10.000E+00 10.000E+00
```

ENVIRONMENT	CONCENTRATION DSTR	CONCENTRATION MODIFIER	MEAN LIFETIME DSTR	MEAN LIFETIME MODIFIER	DISTANCE COEFF.	SURFACE ARRIVAL RATES SURF (1)	SURF (2)	SURF (3)	SURF (4)
1 LAMINAR FLOW TENT	1	70.000E-02	4	10.000E+01	23.000E-01	42.000E-01	14.000E-01	-0.	-0.
2 CLASS 100K CLEAN ROOM	1	30.000E-01	4	10.000E+01	46.000E-02	48.000E-01	16.000E-01	-0.	-0.
3 TEST FACILITIES	1	70.000E-01	4	10.000E+01	46.000E-02	48.000E-01	16.000E-01	-0.	-0.

Listing 5a. Microbial Burden Prediction Program Outputs, Prescribed Environments

MICROBIAL BURDEN PREDICTION MODEL
RUN 1, VOYAGER TEST CASE
STAGE 1, A VERNIER MODULE ASSEMBLY + TEST
TASK 1, 1 SUBSYSTEM POSITIONING
OPERATIONS INPUTS-

	OPERATION	DISTRIBUTION INDICES TIME INTERVAL	DIRTINESS FACTOR	CONTACT CONCEN.
1	MOVE ASSEMBLY MANUALLY	1	3	3
2	POSITION OVERHEAD CRANE	2	3	-0
3	ATTACH CRANE HOOKS	2	3	3
4	HOIST WITH CRANE	2	3	-0
5	MOVE WITH CRANE	3	3	-0
6	LOWER WITH CRANE	2	3	-0
7	DETACH CRANE HOOKS	2	3	-0
8	VISUAL INSPECTION	3	2	-0
9	LIFT WITH MSPF	2	2	2
10	PLACE IN HANDLING CONTAN	1	3	2
11	MOVE IN HANDLING CONTANR	1	2	1
12	CONNECT CABLES,HOSES,ETC	3	3	3
13	DISCONNECT CABLES,ETC.	1	3	3
14	FUNCTIONAL TESTING	3	1	-0
15	INSERT SCREW,BOLT,ETC.	2	2	1
16	TIGHTEN SCREW,BOLT,ETC.	1	2	1
17	LOOSEN SCREW,BOLT,ETC.	1	2	1
18	REMOVE SCREW,BOLT,ETC.	2	2	1
19	REMOVE FRM HNDLNG CONTNR	1	3	2

Listing 5b. Microbial Burden Prediction Program Outputs, Prescribed Operations

MICROBIAL BURDEN PREDICTION MODEL
RUN 1. VOYAGER TEST CASE
STAGE 1. A VERNIER MODULE ASSEMBLY + TEST
TASK 1. 1 SUBSYSTEM POSITIONING
PARTS INPUTS-

	PART	SURFACE AREA/BURDEN DISTRIBUTION INDICES TOP (1)	EXT (2)	MAT (3)	OCC (4)	ENV.	SURFACE DSTR	RETENTION FACTORS FALLOUT MEAN	CONTACT MEAN
1	VERNIER STRUCTURE	7	8	0	0	0	5	40.000E-02	40.000E-02
2	LANDER LEGS	9	10	-0	-0	-0	5	40.000E-02	40.000E-02
3	DEORBIT STRUCTURE	11	12	-0	-0	-0	5	40.000E-02	40.000E-02
5	AEROSHELL STRUCTURE	13	14	-0	-0	-0	5	60.000E-02	90.000E-02
6	CANISTER. FOREWARD	15	16	-0	-0	-0	5	30.000E-02	30.000E-02
7	CANISTER. AFT	17	18	-0	-0	-0	5	30.000E-02	30.000E-02
8	ADAPTER STRUCTURE	19	20	-0	-0	-0	5	20.000E-02	20.000E-02
9	VERNIER PROPULSION S/S	21	22	-0	-0	-0	5	60.000E-02	30.000E-02
10	DEORBIT PROPULSION S/S	23	24	-0	-0	-0	5	60.000E-02	30.000E-02
11	VERNIER MOD. CABLING S/S	25	26	-0	-0	-0	5	90.000E-02	50.000E-02
12	PARACHUTE TRUSS STRUCT.	27	28	-0	-0	-0	5	40.000E-02	40.000E-02
13	DEORBIT MOD. CABLING S/S	29	30	-0	-0	-0	5	90.000E-02	50.000E-02
14	AEROSHELL CABLING S/S	31	32	-0	-0	-0	5	90.000E-02	50.000E-02
15	CANISTER + ADAPT.CABLING	33	34	-0	-0	-0	5	90.000E-02	50.000E-02
16	PARACHUTE TRUSS CABLING	35	36	-0	-0	-0	5	90.000E-02	50.000E-02
17	CONTROL ELECTRONICS S/A	37	38	-0	-0	-0	5	30.000E-02	40.000E-02
18	COMPUTER S/A	39	40	-0	-0	-0	5	40.000E-02	40.000E-02
19	INERTIAL MEAS.UNIT S/A	41	42	-0	-0	-0	5	40.000E-02	40.000E-02
20	TDLR S/A	43	44	-0	-0	-0	5	40.000E-02	40.000E-02
21	AMR S/A	45	46	-0	-0	-0	5	40.000E-02	40.000E-02
22	GUID + CONT BASE FRAME	47	48	-0	-0	-0	5	50.000E-02	50.000E-02
23	TDLR ANTENNA	49	50	-0	-0	-0	5	30.000E-02	50.000E-02
24	AMR ANTENNA	51	52	-0	-0	-0	5	30.000E-02	40.000E-02
25	COMM + SEQ BASE FRAME	53	54	-0	-0	-0	5	50.000E-02	50.000E-02
26	PWR DIST(C/B) BASE FRAME	55	56	-0	-0	-0	5	50.000E-02	50.000E-02
27	PYRO CONT(C/B)BASE FRAME	57	58	-0	-0	-0	5	50.000E-02	50.000E-02
28	PWR DIST(E/P) BASE FRAME	59	60	-0	-0	-0	5	50.000E-02	50.000E-02
29	PYRO CONT(E/P)BASE FRAME	61	62	-0	-0	-0	5	50.000E-02	50.000E-02
30	PYRO CONT(C+A)(C/B) B.F.	63	64	-0	-0	-0	5	50.000E-02	50.000E-02
31	PWR ADAPTR(C+A) BASE F.	65	66	-0	-0	-0	5	50.000E-02	50.000E-02
32	PYRO CONT(C+A)BASE FRAME	67	68	-0	-0	-0	5	50.000E-02	50.000E-02
33	EQUIPMENT BATTERIES	69	70	-0	-0	-0	5	30.000E-02	30.000E-02
34	PYROTECHNICS BATTERIES	71	72	-0	-0	-0	5	30.000E-02	30.000E-02
35	TELEMETRY BASE FRAME	73	74	-0	-0	-0	5	50.000E-02	50.000E-02
36	TRANSDUCERS	75	76	-0	-0	-0	5	40.000E-02	40.000E-02
37	PRESS.CARTRIDGE EED	77	78	-0	-0	-0	5	40.000E-02	40.000E-02
38	DETONATOR EED	79	80	-0	-0	-0	5	40.000E-02	40.000E-02
39	PYROTECHNIC SQUIBS	81	82	-0	-0	-0	5	50.000E-02	30.000E-02
40	RELEASE SPRING ASSY.	83	84	-0	-0	-0	5	50.000E-02	50.000E-02
41	HEATER BLANKETS	85	86	-0	-0	-0	5	80.000E-02	30.000E-02
42	THERMOSTATS	87	88	-0	-0	-0	5	30.000E-02	30.000E-02
43	MULTILAYER INSULATION	89	90	-0	-0	-0	5	15.000E-01	40.000E-02
44	ENTRY HEAT COVERS	91	92	-0	-0	-0	5	50.000E-02	50.000E-02
45	THERMAL COATINGS	93	94	-0	-0	-0	5	60.000E-02	60.000E-02
46	UHF DUAL TRANSMITTER B.F	95	96	-0	-0	-0	5	50.000E-02	50.000E-02
47	ANTENNA + COUPLER	97	98	-0	-0	-0	5	30.000E-02	30.000E-02
48	DATA CONT. UNIT S/A	99	100	-0	-0	-0	5	30.000E-02	30.000E-02
49	DATA STORAGE UNIT S/A	101	102	-0	-0	-0	5	30.000E-02	30.000E-02
50	TV ELECTRONICS S/A	103	104	-0	-0	-0	5	30.000E-02	30.000E-02
51	SCIENCE S/S BASE FRAME	105	106	-0	-0	-0	5	50.000E-02	50.000E-02
52	TDL QUAD MASS SPECT	107	108	-0	-0	-0	5	40.000E-02	60.000E-02

Listing 5e. Microbial Burden Prediction Program Outputs, Prescribed Parts

MICROBIAL BURDEN PREDICTION MODEL
RUN 1, VOYAGER TEST CASE
STAGE 1, A VERNIER MODULE ASSEMBLY + TEST
TASK 1, 1 SUBSYSTEM POSITIONING
DISTRIBUTIONS INPUTS=

DISTRIBUTION	COEFF.	FROM	TO	PROB.
1 NARROW	91.050E-01			
		10.000E-01	30.000E-01	.0300
		30.000E-01	50.000E-01	.0900
		50.000E-01	70.000E-01	.1700
		70.000E-01	11.000E+00	.4700
		11.000E+00	15.000E+00	.1700
		15.000E+00	20.000E+00	.0700
2 MEDIUM	89.550E-01			
		10.000E-01	30.000E-01	.0700
		30.000E-01	50.000E-01	.1400
		50.000E-01	11.000E+00	.4800
		11.000E+00	14.000E+00	.1700
		14.000E+00	17.000E+00	.1000
		17.000E+00	20.000E+00	.0400
3 WIDE	62.150E-01			
		10.000E-01	50.000E-01	.3400
		50.000E-01	10.000E+00	.3800
		10.000E+00	15.000E+00	.1700
		15.000E+00	20.000E+00	.0680
		20.000E+00	25.000E+00	.0250
		25.000E+00	30.000E+00	.0170
4 LIFETIME	89.550E-01			
		10.000E-01	30.000E-01	.0700
		30.000E-01	50.000E-01	.1400
		50.000E-01	11.000E+00	.4800
		11.000E+00	14.000E+00	.1700
		14.000E+00	17.000E+00	.1000
		17.000E+00	20.000E+00	.0400
5 PART STICKINESS	30.000E-02			
		0.	20.000E-02	.2000
		20.000E-02	40.000E-02	.6000
		40.000E-02	60.000E-02	.2000
7 VERNIER STRUCTURE	80.000E+00			
		0.	0.	1.0000
8 VERNIER STRUCTURE	40.000E+01			
		0.	0.	1.0000
9 LANDER LEGS	36.000E+00			
		0.	0.	1.0000
10 LANDER LEGS	18.000E+01			
		0.	0.	1.0000
11 DEORBIT STRUCTURE	40.000E+00			
		0.	0.	1.0000
12 DEORBIT STRUCTURE	20.000E+01			
		0.	0.	1.0000
13 AEROSHELL STRUCTURE	15.000E+01			
		0.	0.	1.0000
14 AEROSHELL STRUCTURE	60.000E+01			
		0.	0.	1.0000
15 CANISTER, FOREWARD	10.000E+00			
		0.	0.	1.0000
16 CANISTER, FOREWARD	10.000E+00			
		0.	0.	1.0000
17 CANISTER, AFT	62.000E+00			
		0.	0.	1.0000

Listing 5d. Microbial Burden Prediction Program Outputs, Prescribed Distribution,

```
MICROBIAL BURDEN PREDICTION MODEL
RUN      1, VOYAGER TEST CASE
STAGE    1, A   VERNIER MODULE ASSEMBLY + TEST
TASK     1, 1   SUBSYSTEM POSITIONING
SUBTASK  1, 1.1  MANEUVER

     SUBTASK START TIME DISTRIBUTION
             MEAN TIME=        0.                FROM        TO           PROB.
                                                 0.          0.           1.0000

     ENVIRONMENT/AREA/RETENTION FACTOR CHANGES-

         CHANGE    1, PART   1,
                                                 NEW ENVIRONMENT   2

     MICROBIAL BURDEN BUILDUP (OPERATIONAL LEVEL)

         OPERATION  9
             ENVIRONMENT  1
             TIME IN HOURS, FROM       0.     TO       .35

     PARTS AFFECTED BY OPERATION-

             PART SURF DSTR     AREA    SOURCE     FROM        TO           PROB.

              1    1    7   80.000E+00 FALLOUT
                                                 0.          43.248E-02   .0050
                                                 43.248E-02  36.457E-01   .0590
                                                 36.457E-01  31.613E+00   .3807
                                                 31.613E+00  17.587E+01   .4240
                                                 17.587E+01  78.064E+01   .1162
                                                 78.064E+01  63.137E+02   .0151
              1    2    8   40.000E+01 FALLOUT
                                                 0.          72.080E-02   .0050
                                                 72.080E-02  60.762E-01   .0590
                                                 60.762E-01  52.689E+00   .3807
                                                 52.689E+00  29.312E+01   .4240
                                                 29.312E+01  13.011E+02   .1162
                                                 13.011E+02  10.523E+03   .0151
              1    2    8   40.000E+01 CONTACT
                                                 0.          45.388E+00   .0197
                                                 45.388E+00  11.774E+01   .0939
                                                 11.774E+01  29.834E+01   .3355
                                                 29.834E+01  60.568E+01   .3701
                                                 60.568E+01  18.067E+02   .1649
                                                 18.067E+02  11.408E+03   .0159
              9    1   21   18.000E+00 FALLOUT
                                                 0.          14.596E-02   .0050
                                                 14.596E-02  12.304E-01   .0590
                                                 12.304E-01  10.669E+00   .3807
                                                 10.669E+00  59.357E+00   .4240
                                                 59.357E+00  26.347E+01   .1162
                                                 26.347E+01  21.309E+02   .0151
              9    2   22   95.000E+00 FALLOUT
                                                 0.          25.678E-02   .0050
                                                 25.678E-02  21.647E-01   .0590
                                                 21.647E-01  18.770E+00   .3807
                                                 18.770E+00  10.442E+01   .4240
                                                 10.442E+01  46.351E+01   .1162
```

Listing 5e. Microbial Burden Prediction Program Outputs, Subtask Results

```
                                              46.351E+01 37.488E+02      .0151
      9    2   22   95.000E+00 CONTACT
                                               0.        33.757E+00      .0318
                                              33.757E+00 85.913E+00      .1234
                                              85.913E+00 20.300E+01      .3520
                                              20.300E+01 33.877E+01      .2790
                                              33.877E+01 84.221E+01      .1977
                                              84.221E+01 44.069E+02      .0161

  OPERATION  2
     ENVIRONMENT  1
     TIME IN HOURS, FROM         .35  TO         .55

PARTS AFFECTED BY OPERATION-

     PART SURF DSTR      AREA     SOURCE      FROM        TO            PROB.

      1    1    7   80.000E+00 FALLOUT
                                               0.        67.196E-02      .0001
                                              67.196E-02 56.856E-01      .0097
                                              56.856E-01 49.307E+00      .2577
                                              49.307E+00 27.500E+01      .5395
                                              27.500E+01 12.226E+02      .1672
                                              12.226E+02 99.332E+02      .0257
      1    2    8   40.000E+01 FALLOUT
                                               0.        45.284E+00      .0063
                                              45.284E+00 12.052E+01      .0479
                                              12.052E+01 32.734E+01      .2933
                                              32.734E+01 77.068E+01      .4330
                                              77.068E+01 25.430E+02      .1938
                                              25.430E+02 17.441E+03      .0257
```

Listing 5e. **Microbial Burden Prediction Program Outputs, Subtask Results**

```
MICROBIAL BURDEN PREDICTION MODEL
RUN     1, VOYAGER TEST CASE
STAGE   1, A  VERNIER MODULE ASSEMBLY + TEST
TASK    1, 1  SUBSYSTEM POSITIONING
TASK SUMMARY-

     START TIME DISTRIBUTION-
             MEAN TIME=       0.  ,             FROM        TO          PROB.

                                              0.          0.          1.0000

     FINISH TIME DISTRIBUTION-
             MEAN TIME=       4.39,             FROM        TO          PROB.

                                             49.572E-02 19  84E-01     .0344
                                             19.984E-01 37.567E-01     .3580
                                             37.567E-01 64.754E-01     .5293
                                             6..754E-01 84.738E-01     .0679
                                             84.738E-01 10.472E+00     .0102
                                             10.472E+00 12.471E+00     .0002

        TOTAL AREA/BURDEN, SURFACE 1-
             TOTAL AREA= 98.000E+00,            FROM        TO          PROB.

                                              0.          24.257E+00   .0000
                                             24.25?E+00 25.857E+01     .0037
                                             25.857E+01 23.177E+02     .2793
                                             23.177E+02 11.518E+03     .5883
                                             11.518E+03 48.076E+03     .1117
                                             48.076E+03 34.904E+04     .0171

                MEAN VALUE= 11.149E+03
                VARIANCE=   67.691E+07

        TOTAL AREA/BURDEN, SURFACE 2-
             TOTAL AREA= 49.500E+01,            FROM        TO          PROB.

                                              0.          12.329E+01   .0000
                                             12.329E+01 66.598E+01     .0039
                                             66.598E+01 43.696E+02     .2813
                                             43.696E+02 19.861E+03     .5817
                                             19.861E+03 81.230E+03     .1160
                                             81.230E+03 58.381E+04     .0171

                MEAN VALUE= 19.314E+03
                VARIANCE=   19.041E+08

        TOTAL AREA/BURDEN, SURFACE 3-
             TOTAL AREA=  0.        ,           FROM        TO          PROB.

                                              0.          0.          1.0000

                MEAN VALUE=  0.
                VARIANCE=    0.

        TOTAL AREA/BURDEN, SURFACE 4-
             TOTAL AREA=  0.        ,           FROM        TO          PROB.

                                              0.          0.          1.0000

                MEAN VALUE=  0.
                VARIANCE=    0.
```

Listing 5f. Microbial Burden Prediction Program Outputs, Task Summary

```
TOTAL AREA/BURDEN, ALL SURFACES-
     TOTAL AREA= 59.300E+01.                     FROM        TO          PROB.

                                          0.          14.755E+01    .0000
                                          14.755E+01  92.455E+01    .0003
                                          92.455E+01  66.874E+02    .1483
                                          66.874E+02  31.379E+03    .6560
                                          31.379E+03  12.931E+04    .1671
                                          12.931E+04  93.285E+04    .0284

        MEAN VALUE= 41.549E+03
        VARIANCE=   75.987E+08
```

Listing 5f. Microbial Burden Prediction Program Outputs, Task Summary

```
MICROBIAL BURDEN PREDICTION MODEL
RUN     1. VOYAGER TEST CASE
STAGE   1. A  VERNIER MODULE ASSEMBLY + TEST
STAGE SUMMARY-
```

TASK	MEAN BURDEN	VARIANCE
1	41.549E+03	75.987E+08
2	79.381E+03	25.303E+09
3	84.810E+03	25.624E+09
4	11.110E+04	41.964E+09
5	13.466E+04	53.906E+09
6	15.210E+04	69.310E+09
7	31.248E+04	21.803E+10
8	32.900E+04	24.673E+10
9	33.044E+04	24.552E+10
10	33.835E+04	25.851E+10
11	34.341E+04	26.101E+10

Listing 5g. Microbial Burden Prediction Program Outputs, Stage Summary

```
1 VOYAGER TEST CASE  (Pasadena Operations)                    Level 1
    1 1   VERNIER MODULE ASSEMBLY + TEST                       Level 2
       1 1   SUBSYSTEM POSITIONING                             Level 6
          1   1.1   MANEUVER                                   Level 7
                9 LIFT WITH MSPF                               Level 8
                    1 VERNIER STRUCTURE                        Level 3,5
                    9 VERNIER PROPULSION S/S                   Level 4
                2 POSITION OVERHEAD CRANE
                    1 VERNIER STRUCTURE
          2   1.2   ATTACHMENT
                3 ATTACH CRANE HOOKS
                    1 VERNIER STRUCTURE
          3   1.3   TRANSPORT
                4 HOIST WITH CRANE
                    1 VERNIER STRUCTURE
                5 MOVE WITH CRANE
                    1 VERNIER STRUCTURE
                6 LOWER WITH CRANE
                    1 VERNIER STRUCTURE
          4   1.4   DETACHMENT
                7 DETACH CRANE HOOKS
                    1 VERNIER STRUCTURE
          5   1.5   INSPECTION
                8 VISUAL INSPECTION
                    1 VERNIER STRUCTURE
       2 4   SUBSYSTEM/OSE INTERCONNECTION
          1   4.1   A/B S/S-OSE E/CONN
               12 CONNECT CABLES,HOSES,ETC
                    9 VERNIER PROPULSION S/S
          2   4.2   A/B S/S-OSE M/CONN
               12 CONNECT CABLES,HOSES,ETC
                    9 VERNIER PROPULSION S/S
       3 5   SUBSYSTEM FUNCTIONAL TESTING
          1   5.1   S/S TEST PROC. SEQ
               14 FUNCTIONAL TESTING
                    9 VERNIER PROPULSION S/S
       4 6   SUBSYSTEM/OSE DISCONNECTION
          1   6.1   A/B S/S-OSE E/DSCN
               13 DISCONNECT CABLES,ETC.
                    9 VERNIER PROPULSION S/S
          2   6.2   A/B S/S-OSE M/DSCN
               13 DISCONNECT CABLES,ETC.
                    9 VERNIER PROPULSION S/S
       5 1   SUBSYSTEM POSITIONING
          1   1.1   MANEUVER
                9 LIFT WITH MSPF
                   11 VERNIER MOD. CABLING S/S
                2 POSITION OVERHEAD CRANE
                   11 VERNIER MOD. CABLING S/S
          2   1.2   ATTACHMENT
                3 ATTACH CRANE HOOKS
                   11 VERNIER MOD. CABLING S/S
          3   1.3   TRANSPORT
                4 HOIST WITH CRANE
                   11 VERNIER MOD. CABLING S/S
                5 MOVE WITH CRANE
                   11 VERNIER MOD. CABLING S/S
                6 LOWER WITH CRANE
                   11 VERNIER MOD. CABLING S/S
          4   1.4   DETACHMENT
                7 DETACH CRANE HOOKS
```

Listing 6. Microbial Burden Special Write Program Output

```
            11 VERNIER MOD. CABLING S/S
    5  1.5  INSPECTION
         8 VISUAL INSPECTION
            11 VERNIER MOD. CABLING S/S
 6 2  SUBSYSTEM PREPARATION
    1  2.18 S/S INTEG FNL TEST
        12 CONNECT CABLES,HOSES,ETC
            11 VERNIER MOD. CABLING S/S
        14 FUNCTIONAL TESTING
            11 VERNIER MOD. CABLING S/S
        13 DISCONNECT CABLES,ETC.
            11 VERNIER MOD. CABLING S/S
 7 3  SUBSYSTEM INSTALLATION
    1  3.1  S/S COMPONNT PLACE
         1 MOVE ASSEMBLY MANUALLY
            11 VERNIER MOD. CABLING S/S
    2  3.2  S/S COMP ATTACH
        15 INSERT SCREW,BOLT,ETC.
            11 VERNIER MOD. CABLING S/S
    3  3.3  S/S COMP INTRCNECT
        12 CONNECT CABLES,HOSES,ETC
            11 VERNIER MOD. CABLING S/S
 8 4  SUBSYSTEM/OSE INTERCONNECTION
    1  4.1  A/B S/S-OSE E/CONN
        12 CONNECT CABLES,HOSES,ETC
            11 VERNIER MOD. CABLING S/S
 9 5  SUBSYSTEM FUNCTIONAL TESTING
    1  5.1  S/S TEST PROC. SEQ
        14 FUNCTIONAL TESTING
            11 VERNIER MOD. CABLING S/S
10 6  SUBSYSTEM/OSE DISCONNECTION
    1  6.1  A/B S/S-OSE E/DSCN
        13 DISCONNECT CABLES,ETC.
            11 VERNIER MOD. CABLING S/S
11 2  SUBSYSTEM PREPARATION
    1  2.1  MANEUVER
         1 MOVE ASSEMBLY MANUALLY
            17 CONTROL ELECTRONICS S/A
        10 PLACE IN HANDLING CONTAN
            17 CONTROL ELECTRONICS S/A
    2  2.2  TRANSPORT
        11 MOVE IN HANDLING CONTANR
            17 CONTROL ELECTRONICS S/A
        19 REMOVE FRM HNDLNG CONTNR
            17 CONTROL ELECTRONICS S/A
    3  2.3  INSPECTION
         8 VISUAL INSPECTION
            17 CONTROL ELECTRONICS S/A
         1 MOVE ASSEMBLY MANUALLY
            17 CONTROL ELECTRONICS S/A
    4  2.4  S/A FUNCT. TEST
        12 CONNECT CABLES,HOSES,ETC
            17 CONTROL ELECTRONICS S/A
        14 FUNCTIONAL TESTING
            17 CONTROL ELECTRONICS S/A
        13 DISCONNECT CABLES,ETC.
            17 CONTROL ELECTRONICS S/A
         1 MOVE ASSEMBLY MANUALLY
            17 CONTROL ELECTRONICS S/A
    5  2.5  S/A BURN IN TEST
        12 CONNECT CABLES,HOSES,ETC
            17 CONTROL ELECTRONICS S/A
```

Listing 6. Microbial Burden Special Write Program Output