BOEING *MILITARY AIRPLANE SYSTEMS DIVISION*

AD0 703269
01
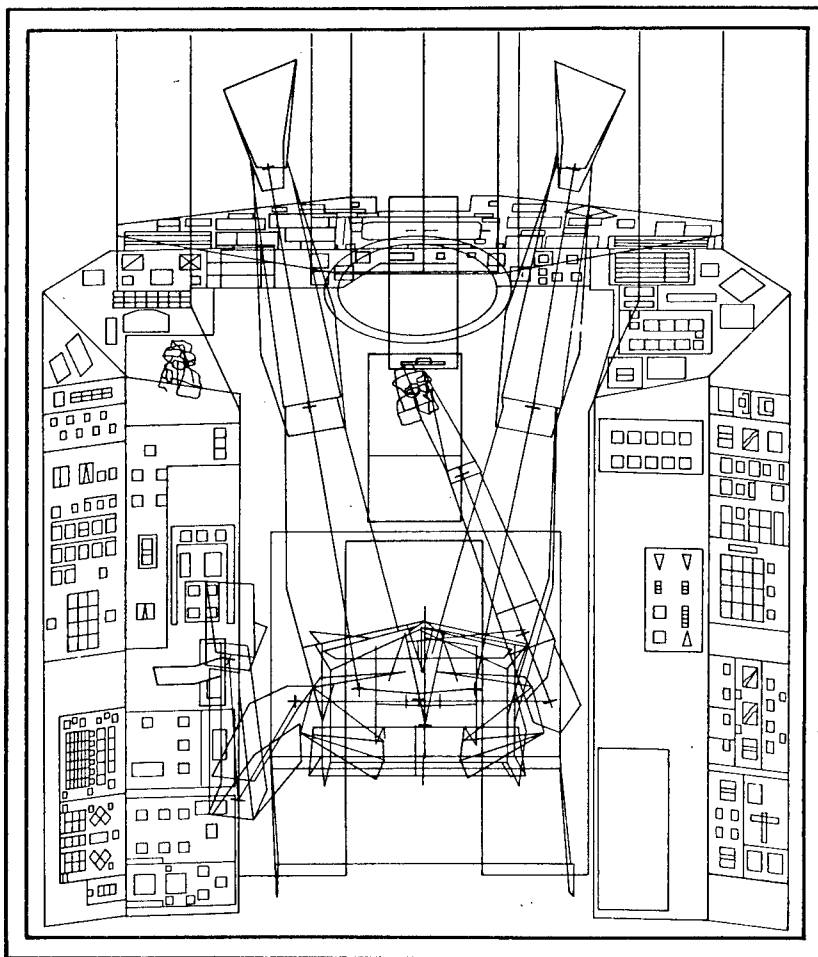
2867

2867

JANAIR

JOINT ARMY-NAVY AIRCRAFT INSTRUMENTATION RESEARCH

JANAIR REPORT 690103
ONR Contract
N00014-68-C-0289
NR 213-065

Supervised by _____
P.H. Stern

Approved by _____
V.G. Vaden

Jointly Sponsored By:

• Office of Naval Research
• Naval Air Systems Command
• U.S. Army Electronics Command

COCKPIT GEOMETRY EVALUATION

PHASE I FINAL REPORT
VOLUME III—COMPUTER PROGRAM

JANUARY 1969

20061017077

D162-10127 -1

VOLUME	DOCUMENT NUMBER	TITLE
I	D162-10125-1	PROGRAM DESCRIPTION AND SUMMARY
II	D162-10126-1	HUMAN DATA
III	D162-10127-1	COMPUTER PROGRAM
IV	D162-10128-1	MATHEMATICAL MODEL
V	D162-10129-1	VALIDATION

COCKPIT GEOMETRY EVALUATION

PHASE I

FINAL REPORT

VOLUME III–COMPUTER PROGRAM

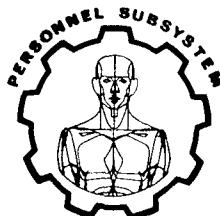|	
Coordinator, Computing Dept	R. Katz
Programmer/Analyst, Comp Dept	M.J. Healy
Programmer, Geometry and Graphics	G.O. Meeker
Group Supervisor, Comp Dept	R.R. Vaughn
Unit Chief, Computing Dept	J.S. Ryter
Project Director	W.E. Springer

20061017077

FOREWORD

This report presents work which was performed under the Joint Army Navy

Aircraft Instrumentation Research (JANAIR) Program, a research and explora-

tory development program directed by the United States Navy, Office of

Naval Research. Special guidance is provided to the program for the

Army Electronics Command, the Naval Air Systems Command, and the Office

of Naval Research through an organization known as the JANAIR Working

Group. The Working Group is currently composed of representatives from

the following offices:

U. S. Navy, Office of Naval Research
Aeronautics, Code 461, Washington, D. C.
  - Aircraft Instrumentation and Control Program Area

U. S. Navy, Naval Air Systems Command
Washington, D. C.
  - Avionics Division; Navigation Instrumentation and
    Display Branch (NAVAIR 5337)
  - Crew Systems Division; Cockpit/Cabin Requirements
    and Standards Branch (NAVAIR 5313)

U. S. Army, Army Electronics Command
Avionics Laboratory, Fort Monmouth, New Jersey
  - Instrumentation Technical Area (AMSEL-VL-I)

The Joint Army Navy Aircraft Instrumentation Research Program objective

is: To conduct applied research using analytical and experimental investi-

gations for identifying, defining and validating advanced concepts which

may be applied to future, improved Naval and Army aircraft instrumentation

systems. This includes sensing elements, data processors, displays, con-

trols and man/machine interfaces for fixed and rotary wing aircraft for

all flight regimes.

NOTICE

Change of Address

Organizations receiving JANAIR Reports on the initial distribution list should confirm correct address. This list is located at the end of the report just prior to the DDC Form 1473. Any change in address or distribution list should be conveyed to the Office of Naval Research, Code 461, Washington, D. C. 20360, Attn: JANAIR Chairman.

Disposition

When this report is no longer needed, it may be transmitted to other organizations. Do not return it to the originator or the monitoring office.

Disclaimer

The findings in this report are not to be construed as an official Department of Defense or Military Department position unless so designated by other official documents.

ABSTRACT

A computer program for the evaluation of cockpit configurations using a 23-pin-joint articulated stick-man (BOEMAN-I) is presented. The program utilizes an updatable bank of anthropological and environmental data, and simulates the motion of a real pilot performing tasks in a crewstation. The program provides information concerning reach capability, locations and orientations of joints, pilot-cockpit visual interferences, numerical performance indicators on joint displacement and deflection, and mass displacements. The program provides also a statistical validation when comparing real pilot and BOEMAN-I paths of motion.

KEYWORD LIST

Anthropometry	Non-Linear Optimization
Cockpit	Motion
Design	Performance
Environment	Simulation
Geometry	System
Human Engineering	Task
Interference	Visual
Line of Sight	Workload

TABLE OF CONTENTS

TABLE OF CONTENTS (Contd)

LIST OF ILLUSTRATIONS

1.0 INTRODUCTION

1.1 PROJECT DESCRIPTION

The Cockpit Geometry Evaluation (CGE) Methods Project is an experimental study funded by the Joint Army Navy Aircraft Instrumentation Research (JANAIR) committee to develop an objective method of evaluating the physical compatibility of crewmen with specified workstations. The first phase of this project includes the development of a Computer Program System, called BOEMAN-I. This program serves as the basic tool to store required anthropological and geometric data, make computations, and conduct evaluations of typical crew movements during performance of tasks in a specific crewstation configuration (geometry). BOEMAN-I is also the name given to the 23-pin-joint man-model upon which the system is based.

This volume of the Phase I Final Report discusses the BOEMAN-I Computer Program System.

1.2 COMPUTER PROGRAM REQUIREMENTS AND ASSUMPTIONS

The BOEMAN-I Computer Program System is a sequential set of FORTRAN IV programs for the CDC 6600 that provide the capability to (1) store large and varied amounts of data, (2) retrieve selected subsets of data, (3) calculate human body joint locations, (4) calculate selected numerical performance indicators (based on (3)), and (5) validate BOEMAN's paths of motion compared with those of real pilots. These capabilities are an outgrowth of the requirements placed on the program at the beginning of Phase I as well as various assumptions made in order that an operational evaluation tool would result.

REQUIREMENTS

The BOEMAN-I Computer Program System is designed and developed primarily
to meet seven requirements for the Cockpit Geometry Evaluation study.
These requirements may be categorized as follows: development of a
cockpit evaluation tool, and the testing of the Phase I computer program.

(1) To establish a common reference system to evaluate the physical
compatibility of an operator/crewstation layout. In the program,
two reference systems are used. The first is a Euclidean coor-
dinate system whose origin is at the cockpit eye reference point.
Data on cockpit geometry plane vertices and control locations
are expressed in this reference system initially. The second is
a Euclidean coordinate system whose origin is at the lumbar
joint, and such that when BOEMAN-I is initially seated in the
cockpit, his eye midpoint is at the cockpit eye reference point.
The program transforms all data from the first system to the
second at the beginning of an evaluation run. These coordinate
systems are necessary because the lumbar joint location is
dependent upon BOEMAN-I's link dimensions.

(2) To produce repeatable crewstation evaluation results regardless
of the investigator. Repeatable results depend on: universal
availability of and well-defined procedures for generating the
anthropological, geometric and flight mission data; consistent
application of the model in regard to step size during a task
sequence, error bounds, weighting coefficients, and preferred
angles (all of these relating to the objective function of

D162-10127-1

2

and the entire optimization procedure); relative insensitivity of the model to differing initial conditions brought about by utilizing different computers.

(3) To permit crewstation evaluations to be accurately performed using an acceptable amount of time and expense. For Phase I, hand joints with respect to the control locations are calculated with tolerance limits of one inch. Currently, on the basis of CDC 6600 computer time required to process a flight mission of seven tasks, joint position calculations (up to 15 positions per task) require between 3 and 4 minutes per task. However, this can be reduced significantly by decreasing the number of intermediate positions required in tasks of relatively small distance.

(4) To permit specific items that interfere with crew movement to be identified and indicate areas where improvement is most beneficial. The program uses bounded cockpit planes and tests each of them for the occurrence of visual interference with BOEMAN's line of sight. If interference occurs, a correction procedure is used to move BOEMAN-I as required to avoid the blocking plane. Physical interference of BOEMAN-I's links with the seatback is also identified but currently no correction is applied.

(5) To permit the evaluator to consider dynamic motion with real time effects, variations in operator size, simple and complex action and physical restraints. The BOEMAN-I system utilizes task data that simulates the duration of a human motion and the generated positions correspond to this time interval. The task sequence used in the sample input (Section 3.5) requires that the operator begin with his hands on the control stick. He then responds to a failure of the hydraulic system by resetting the master caution button, operating the utility power control, and then he resumes the initial position.

The individual link percentiles are user specified, providing for size variations of the operator. Physical restraints such as lap belts or shoulder harnesses are provided for by restrictions on the angular limits of pertinent joints.

(6) To produce results in a form applicable to either program management or design development decisions. The program produces a printed history of the flight mission or task sequence for evaluation. There are user-controlled options available to vary the size and content of the output depending on the purpose of the evaluation. The options include

suppression of any or all input data, and expansion of the processing and summation sections. The system automatically provides for a minimum of printout when a task is performed feasibly. Section 3.1 contains the user input options; Section 3.5 contains a sample printout of all input data and the processing and summation data for one task.

(7) <u>To provide a method for validation of the mathematical model of human motion.</u> The BOEMAN-I program system contains a statistical program used in the validation during Phase I. It is discussed in Section 2.1.3 of this document; comparison of BOEMAN-I and human paths of motion is discussion in the Validation Document (Volume V, D6-53620-3).

ASSUMPTIONS

The computerized model utilizes five operational assumptions:

(1) Each of BOEMAN-I's hands (palm joints) moves in a straight line during the task (provided there is no blocking object between the initial and final positions).

(2) BOEMAN-I's upper torso is movable whereas his lumbar joint, hips and legs are fixed in a seated position.

(3) Joint angular constraint limits are not part of the optimization procedure, although violations are identified for an infeasible task.

(4) Correction for visual interference is made assuming only one interfering cockpit plane.

(5) BOEMAN-I's movement during a task neglects the effects of acceleration, deceleration, and gravitational forces as well as familiarity with any given crewstation.

The remainder of this document contains a discussion of the programs in the BOEMAN-I system, a user's guide and a set of appendices detailing program subroutines, definitions, block common and control code names.

1.3 RESULTS

The computer program system developed for the Cockpit Geometry Evaluation Methods Program has the capability of positioning a 23-joint, articulated stick-man (BOEMAN-I), with variable sized link dimensions. The baseline model performs portions of a flight mission (task sequence) in a given cockpit configuration. For each mission task, numerical performance indicators are calculated and visual interference is detected and eliminated. The capability of storing and retrieving anthropological, geometric, and flight mission data is built into the program. In addition, a validation program compares BOEMAN-I's paths of motion with those of real pilots.

Successful running of the BOEMAN-I Computer Program System has been accomplished and validations performed against human movement criteria.

1.4 CONCLUSIONS

The BOEMAN-I Computer Program System demonstrates that applying a computerized human motion model to a crewstation environment is feasible. Its ability to generate relatively smooth, continuous and stable joint paths of motion, test for visual and physical seatback interference, and calculate numerical performance indicators provide a first step toward evaluating the overall physical utility of a workstation design.

1.5 RECOMMENDATIONS

(1) The present program should be refined to minimize core storage. This would reduce the problem of conversion to computers other than the CDC 6600.

(2) The program should be made more efficient in terms of both program logic and mathematical techniques used. These revisions would reduce the machine time needed to calculate joint positions.

(3) A study should be made to determine the feasibility of performing an interference analysis during joint position calculations rather than after such calculations. This would provide corrections to the paths of motion while being generated.

(4) Card input data (supplied by the user) should be reduced and made more compact to prevent non-execution runs due to card input errors.

(5) The output format of the program should be revised to provide complete flexibility, allowing the user to specify content and format of the output.

2.0   PROGRAM DESIGN

2.1   GENERAL

The set of three program groups - Storage and Retrieval, Cockpit Geometry
Evaluation (CGE), and Validation - is called the BOEMAN-I Computer Program
System. The Storage and Retrieval programs provide an input stream to the
CGE. The Cockpit Geometry Evaluation's functions include accepting and
transforming user input, synthesizing joint locations, identifying inter-
ference, providing output for the user, and generating an input stream
for the Validation. Validation compares man-made and computer-generated
paths of motion; therefore, it requires a separate stream for the experi-
mentally determined data. Each of the groups (except the storage program),
therefore, must be run sequentially since they are each dependent upon
previously generated data tapes. The activity flow is shown in Figure 1.

The Computer Program System provides a data bank of information relating
to the workstation, the crew, and the flight mission to be performed.
Data on the workstation consists of vertices of identified cockpit planes,
and a table referencing coded names and locations of controls in the
cockpit (See Appendix VIII). Crew data include a survey of size, mass, and
centroids of links stored as means, standard deviations, and a
table of normal deviates. In addition, angular limits of joints and a
standard angular orientation are stored.

Flight mission data consist of the successive controls to be reached, and
viewed, corresponding hand orientations, and time values for reaching and
holding controls. (See Section 3.5 for a listing of the data bank contents
used in Phase I.) Hence the flight mission is a sequence of tasks. A
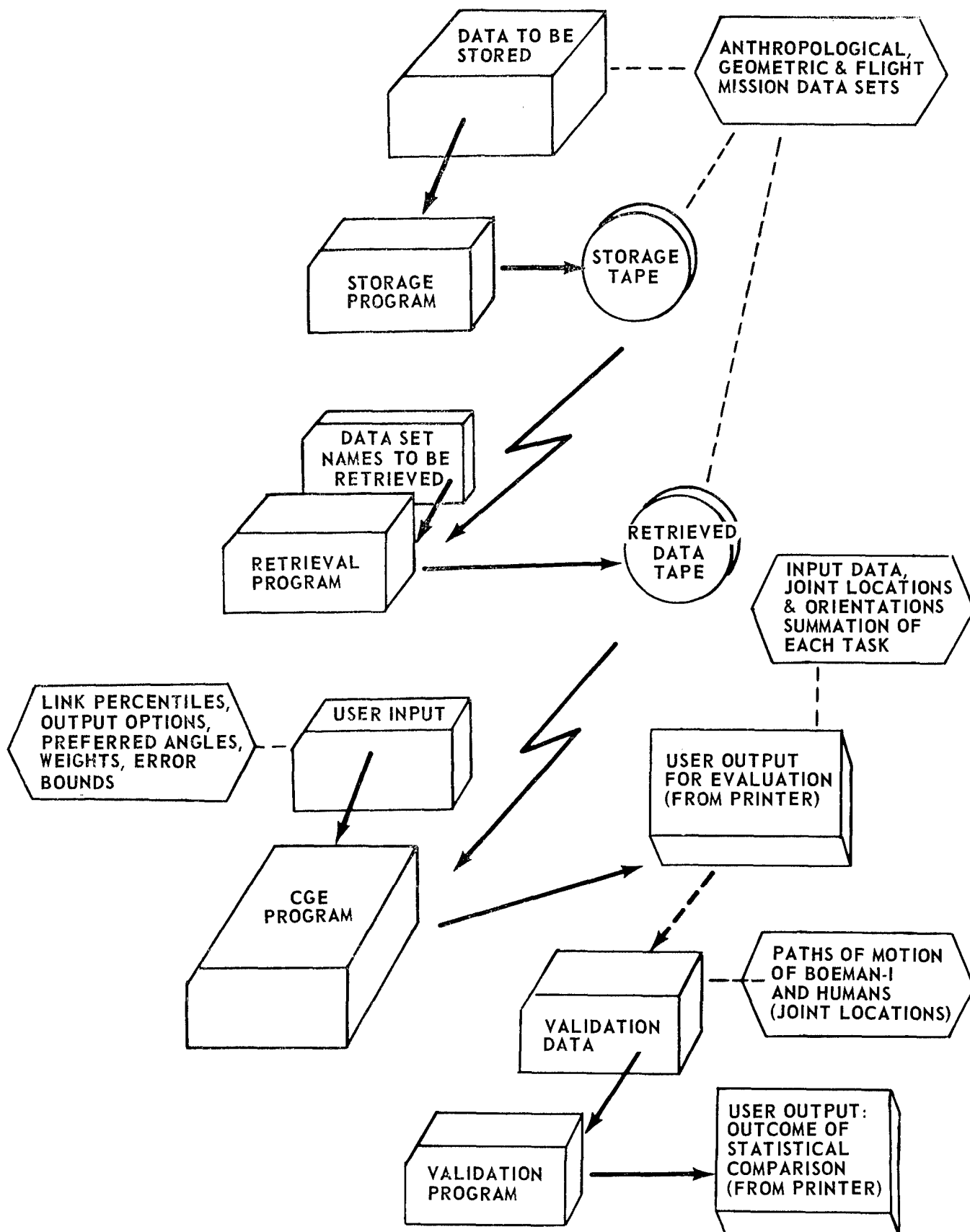
Figure 1. Data Flow Diagram

task is defined as a movement of BOEMAN-I's hand link from an initial position to a final position and/or the viewing of a specified control.

The storage program allows the user to generate on magnetic tape his own evaluation data or to add data to the existing data bank. The retrieval program allows selection of data sets from the data bank. Both of these programs are described in Sections 2.1.1, 3.1, and Appendix V. A more general description of usage may be found in Reference 13.

BOEMAN-I is required to perform a task sequence in a given cockpit configuration. Each hand must be on some control and a specific cockpit location is being viewed when the task is completed. The task is subdivided into steps, where each step is defined as that position which corresponds to an interval on a straight line path for each hand joint. For each task, paths of motion are formed based on the successive calculation of the locations and orientations of BOEMAN-I's joints.

As BOEMAN-I begins a task, it is necessary to determine whether the controls to be actuated are within reach. If they are not, the given task is deemed infeasible and is redefined to reflect the closest distance to the original task that may be feasibly reached.

Interference occurs if a control cannot be viewed or if any links intersect with the seatback.

The calculation of joint locations and orientations, discussions of reach feasibility (reach analysis), and resulting BOEMAN-workstation geometry interactions (interference analysis) may be found in Volume IV, Mathematical Model (D6-53620-2), Sections 3.2 through 3.4.

The numerical performance indicators of the evaluation include:
(1) individual task as well as cumulative task displacement of joint and
mass centroid locations, (2) work done during each task and the entire
task sequence, (3) joint angular deflections per task and task sequence
(using only twist), and (4) head and eye deflections per task and task
sequence (using full rotational capability). The calculations are de-
tailed in Section 2.2.5, Summation Overlay.

Validation consists of a statistical procedure for comparing the paths of
motion generated by BOEMAN-I with corresponding (experimentally determined)
paths of motions of humans. By use of an F-test, the hypothesis that
BOEMAN-I paths do not significantly differ from those of humans is examined.

2.1.1   Storage and Retrieval

The storage program creates a data bank of information used by the BOEMAN-I
system. The bank includes data such as anthropometric and physical charac-
teristics of a human population and is stored on magnetic tape for later
use. The bank is partitioned into data sets, each of which contains a
category of anthropometric data such as joint angular excursion limits
(or a particular version of a category, such as link dimensions and mass
quantities). The data sets must be an acceptable format for the input
transformation overlay of the evaluation program[*]. In creating the bank,
data set identification labels are supplied by the user and are utilized
in the retrieval program. In addition, the user must specify the number
of data sets to be stored and provide delimiters between data sets.

_____
*Data sets in some alternate form (or order) would require appropriate
 modification of the input transformation overlay.

The retrieval program allows for selective retrieval of one or more of the data sets using the appropriate identification labels. The retrieval data is also stored on magnetic tape and serves as a basis for the particular evaluation "run". A restriction on the retrieval program is that the order of the retrieved data sets must agree with the order required by the input transformation overlay[*]. The usage of the storage and retrieval programs may be found in Section 3.0. A list of subroutine names is given in Appendix V. Both of these programs were derived from a standard Boeing package for use in the BOEMAN-I Computer Program System.

2.1.2   Cockpit Geometry Evaluation

The Cockpit Geometry Evaluation Computer Program consists of six overlays, each having a specific function within the evaluation. These overlays are:

(1)	INTRAN	(Input Transformation)
(2)	REACHA	(Reach Analysis)
(3)	MAN1	(Baseline Man-model)
(4)	INTERF	(Interference Analysis)
(5)	SUMM	(Summation)
(6)	OUTGO	(Output)

The CGE program provides joint locations and orientations of BOEMAN-I as well as numerical indicators summarizing information about the path generated in performing a task in a cockpit configuration. To reduce storage requirements and provide separation of function, the computer program is designed in overlay form with the above programs as primary

*Data sets in some alternate form (or order) would require appropriate modification of the input transformation overlay.

overlays and with a driver program as the main overlay. The main overlay directs calls to the primaries, and communicates data to them by means of Block Common. A flow diagram for the main overlay is given in Figure 2.

Subroutine POOL is a system routine which allows a collection of files, communicating with central memory solely via binary type read/write statements, to share a single common buffer. This also yields a substantial saving in core storage.

The variables set up in the main overlay (Block Common) are called global variables as they serve the same function (or one function at a time) throughout each overlay. The variables defined in each overlay are called local variables.

The purpose or function of each subroutine in the CGE program along with the input variables necessary for the subroutine and variables calculated are given in Appendix VI. The subroutines are arranged by overlay and according to when they are first encountered in the program. Appendix X gives an index of subroutine names and page references for their description.

A list of variables belonging to Block Common statements may be found in Appendix III. These account for input/output variables when a subroutine argument list is not used.

2.1.3   Validation

The validation program tests the hypothesis that the paths of motion generated by the BOEMAN-I system and the mean motions of a like-sized individual performing the same tasks are not significantly different. The
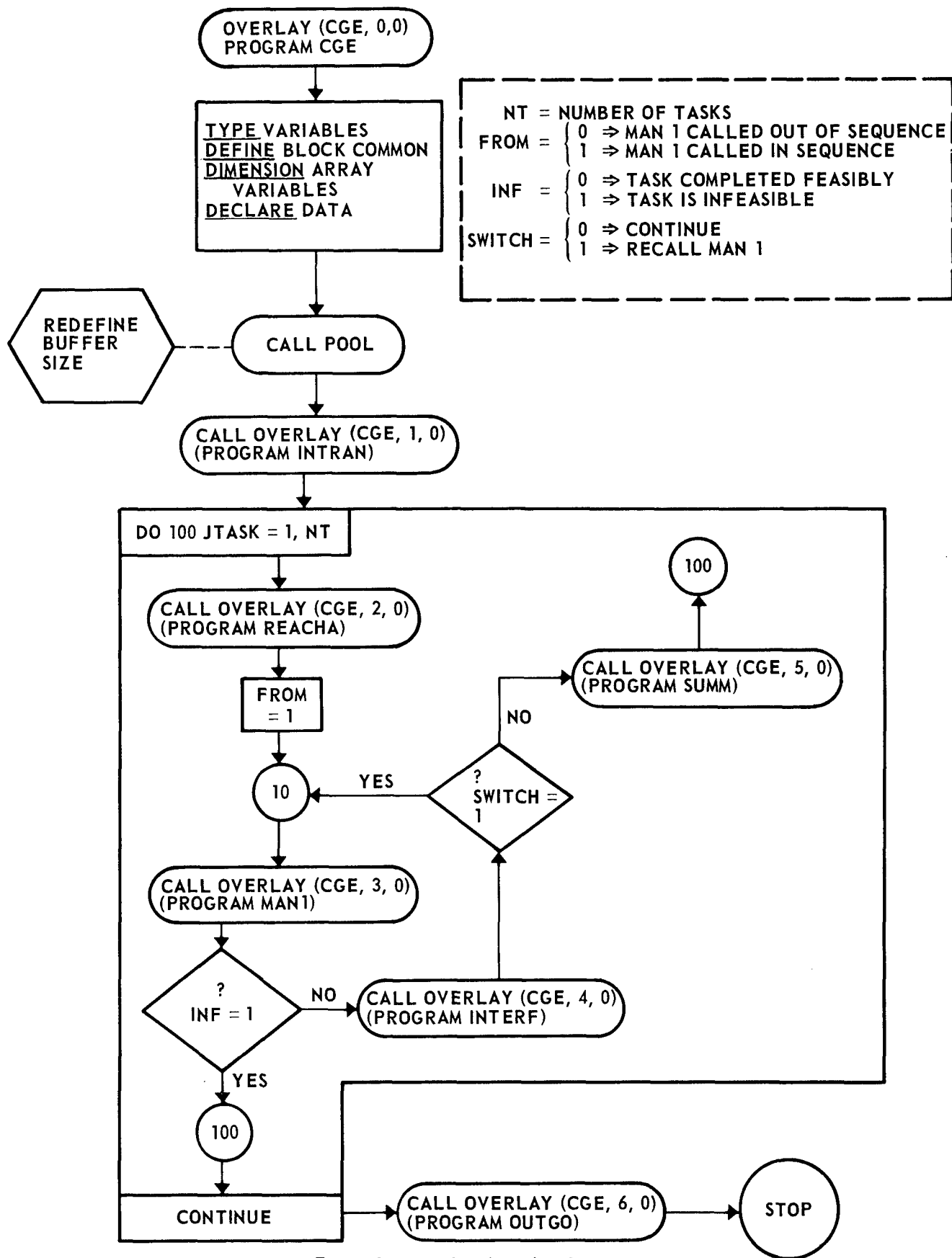
Figure 2. Main Overlay Flow Diagram

statistical procedure used to test this hypothesis involves computing an F-value that indicates whether the hypothesis should be accepted or rejected.

Thus, the input stream includes both BOEMAN's path and repeated motions of a corresponding individual during a task. The program calculates the individual's mean path and generates and inverts a covariance matrix from both paths. The result allows for the calculation of the F-value.

If the generated covariance matrix is singular, then an inverse cannot be found and no F-value can be calculated. Singularity is a function of the data used and if it exists, cannot be circumvented.

The method of the validation program is described in the validation document (Volume V). The usage of the program is contained in Section 3.0 and subroutine descriptions are presented in Appendix VII.

2.2   COCKPIT GEOMETRY EVALUATION OVERLAYS

2.2.1   Input Transformation Overlay

The purpose of Program INTRAN (OVERLAY, (CGE, 1, 0)) is to read data from the retrieval tape and user-specified punched cards. As each data set is read, it is transformed as required by MAN1 and other overlays. All input data and other variables calculated by INTRAN are written onto an intermediate file for use by the output overlay. The INTRAN overlay is called only once, handles all input to CGE, and contributes to the initial output using the intermediate file.

The program reads tape handling parameters and control options set by the user, for amount of output and an indication of whether a non-standard link dimension and mass (user defined) survey is being used. With these initialized, the link and mass data are read from the retrieval tape and with both the card input percentiles for each link and a normal distribution table (on tape), the percentiles are transformed into link lengths and masses. Task sequence information and joint angular limits are read. The angles are converted to radians. Upon completion of this, SUBROUTINE ANGLET is called upon to calculate the standard joint locations and initialize variables for the MAN1 overlay. Data describing the cockpit control codes and location of planes (with respect to the cockpit eye reference point) are then read. These and all of the data specifying "position" are transformed into locations with respect to the lumbar joint, which is dependent upon the calculated link size. Finally, the control locations specified by the task sequence are selected from the cockpit codes and stored. A list of subroutines local to this overlay may be found in Appendix I. A flow diagram for the INTRAN program is shown in Figure 3. A more detailed discussion of this overlay's subroutines may be found in Appendix VI and in user's guide (Section 4.0).

2.2.2   Reach Analysis Overlay

For each task, the reach analysis overlay determines if the control locations for each hand are within the specified sized BOEMAN's reach. If not, the control locations are repositioned to put BOEMAN within the reach envelope. This change is noted on the intermediate output file. Evaluation then proceeds with the new task based on the redefined control locations.
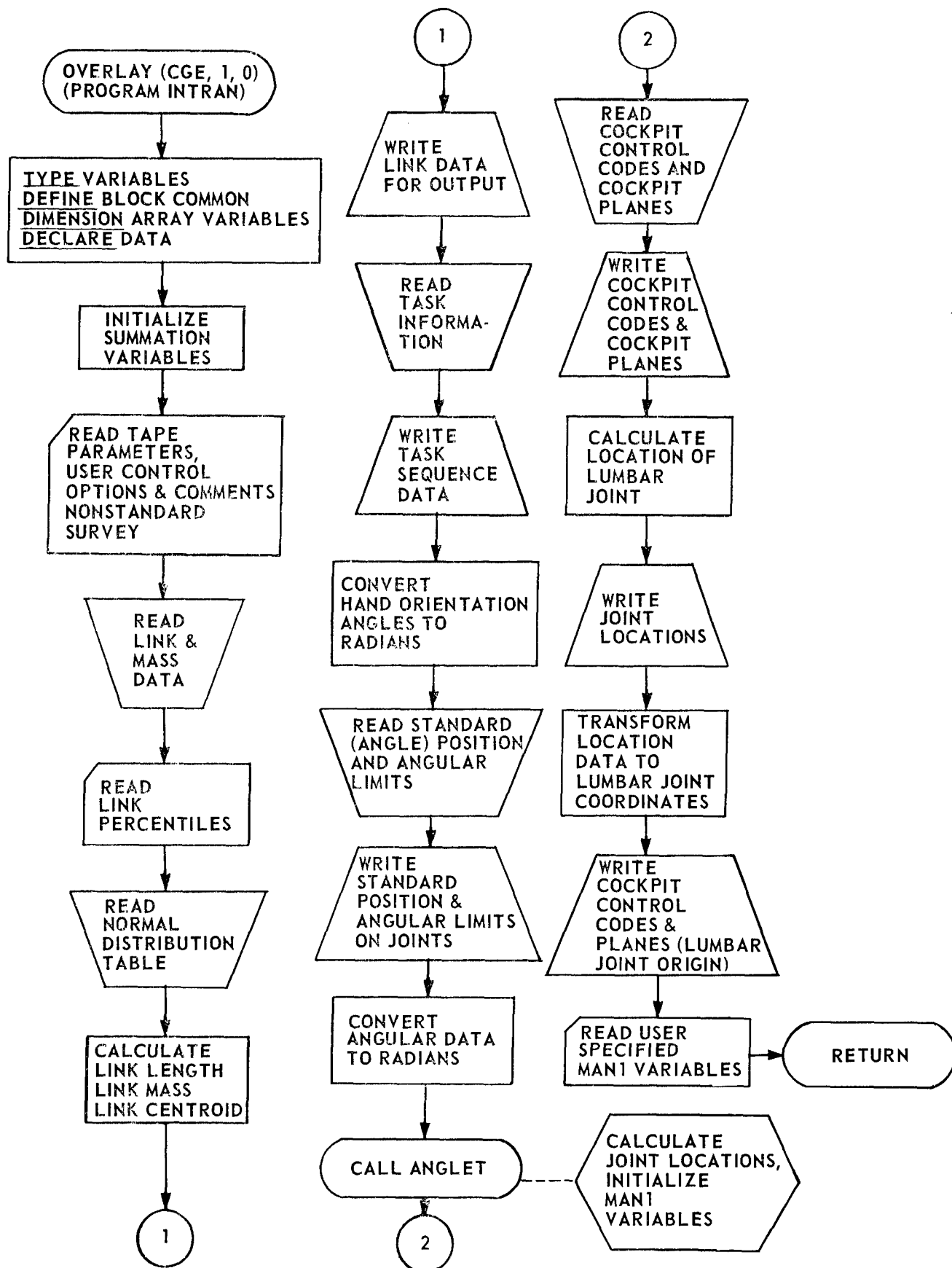
Figure 3. Program INTRAN—Input Transformation Overlay

The input required for this overlay includes link lengths, current top of spine location, joint angular limits for the spine, and current task information. Using this, a top of spine position is calculated for the given task based on the discussion in the Mathematical Model Document (Volume IV, Section 3.2) and utilizing an optimization routine called MINUM.

MINUM determines the minimum value of a function of several parameters. (See Appendix IX for a description of MINUM.) If the task is infeasible, a new task is defined and the distance between old and new hand locations is calculated. This information along with the top of spine position is saved for the output overlay on the intermediate file. A flow diagram for program REACHA is given in Figure 4.

2.2.3 Baseline Man-Model Overlay

The baseline man-model overlay is concerned with the generation of BOEMAN's joint locations and orientations at specified intervals during the performance of a task. For this, link lengths, task time, terminal joint location and orientation, joint angular limits and the initial task position are required as input. These data, along with data read from cards in the input transformation overlay are used in the optimization routine LYNX, which determines joint angular values. LYNX uses an iterative process to determine the joint angles corresponding to a minimized objective function. This is done for each step of the task. If the process does not converge (i.e., a minimum is not realized), then it is assumed that some constraints are violated, and the amount of the violation is measured. A discussion of the objective function and the optimization
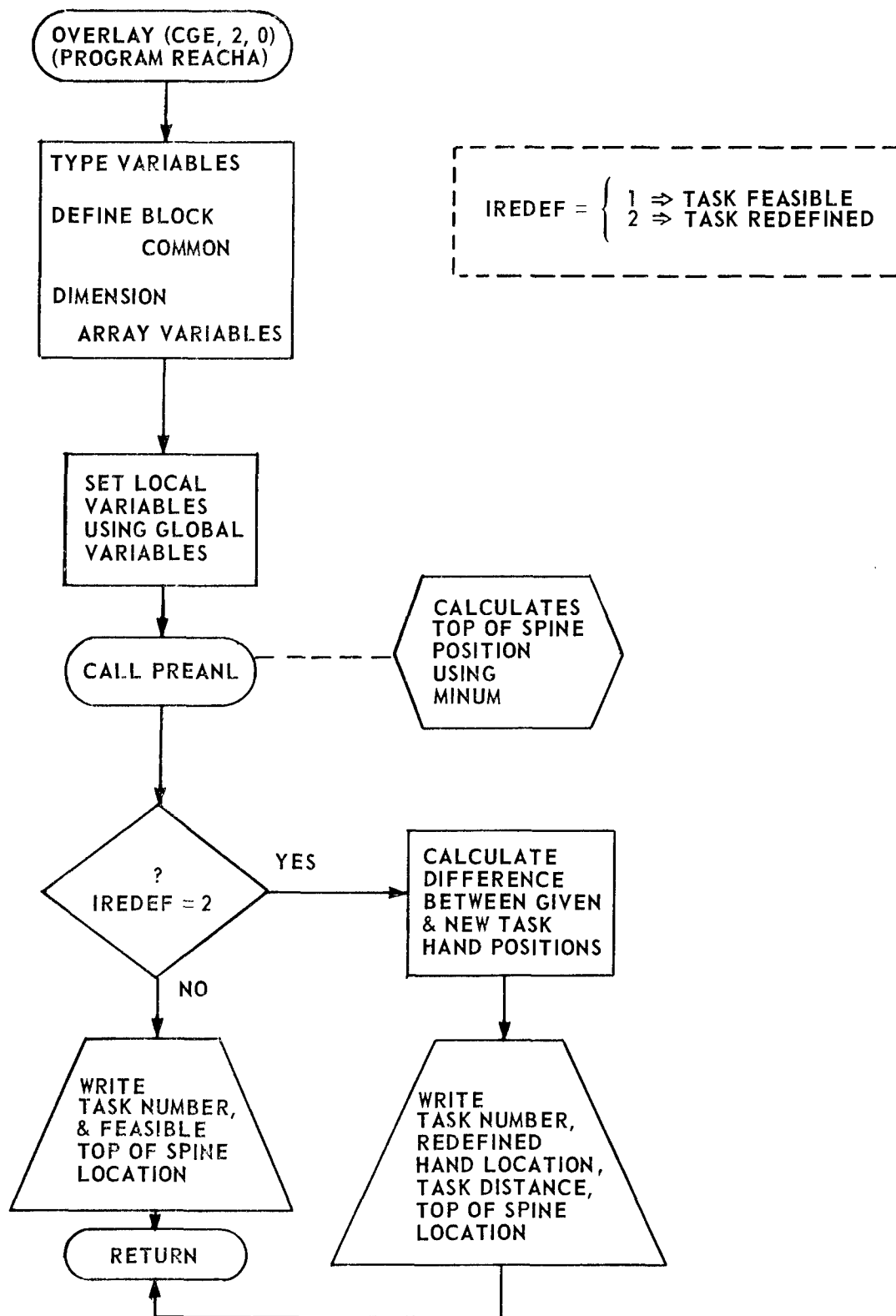
Figure 4. Reach Analysis Overlay – Program REACHA

technique may be found in Section 3.3, Mathematical Model Document
(D6-53620-2).

Program MAN1 is designed to generate supplementary output concerning
variables of the optimization algorithm, if there is an infeasibility
condition present. These variables are printed out in a separate section
for later analysis. Output for the evaluation consists of joint angle
values and joint positions at each step of the task and, for an infeasi-
bility condition, those joint angles exceeding their limits, and the
distance of each palm from its preset location. A flow diagram for
Program MAN1 is presented in Figure 5.

2.2.4   Interference Analysis Overlay

Once a set of joint locations has been generated at the end of a task,
the task is further scrutinized to determine if BOEMAN's position inter-
acts visually or physically with the cockpit. If visual interference
occurs, a correction procedure describing a new and uninterrupted line of
sight is utilized. The correction procedure is not iterative. The
testing of seatback interference occurs when the visual tests are com-
pleted. A discussion of the interference analysis may be found in Section
3.4 of the Mathematical Model Document (D6-53620-2).

Program INTERF (the interference analysis overlay) requires, as input, the
task data, the location of the eye aiming point, the vertices of each
plane comprising the cockpit geometry, the link lengths, and the calculated
joint locations during and at the end of the task. These data are stored
as local variables. The program first tests each plane for intersection
with the line of sight. If visual interference occurs, the offending

D162-10127-1

21
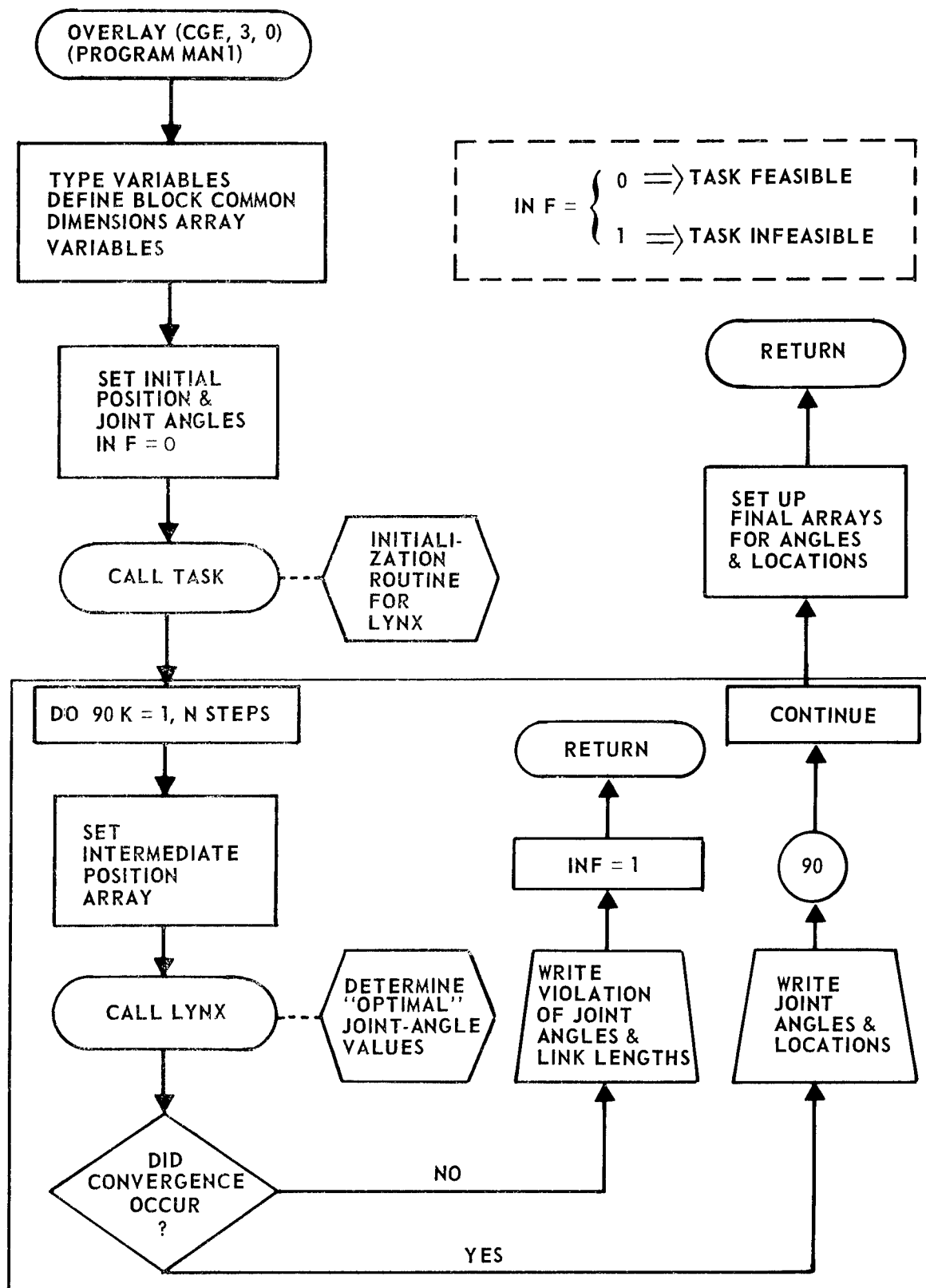
Figure 5. Baseline Man-Model Overlay—Program MAN 1

planes with each intersection point are written on the intermediate output file. Similar tests are performed during each step of the task for physical interference with the seatback. All links on BOEMAN-I which intersect the seatback are also written on the output file but no correction for this physical interference is made in the Phase I model. If no interference occurs for the task, no output is generated by the Interference overlay. A general flow diagram for Program INTERF is given in Figure 6.

2.2.5 Summation Overlay

For successfully completed tasks, it is necessary to calculate numerical performance indicators describing BOEMAN's relative change in position and orientation. Several quantities have been selected as representative of his performance for comparative analysis within the task sequence and/or with similar tasks in other cockpits. Each quantity is calculated for each task and cumulatively for the specified task sequence. The performance indicators used are:

(1) Joint displacements for each joint

(2) Centroid displacements for each link

(3) Products of link mass and centroid displacement (work)
for each link

(4) Joint angular rotations (twist) for each joint

(5) Head deflection

(6) Eye deflection

The input required for these computations includes initial and final joint locations during a task, centroid percentiles and mass on each link, task control locations, and initial joint angular and positional
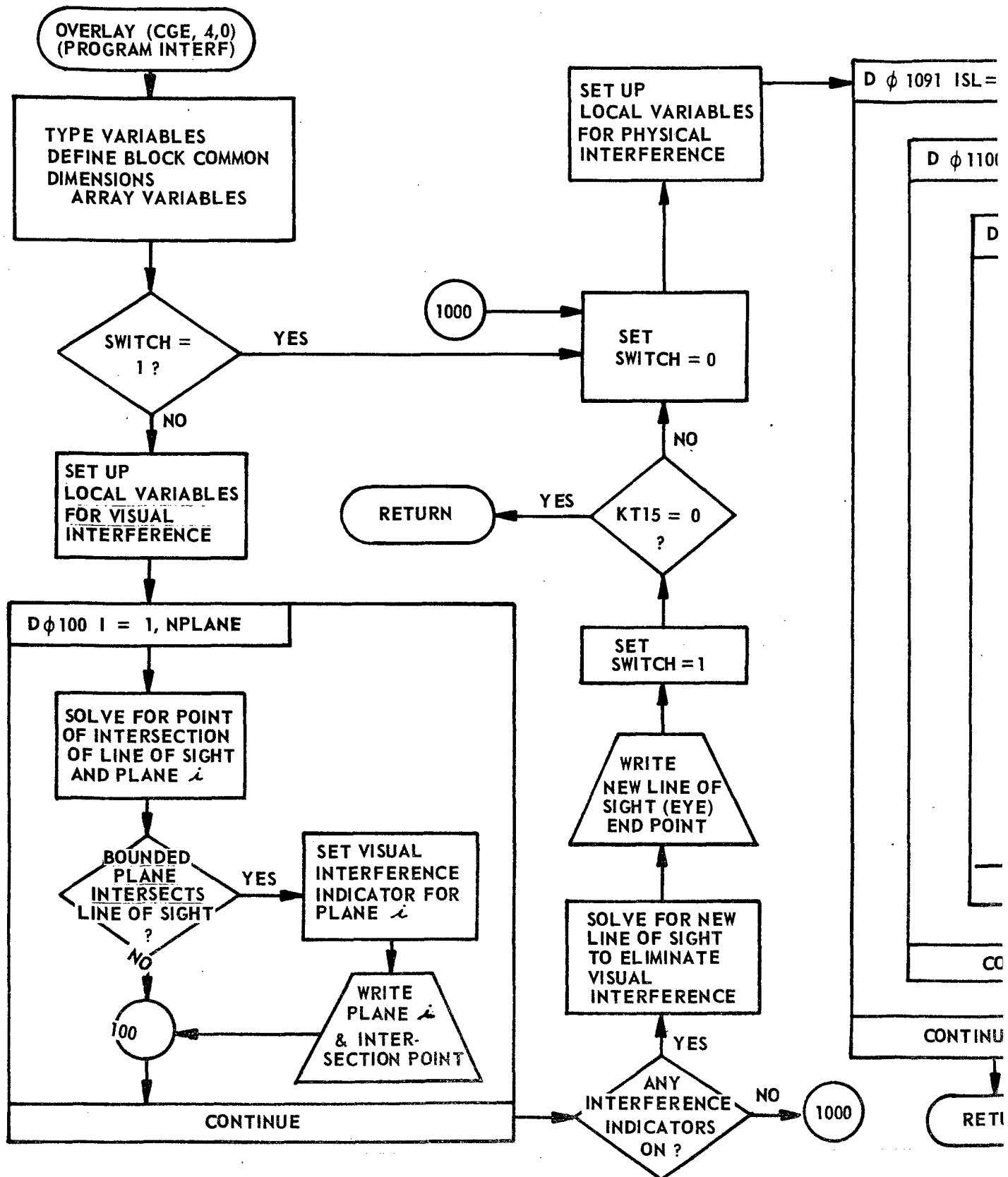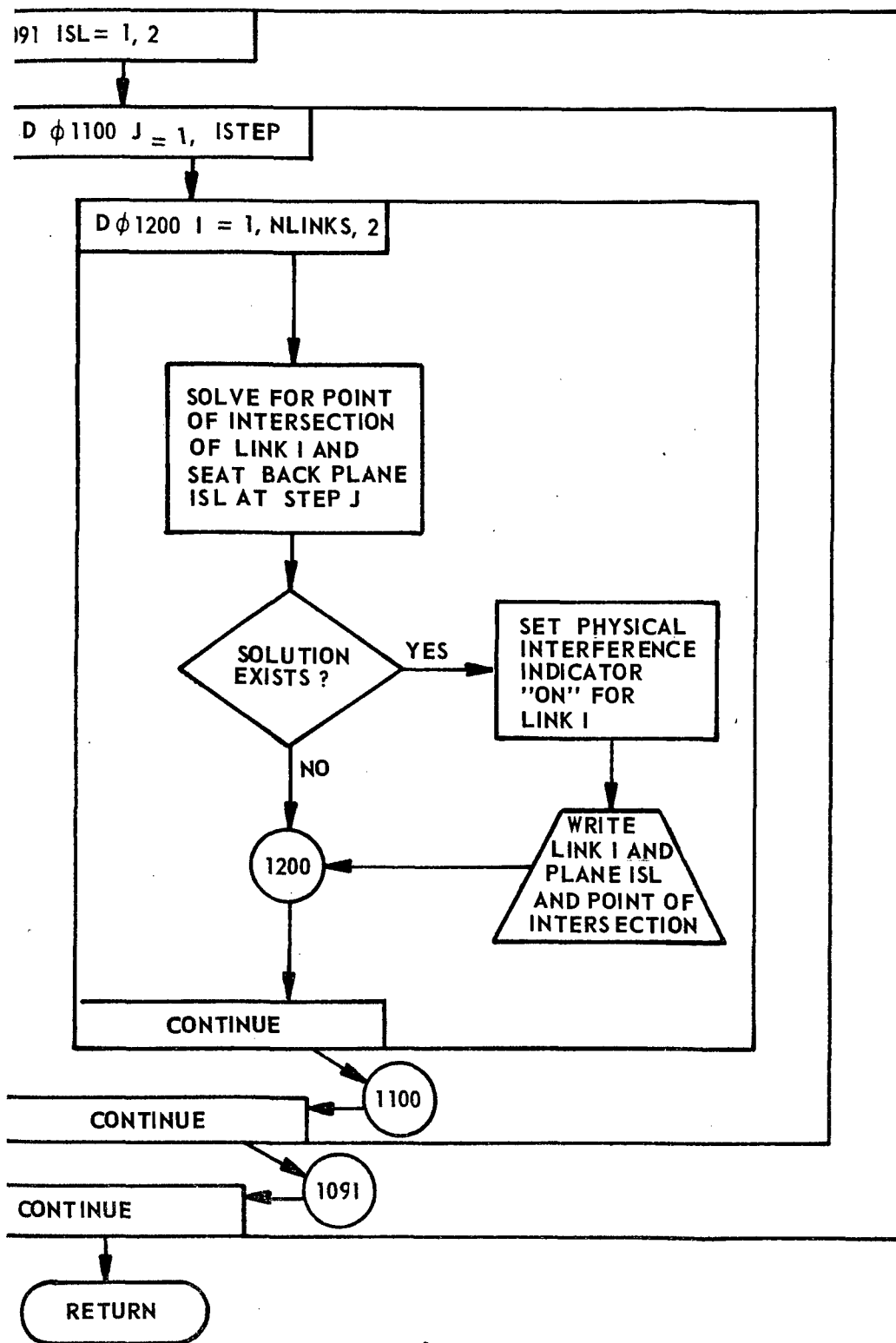
D162-10127-1

VISUAL INTERFERENCE

```
    ╭─────────────────────╮
    │ OVERLAY (CGE, 4,0)  │
    │ (PROGRAM INTERF)    │
    ╰─────────────────────╯
              │
    ┌─────────────────────┐
    │ TYPE VARIABLES      │
    │ DEFINE BLOCK COMMON │
    │ DIMENSIONS          │
    │    ARRAY VARIABLES  │
    └─────────────────────┘
              │
           SWITCH = 1 ?  ──YES──→  SET SWITCH = 0  ←── (1000)
              │ NO                        │
    ┌─────────────────────┐              │ NO
    │ SET UP              │         KT15 = 0 ?  ──YES──→ RETURN
    │ LOCAL VARIABLES     │              │
    │ FOR VISUAL          │              │
    │ INTERFERENCE        │         SET SWITCH = 1
    └─────────────────────┘              │
                                    WRITE NEW LINE OF
    DØ100 I = 1, NPLANE             SIGHT (EYE) END POINT
    ┌──────────────────────┐             │
    │ SOLVE FOR POINT      │        SOLVE FOR NEW
    │ OF INTERSECTION      │        LINE OF SIGHT
    │ OF LINE OF SIGHT     │        TO ELIMINATE
    │ AND PLANE i          │        VISUAL
    └──────────────────────┘        INTERFERENCE
                                         │ YES
    BOUNDED PLANE          SET VISUAL    │
    INTERSECTS      ──YES──→ INTERFERENCE  ANY INTERFERENCE
    LINE OF SIGHT ?          INDICATOR FOR  INDICATORS ON ? ──NO──→ (1000)
         │ NO               PLANE i
        (100)                   │
         │                 WRITE PLANE i
    CONTINUE               & INTERSECTION POINT
```

SET UP LOCAL VARIABLES FOR PHYSICAL INTERFERENCE

DØ 1091 ISL=

DØ 110(

CONTINU

RETU

Figure 6. Interference Analysis Overlay Program Interf.

values. The joint arrays are reset for the next task with BOEMAN's current orientation and position. A flow diagram for this overlay is given in Figure 7.

2.2.6   Output Overlay

Program OUTGO, the output overlay, provides for a printed history of the evaluation run, data for graphs and charts, and a magnetic tape for the validation section. The printed history is divided into three parts:

(1) Input data

(2) Results of task sequence processing

(3) Summation data

Data for charts and graphs are stored on tape to be used generally with another evaluation run for comparison purposes. These data include the numerical performance indicators calculated in the previous overlay (Program SUMM). The data required for validation include BOEMAN's joint locations during each step of a task for which comparable laboratory data exist.

The printed history is generated using the intermediate output file along with control variables (user specified) which determine the amount and kind of history required for the evaluation. These control variables are discussed in Section 3.0 (user's guide). A sample output for the evaluation may be found in Section 3.5 and a flow diagram for Program OUTGO is shown in Figure 8.
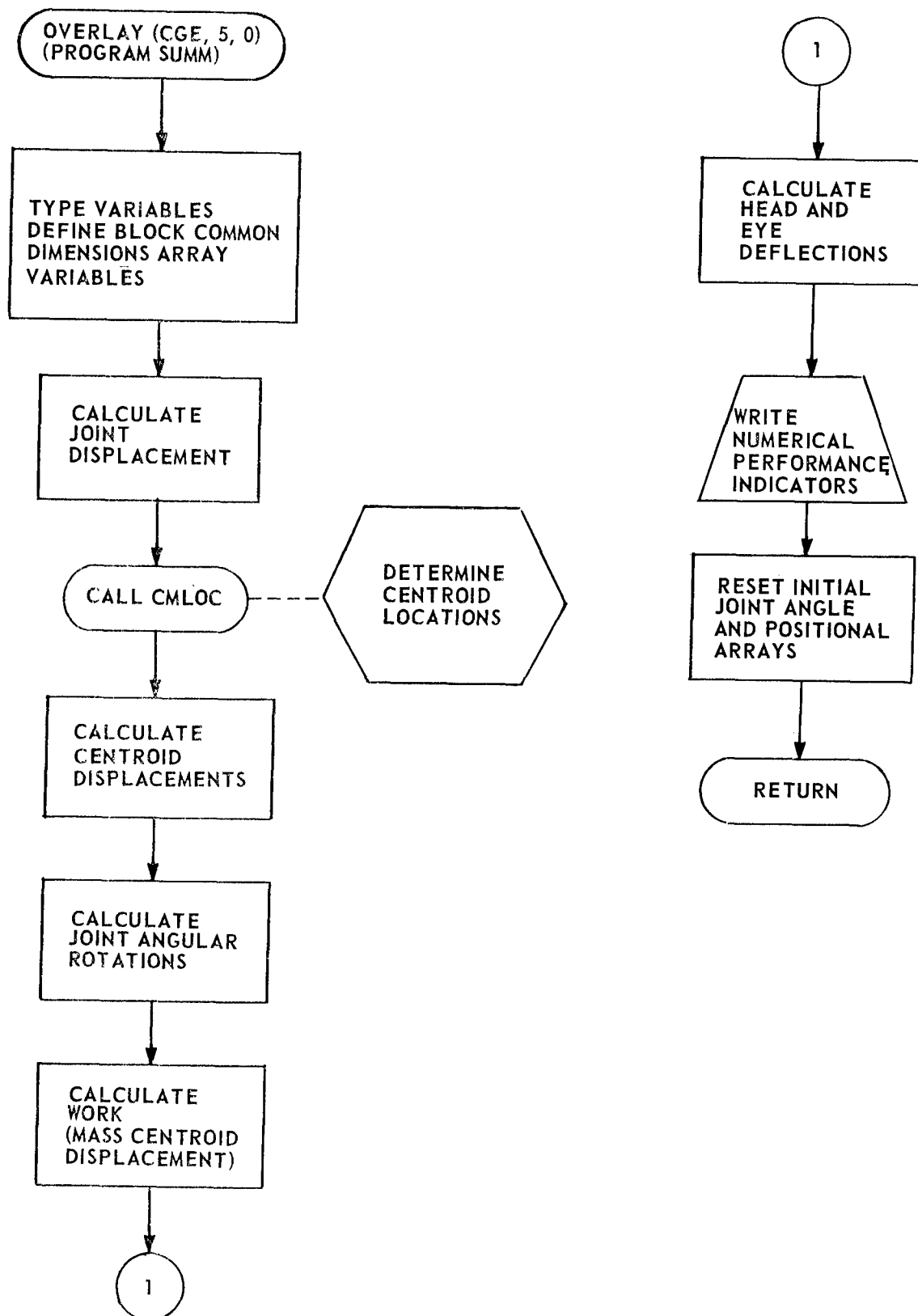
Figure 7. Summation Overlay—Program SUMM
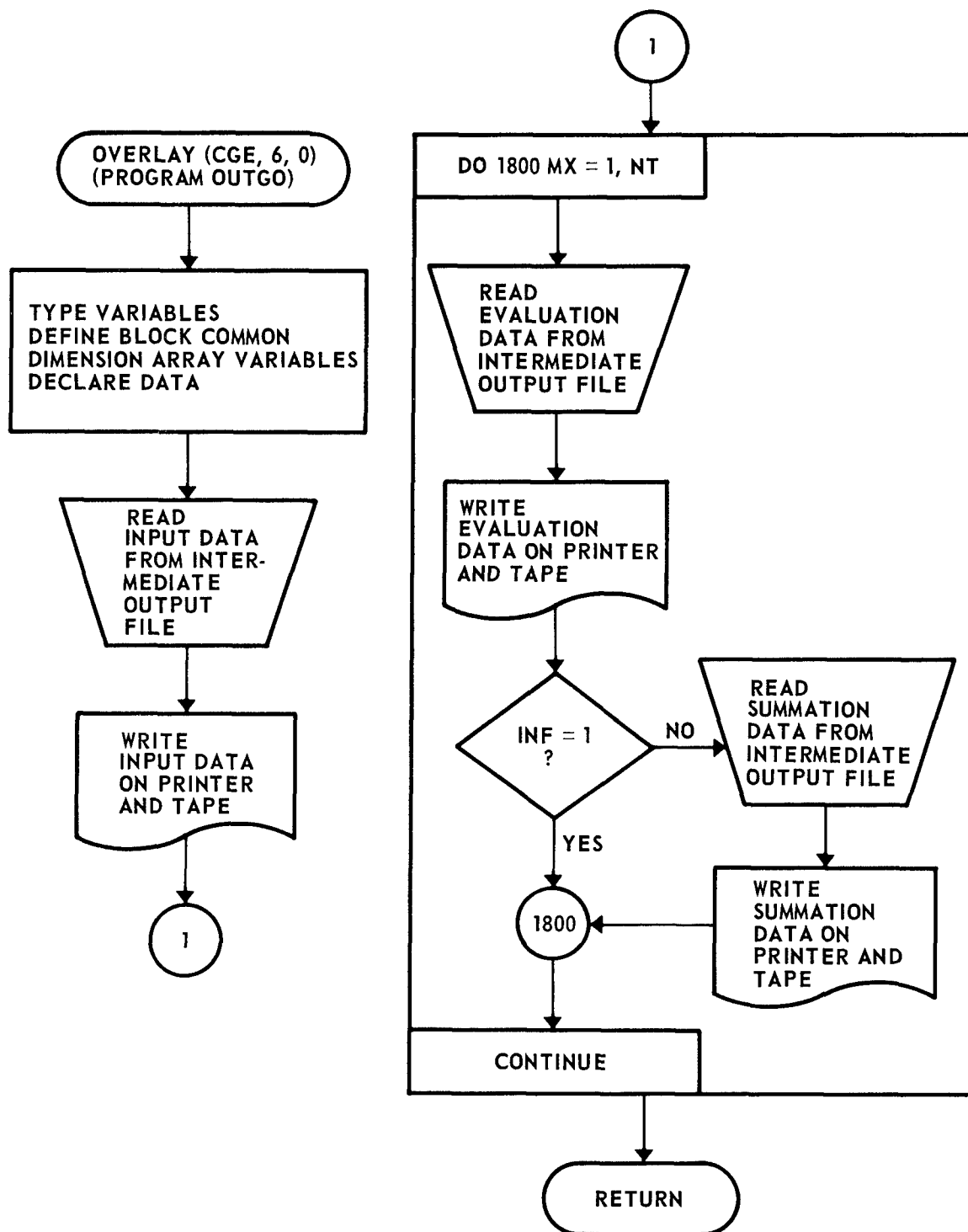
Figure 8. Output Overlay—Program OUTGO

3.0   USER'S GUIDE

The usage of Programs CAPSIS (Creation section), RAPSIS (Retrieval section),
CGE (Evaluation section) and SVØMM (Validation section) is described below
with respect to input from cards or tape needed for these programs to operate
properly.   Following this is a description of the output from each program.
Finally errors and diagnostic messages and other machine dependent informa-
tion are set forth and sample input and output from the program is presented.

3.1   INPUT TO THE PROGRAM

INPUT: PROGRAM CAPSIS

The Creation section (Program CAPSIS) generates a data bank on magnetic
tape.   The user specifies (on punched cards) the number of data sets to be
stored, and an identifying set name delimiting each data set to be stored in
the bank.   These cards are read from the input file and are stored using the
routines from the referenced "DATA POOL" system.   The only restriction on
the input data is that the content of the data sets must conform to the
formats used by Program CGE (INTRAN Overlay).   After the tape is written,
it will contain all of the above input data and be ready for retrieval.   Non-
implemented capabilities of this program include adding or deleting data
sets or combining two or more data sets to form a new data set.   Currently,
no portions of data sets may be modified other than by replacement of the
entire set.   A list of input cards required and their contents (for creating
the data bank) is given below:

Cards for First Data Set

Card No.	Columns	Format	Description	Restriction	Sample Value
1	1- 2	I2	NDS = Number of data sets to store	NDS > 0	07
2	1-80	8A10	IT(I) = Identifying set name (I=1,7)	First 12 characters must be TABLEbNAMEb= ; set name may not exceed 10 non-blank (b) characters.	TABLEbNAMEb= LINKSURVEY
3 to M	1-80	8A10	Input (LINKSURVEY) data on (M-2) data cards	Must conform to Overlay INTRAN formats (M-2)	(See sample output tape of CAPSIS)
M+1	1-80	8A10	IT(I) = Terminator of data set (I = 1,7)	First 12 characters must be TABLEbNAMEb= ; Columns 13-80 must be blank (b).	TABLEbNAMEb= b . . . b

Cards 2 through M+1 are repeated using the succeeding set names until NDS sets have been specified and delimited.

INPUT: PROGRAM RAPSIS

The Retrieval section (Program RAPSIS) extracts data via user specified set names. Data may be retrieved from a previously generated data bank by first specifying the number of data sets to be retrieved and then cards specifying an identifying set name for each set required. The only restriction on these sets is that they must be retrieved in the order specified by the INTRAN Overlay (Program CGE). The retrieval data is written on a magnetic tape for use by Program CGE.

Card No.	Columns	Format	Description	Restriction	Sample Value
1	1- 2	I2	NDS - Number of data sets to be re- trieved	NDS > 0	07
2	1-10	A10	NUM(1) = 1st set name	NUM ≤ 10 characters	LINKSURVEY
.					
.					
.					
(NDS-1)	1-10	A10	NUM(NDS) = NDSth set name	NUM ≤ 10 characters	COCKPITMMS

Card 2 is repeated until all NDS set names have been specified.

INPUT: PROGRAM CGE

The following input description is partitioned according to CGE card input and tape input from the retrieval program. Both types of data are read by the INTRAN overlay.

CGE Card Input

Data Card 1

Format (16I5)

Column	Name	Description
1- 5	NT	Number of tasks
6-10	NLF	Input file number from which data link is retrieved
11-15	NSF	Input file number for standard position data
16-20	NAF	Input file number for angular limitation data
21-25	NTF	Input file number for task data
26-30	NCF	Input file number for cockpit control codes

Column	Name	Description
31-35	NPF	Input file number for cockpit interference planes
36-40	NOF	Input file number for optimization parameters
41-45	NNF	Input file number for normal distribution table (Input file numbers must agree with 6600 Control Card designations)

Data Cards 2-3

Format (16I5)

Column	Name	Description
1- 5	KT1	0 → Printout input data 1 → Do not print
6-10	KT2	0 → Do not print 1 → Print link dimensions
11-15	KT3	0 → Do not print 1 → Print task data
16-20	KT4	0 → Do not print 1 → Print standard angular position
21-25	KT5	0 → Do not print 1 → Print joint angle limits
26-30	KT6	0 → Do not print 1 → Print cockpit geometry and codes
31-35	KT7	0 → Standard link survey used 1 → Non-standard Survey Used
36-40	KT8	0 → Do not print 1 → Print standard position joint coordinates
41-45	KT9	0 → Do not print 1 → Print control codes (with respect to lumbar joint)
46-50	KT10	0 → Extended hand position tested 1 → Clenched hand position tested
51-55	KT11	Not used as input
56-60	KT12	0 → Do not print 1 → Print reach analysis output

D162-10127-1

34

Column	Name	Description
61-65	KT13	Not currently used
66-70	KT14	Not used as input
71-75	KT15	$\begin{cases} 0 \to \text{Determine new position for predefined line of sight} \\ 1 \to \text{Do not calculate new position} \end{cases}$
76-80	KT16	Not used as input
1- 5	KT17	Not currently used
6-10	KT18	Not currently used
11-15	KT19	Not currently used
16-20	KT20	Not currently used

KT17–KT20: Card 3

Data Cards 4-6

Format (8A10)

Column	Name	Description
1-80	COMENT	Three cards that contain a description of the computer run

Data Card 7

Format (8A10)

Column	Name	Description
1-10	OSERV	Name of alternate link survey (non-standard)

Data Cards 8-12

Format (8F10.0)

Column	Name	Description
1-10	PCT	Percentile value for each link of man-model
11-20		
21-30		
.		
.		
.		
71-80		

Data Card 13

Format (3E10.2, 6I5)

Column	Name	Description
1-10	ERR	Allowed error for Davidon minimization (LYNX)
11-20	SCALE	Penalty coefficient scale factor
21-30	ERC	Allowed error in satisfying each equality constraint
31-35	ISKIP	$\begin{cases} 0 \to \text{Optimize first position} \\ 1 \to \text{First position optimization to be bypassed} \end{cases}$
36-40	MAN	$\begin{cases} 0 \to \text{Suppress MAN1 output} \\ 1 \to \text{Allow optimization data output} \end{cases}$
41-45	IPOSE	$\begin{cases} 0 \to \text{MAN1's position already known} \\ 1 \to \text{MAN1's position to be calculated from input angles} \end{cases}$
46-50	NSTEPS	Step size parameter
51-55	NMJ	Maximum number of joints
56-60	IP	Denotes MAN1 error output option

Data Cards 14-23

Format (10F83)

Column	Name	Description
1- 8	CONST	Array of constant Euler angles and weighting coefficients
9-16		Each of these preferred angles and weights is associated with the 27 variable angles of the motion model objective function.
.		
.		
.		
65-72		
73-80		

CGE Tape Input

The following is a description of the format and order of data retrieved from tape, and read by INTRAN.

Link Data

902 Format (XI2/(X2A10,A8,I2,5F10.2))

Card Image #1

    Col 1- 2 NL - number of links

Card Image #2

    .
    .
    .

Card Image #NL + 1

    Col 1-28 LVØC = 28 character link description

    Col 29-30 LN = link number

    Col 31-40 LMEAN= mean link length

    Col 41-50 LSTD = link length standard deviation

    Col 51-60 MMEAN= mean mass

    Col 61-70 MSTD = mass standard deviation

    Col 71-80 MPCT = percent of link length from proximal
                 end of link to centroid

Normal Distribution Table F = .5 to F = 1.0

903 Format (X16F5.2)

    16 values/card image to get 4 card images with 51 values
       (the positive half of the table)

Task Data

905 Format (I3,7X,7A10/3A10,2F7.3/6F10.1/(I3,7X,7A10/3A10,2F7.3/6F10.1))

Card Image #1

Col 1- 3	TASKNØ	Task number
Col 4-10		Blank
Col 11-80	TDES	70 character task description

Card Image #2

Col 1-10	RHTC	Right hand terminal code
Col 11-20	LHTC	Left hand terminal code
Col 21-30	ETC	Eye terminal code
Col 31-37	TDUR	Task duration time
Col 38-44	THOLD	Holding time at end of task

Card Image #3

Col 1-10		
Col 11-20	RHØRT	Euler angle for right hand orientation
Col 21-30		
Col 31-40		
Col 41-50	LHØRT	Euler angle for left hand orientation
Col 51-60		

Standard Position Data

904 Format (I3/(3F10.1))

Card Image #1

Col 1- 3	NL	Number of links

Card Image #2 to Card Image # (NL + 1)

Col 1-10	THETA	
Col 11-20	PHI	Euler angles to define standard position
Col 21-30	PSI	

D162-10127-1

Angular Limitations

909 Format (I3/(6F10.1))

Card Image #1

                Col 1- 3   NL   Number of links

Card Image #2 to Card Image # (NL + 1)

Col 1-10	Minimum THETA	
Col 11-20	Maximum THETA	
Col 21-30	Minimum PHI	
Col 31-40	Maximum PHI	Euler angles
Col 41-50	Minimum PSI	
Col 51-60	Maximum PSI	

Cockpit Control Codes

906 Format (I3/(A10,3F10.3)

Card Image #1

                Col 1- 3   NCC   Number of cockpit control codes

Card Image #2 to Card Image # (NCC + 1)

Col 1-10	TCVOC		10 character description of control point in cockpit
Col 11-20	TCLOC	X	Euclidean coordinate location
Col 21-30		Y	with respect to cockpit eye
Col 31-40		Z	reference point

Cockpit Planes

907 Format (I2/4A10,I2/(9F8.3))

Card Image #1

                Col 1- 2   NPL   =   Number of planes in cockpit

Card Image #2

                Col 1-40   40 character description of plane

                Col 41-42   Number of vertices in plane

Card Image #3 (9F8.3)

> Col 1- 9   Euclidean coordinates of vertices
>
> Col 10-18   Three vertices per card - 9 values total
>
> Col 19-27   Repeated for as many vertices
>
> Repeat Cards #2 and #3 NPL times

INPUT: PROGRAM SVØMM

The following is a description of the input data required by the validation program:

Card No.	Columns	Format	Description	Restriction	Sample Value
1	1- 6	A6	Code = problem identifier	Code = 6HPRØBLM	PRØBLM
	7-16	A10	PN = problem description		WES80DD
	17-21	I5	N = Number of joint coordinates compared	N > 0	bbbb2
	22-26	I5	NR = Number of repetitions	NR > 0	bbb11
	27-31	I5	MX = Maximum order of matrix	MX = 12	bbb12
	32-36	I5	INP $\neq$ 1 suppress input data		bbbb0
2	1-72	12A6	FMT = Format of input data		(50X,8F10.2)
3 to (3+NR)	1-80	8F10.0	X(I,J) = input data		(See sample output)
			I = Number of joint coordinates compared		
			J = Number of repetitions		

4 + NR	1- 4	A4	Code = data iden- tifier	Code = 4HHYPØ	HYPØ
	11-80	7F10.0	SXM(I) I = 1,N BOEMAN-I data		

3.2 OUTPUT FROM THE PROGRAM

The output from each of the programs is described below. PROGRAMS CAPSIS (Storage) and RAPSIS (Retrieval) provide output on tape only. PROGRAM CGE (Cockpit Geometry Evaluation) provides output on both tape and the printer. PROGRAM SVØMM (Statistical Validation) provides printed output only.

PROGRAM CAPSIS

CAPSIS gives an inventory of the names, locations, lengths and contents of the tables (data sets) created on the file. The data set names used are in the following initial order:

        (1)   COCKPITMMS   (Cockpit Plane Vertex Locations)

        (2)   TASKSEQ   (Task Sequence Information)

        (3)   LINKSURVEY   (Link Survey Data)

        (4)   CONTRØLCØD   (Control Codes and Locations)

        (5)   ANGLELIMIT   (Upper and Lower Bounds on All Joint Angles)

        (6)   SPØSITIØN   (Standard position of all joint angles)

        (7)   NØRMALDIST   (Normal distribution table)

The file name used to store these data sets is "BPØØL" or "TAPE10".

PROGRAM RAPSIS

RAPSIS gives an inventory of the names, locations, lengths and contents of the retrieved tables written on a file. The data set names and their order is given by:

(1) LINKSURVEY

(2) NORMALDIST

(3) TASKSEQ

(4) SPØSITION

(5) ANGLELIMIT

(6) CONTRØLCØD

(7) COCKPITMMS

The file name used in retrieval of these data sets is "WØRK" or "TAPE11".

In both the storage and retrieval programs, control card language is used to

copy these output tapes to the printer.

PROGRAM CGE

Output from CGE100 is written upon four tape files and the printer.

The file names and their general contents are:

TAPE 12 = Intermediate output tape (written on by all primary overlays) ("BCD" records)

TAPE 13 = Backup tape (written on by MAN1 and OUTGO Overlays ("BCD" records)

TAPE 14 = MAN1 intermediate positional data during each task for use of OUTGO overlay ("Binary" records)

TAPE 16 = MAN1 intermediate positional data during each task for use by the INTERF overlay ("Binary" record)

TAPE 12 is used to collect all numerical output. It is read by ØUTGØ and

printed along with a descriptive and tabular information provided by the

OUTGO overlay. Parts of this tape are also to be used by the Validation

program as input.

TAPE 13 is the backup tape which yields the original output from MAN1 in

case of infeasibility conditions. The printed history of the evaluation

D162-10127-1

may be produced at the option of the user.

TAPE 14 contains joint locations and orientations of BOEMAN-I at each step of the task and is used by the OUTGO overlay to print out all positions prior to an infeasibility condition.

TAPE 16 also contains the joint locations of BOEMAN-I at each step of the task and is used by the INTERF overlay to test for seatback interference.

The printed output is in three parts: Input data, results of task processing, and summation data. It begins with an overall description of the run. The input data portion consists of seven tables:

1. Link Dimensions

This table provides an ordering of the thirty-six links that describe BOEMAN-I's physique. For each link name and number, the length in inches, the mass percentile, the weight in pounds, the centroid percentage distance from the proximal end and the distance between the link centroid and the proximal ends are printed.

2. Task Sequence

This table gives the components that make up each task in a flight mission (task sequence). Specified for each task is the right hand, left hand and eye aiming control codes, the duration of the task (in seconds), the duration between tasks, and the right and left hand orientation angles (Euler angles).

3. Standard Position

The standard orientation of each joint in Euler angles is presented in this table. The joint numbers are the same as the link numbers since each joint corresponds to the proximal end of that link.

4. Joint Angular Limits

This table describes the lower and upper bounds on each Euler angle of each joint. Joints with fixed Euler angles show the same value for the upper and lower bounds. The sign conventions of the angles, Theta, Phi, and Psi, are shown in Figure 9. A rule to be followed is that neither θ or φ may have negative signs at the same joint. In addition, if θ is zero, then that link does not deflect for any value of φ.

5. Cockpit Geometry

This table is in two parts. The first part describes the Cockpit Control Codes. Each control name (coded) and the Euclidean coordinates of that control are listed with the eye reference point as the origin of the coordinate system. The codes are abbreviations for controls according to their function. Appendix VIII lists the full control names with their abbreviations.

The second part provides a description of the cockpit planes by name and with a list of each vertex location in consecutive order. The vertex coordinates are also expressed with respect to the eye reference point.

6. Standard Position

For each joint, the Euclidean coordinates of the standard position are given. These coordinates are with respect to the computer's seat reference point (lumbar joint). In addition, the orientation angles of each joint are given in the standard position.
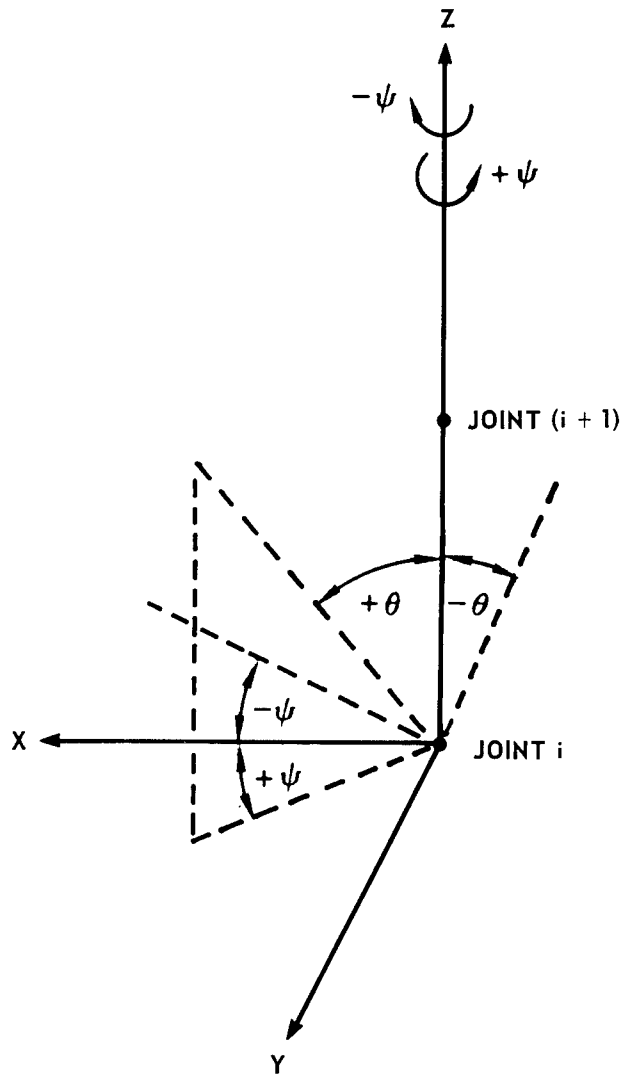
Figure 9. Euler Angle Coordinate System Sign Conventions

7. Cockpit Geometry

This table is the same as Table 5 except that the coordinates use the lumbar joint as the origin rather than the eye reference point.

Part II (task processing) provides the output from the REACHA, MAN1 and INTERF overlays. For each task performed feasibly (no interference), the output consists of BOEMAN's joint locations and orientations at the beginning and at the end of the task.

If the reach analysis deems the task infeasible, it is redefined. The new control locations for each hand are printed, with a corresponding "feasible" top of spine position. The distance from the redefined controls to the original controls are calculated and printed.

If, during the MAN1 optimization, the task becomes infeasible, all positions from the initial to the last feasible position are printed. Following this is a list of joint angles which have exceeded their angular limits. Then the distances of the palms from the final control point positions, along the straight line palm path, are printed.

If during the interference analysis, visual interference occurs, each plane and the intersection location with the line of sight is printed. The new line of sight (two end points) found to correct the blockage is also printed. If physical interference between BOEMAN-I and the seatback plane occurs, the plane and interfering links, the position number during the task, and the intersection point between the link and the seatback are printed.

Part three is the summation section. It provides data by task and cumulatively during the sequence on the joint and centroid displacement for each

joint and centroid, the mass-centroid products for each link (giving an indication of work done) with totals for all links. Joint angular deflections (twist only) in degrees and head and eye deflections (total angle) in degrees are also provided. All of the above assumes a feasible task since this section is written only if the task currently being processed is feasible.

PROGRAM SVØMM

SVØMM gives a statistical account of the comparison of BOEMAN-I and real pilot paths. The output begins with the pilot and task identification codes. The real pilot's input data, giving two out of the three coordinates (the third is assumed fixed) of a joint position on its path, repeated n times. The positions of the human may be printed at the option of the evaluator. Following the joint locations of the human is the corresponding man-model synthesized locations (1 repetition). The calculated covariance matrix for the pilot data and the inverse matrix are then recorded. Finally the computed F-value, with the associated degrees-of-freedom are printed. This output scheme is followed for each of the sequenced paths to be tested.

3.3   DIAGNOSTICS AND ERROR MESSAGES

Instead of error messages and diagnostics being interspersed with the printout, a BCD file containing all of the MAN1 calculations is dumped when an infeasible task occurs. Errors occurring in other overlays are usually fatal and the SCOPE 3.0 operating system would abnormally exit. When this occurs, an octal dump of memory is given for that overlay as well as a listing of any input tape (BCD) for error tracing. The only error message occurs in Program SVØMM when a singular matrix is inverted since the F-statistic cannot be calculated. The storage and retrieval programs do

not provide diagnostics except for the dumping of input and output files that are used.

3.4 SYSTEM REQUIREMENTS

These programs were written in CDC FORTRAN IV for the CDC 6600 using a Boeing-modified SCOPE 3.1 operating system. No unusual machine components are used other than disc and tape drives (1 input tape and 4 files on disc). The SCOPE 3.1 operating system possesses a flexible control card language which allows for memory, tape and file dumps which are used after an abnormal exit. The control cards for each program are described in the following sections.

PROGRAM CAPSIS

1.	SEQUENCE CARD	(Priority Card)
2.	JOB CARD	(Containing the job name, time estimate, field length estimate)
3.	ACCOUNT CARD	
4.	TAPE REQUEST CARD FOR OUTPUT TAPE "BPØØL"	
5.	COMPILE CARD RUN(S)	(Compile and list)
6.	SETCØRE	(Set all locations in memory to zero)
7.	EXECUTE CARD LGO	
8.	EXIT	(Exit from SCOPE 3.1 operating system)
9.	DMP (0,FL)	(Memory dump of program's field length FL))
10.	EØR	(End of record)

PROGRAM RAPSIS

Same as CAPSIS except:

4. TAPE REQUEST CARDS   a) Input tape "BPØØL"
                                       b) Output tape "WORK"

7.1 COPYCR (WORK,ØUTPUT,7)(Work tape is printed out)

REWIND (WORK)

UNLODE (WORK)

DROPFIL (WORK)

PROGRAM CGE

Same as CAPSIS except:

4. TAPE REQUEST CARD   TAPE 11 (≡ WØRK) as INPUT

7.1 & 9.1 REWIND (TAPE i)

7.2 & 9.2 COPYSBF (TAPE i, OUTPUT)   (TAPE i, i=12,13,14,16 as a file
                                          is printed out)

7.3 & 9.3 REWIND (TAPE i)

7.4 & 9.4 REQUEST (TAPES)

7.5 & 9.5 REWIND (TAPES)

7.6 & 9.6 COPYCF (TAPE i, TAPES)   (TAPE i as a file is written on
                                          magnetic tape)

7.7 & 9.7 REWIND (TAPES)

7.8 & 9.8 DECDUMP (,,,TAPES, 10)   (The magnetic tape is printed
                                          for comparison)

7.9 & 9.9 UNLØDE (TAPES)

          DROPFIL (TAPES)

PROGRAM SVØMM

Program SVØMM is the same as CAPSIS except no tapes are required (i.e.,

delete tape request card).

Accurate time estimates are not presently available for the execution of all programs. However, the storage and retrieval programs have executed (without compilation) in 16.3 and 15.4 seconds respectively. The Cockpit Geometry Evaluation Program would require about 25 minutes for a 7-task flight mission with each task requiring between 3 and 4 minutes. Compilation requires approximately 30 seconds and the other overlays require about 60 seconds for the entire run. The Validation Program has executed in less than 2 seconds per F-statistic. The CDC uses a 60-bit word for storing numbers, hence the calculations are extremely accurate. Positional coordinates and angular values are given to four decimal places. Cockpit geometry coordinates are accurate to one decimal place.

The most important program restriction is that the retrieved data tape must be written in a format acceptable to the INTRAN overlay. Other pertinent limitations in the present system include the use of no more than 20 tasks per sequence, 27 variable angles, 36 cockpit planes and 36 joints and links of BOEMAN-I.

3.5   SAMPLE INPUT AND OUTPUT

This section is concerned with the sample input and output from the Cockpit Geometry Evaluation Program.

SAMPLE INPUT

a)   Card Input

1	56	10 11	15 16	20 21	25 26	30 31	35 36	40 41	45

Card 1

1	11	11	11	11	11	11	11	11
NT	NLF	NSF	NAF	NTF	NCF	NPF	NØF	NNF

Card 2

Column numbers: 5 6 10 11 15 16 20 21 25 26 30 31 35 36 40 41 45 46 50 51 55 56 60 61 65 66 70 71 75 76 80

Row 1 punches: 1

Fields:
KT1 KT2 KT3 KT4 KT5 KT6 KT7 KT8 KT9 KT10 KT11 KT12 KT13 KT14 KT15 KT16

Card 3

Fields:
KT17 KT18 KT19 KT20

Card 4

```
This run is for the multimission simulator using the standard
link survey and




                    CØMENT (I), I = 1,8
```

Card 5

Using a task sequence consisting of 7 tasks

CØMENT (I), I = 9,16

Card 6

CØMENT (I), I - 17,24

1   10

OBSERV

Cards 8,9,10, 11

| | 10 | 11 | 20 | 21 | 30 | 31 | 40 | 41 | 50 | 51 | 60 | 61 | 70 | 71 | 80 |

Cards 8,9,10, 11 — card columns with values: 5. | 5. | 5. | 5. | 5. | 5. | 5. | 5.

PCT(I)   I = 1,32

Card 12

5. | 5. | 5. | 5.

PCT(33) | PCT(34) | PCT(35) | PCT(36)

Card 13

| 10 | 11 | 20 | 21 | 30 | 31 | 35 | 36 | 40 | 41 | 45 | 46 | 50 | 51 | 60 |

| 1.E-4 | 100. | .01 | 1 | 0 | 1 | 1 | 27 | 4 |
| ERR | SCALE | ERC | ISKIP | MAN | IPOSE | NSTEPS | NMJ | IP |

Card 14

	8 9	16 17	24 25	32 33	40 41	48 49	56 57	64 65	72 73	80
1	500.	200.	.0001	500.	100.	.0001	40.	.0001	10.	10.

Weighting Coefficients on Obj. Function

Card 15

10.	10.	.0001	100.	.0001	40.	.0001	10.	10.	10.

Card 16

10.	.0001	25.	.0001	25.	100.	.0001

CONST (I), I - 1,27

Card 16
(Contd)

Preferred Euler Angles

0.	0.	0.

D162-10127-1

54

0.	0.	0.	90.	0.	0.	45.	0.	-15.	0.

Card 17

Preferred Euler Angles

0.	0.	-90.	0.	0.	45.	0.	15.	0.	90.

Card 18

Preferred Euler Angles

90.	0.	0.	0.

Card 19

Preferred Euler Angles

CONST(I)   I - 28,54

Blank

Cards
20-23

CONST(I)   I - 61,100

b)   Tape Input (TAPE 11)

The following listing of TAPE 11 provides the user with the arrangement of the data in each data set:

THIS SET OF DATA CONTAINS THE MEAN AND STANDARD DEVIATION FOR BOTH LINK DIMENSIO
~" AND FOR LINK MASS AS WELL AS A PERCENTAGE FOR THE CENTROID LOCATION , FOR EAC
~. LINK ON THE BOEMAN FIGURE.

36

LUMBAR LINK	1	3.70	0.25	11.80	1.50	50.00
DUMMY THORACIC	2	0.00	0.00	0.00	0.00	0.00
THORACIC LINK	3	14.40	0.14	37.00	4.71	50.00
NECK LINK,VERTICAL	4	2.20	0.09	0.00	0.00	50.00
NECK LINK,HORIZONTAL	5	1.50	0.00	3.30	0.42	50.00
HEAD LINK	6	6.00	0.28	11.50	1.46	50.00
EYEMIDPOINT TO HEAD LINK	7	5.50	0.00	0.00	0.00	50.00
UNIT VECTOR FROM EYE	8	1.00	0.00	0.00	0.00	0.00
LEFT EYEBALL TO HEAD	9	5.50	0.00	0.00	0.00	50.00
RIGHT EYEBALL TO HEAD	10	5.50	0.00	0.00	0.00	50.00
INTERCLAVICULAR LINK,LEFT	11	1.00	0.00	0.00	0.00	50.00
INTERCLAVICULAR LINK,RIGHT	12	1.00	0.00	0.00	0.00	50.00
CLAVICULAR LINK,LEFT	13	6.60	0.25	4.90	0.63	50.00
CLAVICULAR LINK,RIGHT	14	6.60	0.25	4.90	0.63	50.00
DUMMY SHOULDER LINK,LEFT	15	0.00	0.00	0.00	0.00	50.00
DUMMY SHOULDER LINK,RIGHT	16	0.00	0.00	0.00	0.00	50.00
HUMERAL LINK,LEFT	17	11.90	0.41	5.20	0.67	46.10
HUMERAL LINK,RIGHT	18	11.90	0.41	5.20	0.67	46.10
RADICAL LINK,LEFT	19	10.70	0.34	2.90	0.38	42.50
RADICAL LINK,RIGHT	20	10.70	0.34	2.90	0.38	42.50
DUMMY HAND LINK,LEFT	21	0.00	0.00	0.00	0.00	0.00
DUMMY HAND LINK,RIGHT	22	0.00	0.00	0.00	0.00	0.00
HAND LINK (EXTENDED),LEFT	23	7.50	0.34	1.30	0.17	47.40
H D LINK (EXTENDED),RIGHT	24	7.50	0.34	1.30	0.17	47.40
HAND LINK (CLENCHED),LEFT	25	2.80	0.08	1.30	0.17	39.20
HAND LINK (CLENCHED),RIGHT	26	2.80	0.08	1.30	0.17	39.20
PELVIC LINK,LATERAL,LEFT	27	3.80	0.21	9.20	1.17	50.00
PELVIC LINK,LATERAL,RIGHT	28	3.80	0.21	9.20	1.17	50.00
FEMORAL LINK,LEFT	29	17.10	0.66	16.40	2.09	42.70
FEMORAL LINK,RIGHT	30	17.10	0.66	16.40	2.09	42.70
TIBIAL LINK,LEFT	31	16.10	0.71	7.50	0.96	40.40
TIBIAL LINK,RIGHT	32	16.10	0.71	7.50	0.96	40.40
FOOT LINK,LEFT	33	3.40	0.15	2.60	0.33	58.10
FOOT LINK,RIGHT	34	3.40	0.15	2.60	0.33	58.10
HEEL-TOE LINK,LEFT	35	10.50	0.45	0.00	0.00	50.00
HEEL-TOE LINK,RIGHT	36	10.50	0.45	0.00	0.00	50.00

THIS DATA SET CONTAINS THE ORDINATE VALUES OF THE NORMAL DISTRIBUTION FROM F=.50
TO F=1.00 NEEDED TO DETERMINE LINK LENGTH AND LINK MASS
0.00 0.03 0.05 0.08 0.10 0.13 0.15 0.18 0.20 0.23 0.25 0.28 0.31 0.33 0.36 0.39
0.41 0.44 0.47 0.50 0.53 0.55 0.58 0.61 0.64 0.67 0.71 0.74 0.77 0.81 0.84 0.8~
0.92 0.95 0.99 1.04 1.08 1.13 1.18 1.23 1.28 1.34 1.41 1.48 1.56 1.64 1.75 1.8~
J5 2.33 5.00

THIS SET OF DATA CONTAINS THE RIGHT HAND CONTROL CODE,THE LEFT HAND CONTROL COD
-E, THE EYE FOCAL POINT CODE,THE TASK DURATION IN SECONDS, THE TASK HOLDING TIME
,AND THE RIGHT AND LEFT HAND ORIENTATION ANGLES ,FOR EACH TASK IN THE SEQUENCE.
001   STANDARD POSITION (ANGULAR) TO STARTING POSITION (BOTH HANDS ON STICK)
FCCSC   FCCSC   FCCSC   1.000 1.000
        80.0   85.0   180.0   100.0   95.0   -180.0
002   START POSITION TO LEFT HAND ON THROTTLES,RIGHT HAND ON STICK
FCCSC   FCT   FCT   1.000 10.000
        80.0   85.0   180.0   90.0   110.0   -90.0
003   BRING LEFT HAND FROM THROTTLES TO STICK
FCCSC   FCCSC   FCCSC   1.000 0.001
        80.0   85.0   180.0   100.0   95.0   -180.0
004   BRING RIGHT HAND UP TO MASTER CAUTION BUTTON TO RESET IT
ACAMC   FCCSC   ACAMC   0.900 0.100
        100.0   90.0   110.0   100.0   95.0   -180.0
005   BRING RIGHT HAND TO UTILITY POWER CONTROL ON HYDRAULIC PANEL
MSHP   FCCSC   MSHP   1.500 0.500
        100.0   80.0   170.0   100.0   95.0   -180.0
10   BRING RIGHT HAND BACK TO STICK
FCCSC   FCCSC   FCCSC   1.000 0.001
        80.0   85.0   180.0   100.0   95.0   -180.0
007   BRING LEFT HAND BACK TO THROTTLES
FCCSC   FCT   FCT   1.000 10.000
        100.0   85.0   180.0   90.0   110.0   -90.0

THIS SET OF DATA CONTAINS BOEMANS STANDARD POSITION IN THE COCKPIT.

36	Θ	φ	ψ
	0.	90.	0.
	20.	90.	0.
	20.	90.	0.
	0.	0.	0.
	90.	90.	0.
	90.	90.	0.
	90.	90.	0.
	0.	90.	0.
	90.	120.	0.
	90.	60.	0.
	90.	180.	0.
	90.	0.	0.
	0.	90.	0.
	0.	90.	0.
	-30.	90.	0.
	-30.	90.	0.
	-90.	0.	30.
	90.	0.	30.
	90.	90.	0.
	90.	90.	0.
	10.	90.	0.
	10.	90.	0.
	-10.	90.	0.
	-10.	90.	0.
	-10.	90.	0.
	-10.	90.	0.
	118.165	180.	0.
	118.165	0.	0.
	90.	90.	-61.835
	90.	90.	61.835
	90.	-90.	0.
	90.	-90.	0.
	0.	90.	0.
	0.	90.	0.
	90.	90.	0.
	90.	90.	0.

THIS SET OF DATA CONTAINS THE MINIMUM AND MAXIMUM ANGLES OF JOINT EXCURSION
WITH RESPECT TO AN EULERIAN COORDINATE SYSTEM

36	MIN θ	MAX	MIN φ	MAX	MIN ψ	MAX
	-10.	10.	90.	90.	0.	0.
	20.	20.	90.	90.	0.	0.
	-40.	40.	0.	180.	-35.	35.
	0.	0.	0.	0.	0.	0.
	90.	90.	-90.	-90.	0.	0.
	30.	150.	0.	180.	-73.	73.
	90.	90.	90.	90.	0.	0.
	-61.	61.	0.	180.	0.	0.
	90.	90.	120.	120.	0.	0.
	90.	90.	60.	60.	0.	0.
	90.	90.	180.	180.	0.	0.
	90.	90.	0.	0.	0.	0.
	-10.	10.	0.	180.	0.	0.
	-10.	10.	0.	180.	0.	0.
	-30.	-30.	90.	90.	0.	0.
	-30.	-30.	90.	90.	0.	0.
	0.	-180.	0.	180.	-34.	97.
	0.	180.	0.	180.	-97.	34.
	0.	142.	90.	90.	-113.	77.
	0.	142.	90.	90.	-77.	113.
	10.	10.	90.	90.	0.	0.
	10.	10.	90.	90.	0.	0.
	-37.	37.	0.	180.	0.	0.
	-37.	37.	0.	180.	0.	0.
	-37.	37.	0.	180.	0.	0.
	-37.	37.	0.	180.	0.	0.
	118.165	118.165	180.	180.	0.	0.
	118.165	118.165	0.	0.	0.	0.
	48.	132.	0.	180.	-95.835	-22.835
	48.	132.	0.	180.	22.835	95.835
	0.	113.	-90.	-90.	-35.	43.
	0.	113.	-90.	-90.	-43.	35.
	-23.	23.	0.	180.	0.	0.
	-23.	23.	0.	180.	0.	0.
	90.	90.	90.	90.	0.	0.
	90.	90.	90.	90.	0.	0.

IS SET OF DATA CONTAINS A COCKPIT CONTROL CODE DICTIONARY REFERENCING CONTROL
NAMES IN THE MULTI-MISSION SIMULATOR WITH THEIR EUCLIDEAN COORDINATE LOCATIONS
WITH RESPECT TO THE COCKPIT EYE REFERENCE POINT.

091	X	Y	Z
FCFWS	-13.6	.25	-31.2
FCEPS	-11.45	-4.35	-31.2
FCRT	-15.8	3.9	-30.45
FCEPT	-15.7	7.5	-29.6
FCCSC	-.66	17.9	-23.2
FCSPS	-11.6	6.1	-30.4
FCT	-13.6	6.68	-30.2
FIRPML	-11.4	26.0	-17.0
FIRPMR	-9.0	26.0	-17.0
FIPT	-.24	16.8	-23.2
FIAT	-.24	16.8	-23.2
FIWSFP	-13.6	.25	-31.2
FIFT	-10.6	27.30	-11.0
FIFI	-10.6	26.82	-13.30
FITSI	-2.6	25.6	-19.2
FIAAT	-6.50	27.8	-8.6
FIAAA	-9.45	26.8	-8.94
FIHUC	0.	18.538	-6.150
FIHUD	0.	28.0	-6.15
FIAI	-6.91	26.0	-12.44
FIAS	-6.46	26.3	-16.3
MM	-6.0	25.6	-18.90
FIVSD	0.	26.36	-15.2
FIVSDSS	-3.50	26.9	-12.4
FIVSDH	3.50	26.9	-12.4
FIVSDI	3.50	25.77	-18.250
FIVSDC	-3.50	25.770	-18.250
FIC	6.1	27.65	-10.28
FIBA	6.48	26.80	-13.1
FIRA	6.55	26.30	-15.8
FIMC	10.2	26.8	-8.94
FIRC	9.50	26.80	-13.1
FIGM	9.00	26.32	-15.4
FIRMI	8.86	25.72	-18.4
FIHSD1	0.	23.0	-28.05
FIHSD2	0.	20.9	-31.9
FIHSDGC	-5.90	24.6	-20.84
FIHSDCI	5.90	24.6	-20.84
AFCSCP	13.75	13.7	-29.15
AFCSSP	10.6	19.30	-30.28
AFCSAP	14.25	19.30	-29.83
AFCADC	15.1	4.51	-31.1

MSESP	-12.50	5.13	-30.5
MSEPS	-15.7	8.4	-29.9
MSLCI	-13.91	-12.41	-32.6
MSLCE	-13.94	-9.2	-32.16
MSLGC	-18.02	20.587	-24.829
MSFMP	-14.50	4.35	-31.6
MSECP	18.3	-9.0	-30.0
MSCPI	18.25	20.5	-25.63
MSLOQ	18.25	23.3	-22.755
MSHP	20.45	-8.0	-28.03
MSAHC	11.90	26.753	-13.573
MSRAT	15.4	22.6	-24.08
MSEBI	15.78	24.6	-21.5
MSMLS	-11.5	8.43	-30.2
MSAS	-13.5	8.38	-29.9
DCNSCP	20.84	-4.6	-29.1
DCMRHAWI	9.44	27.45	-10.2
DCMLWI	10.66	27.16	-11.4
DCMEDSP	14.56	26.48	-14.8
ACAMC	3.533	27.283	-11.108
ACAEFL	-4.0	27.9	-16.14
ACAEFR	4.0	27.9	-16.14
ACALP	14.13	25.7	-18.38
ACLFW	-8.4	26.6	-9.42
ACAWW	-3.18	27.25	-11.04
ACALAW	3.80	25.0	-19.20
WSAIS	-18.5	-9.54	-30.2
WSMP	-19.0	-4.6	-29.10
WSASP	-19.5	16.75	-25.75
WSMAS	-19.08	22.10	-22.30
WSGS	-19.90	19.20	-25.05
WSWS	-15.56	22.48	-23.0
WSSPS	-18.0	-12.7	-31.1
WSSPAI	-18.03	-13.82	-31.15
WSEJ	-19.060	23.781	-21.745
WSLLLTV	14.0	21.0	-26.88
CNIMPS	-18.375	6.25	-28.6
CNIMSS	-21.375	3.22	-25.7
CNITS	-20.8	11.95	-25.0
CNIICSP	-19.5	15.2	-25.95
CNIRBC	19.48	16.56	-25.4
CNIDLR	12.65	24.6	-21.47
CNIDLC	13.30	24.0	-23.72
CNINMS	18.2	14.04	-27.2
CNILOS	20.84	5.22	-25.70
CNIDSS	22.3	5.2	-29.5
CNINDP	13.90	9.8	-29.64
CNIHTTD	6.050	25.70	-18.578
IMDS	12.75	-9.08	-32.3

THIS DATA SET CONTAINS EACH SET OF VERTICES CORRESPONDING TO EACH COCKPIT PLANE
IN THE MULTI-MISSION SIMULATOR COCKPIT

36	X	Y	Z		X	Y	Z		X	Y	Z

UPPER FRONT PANEL   06
   .7.720   25.349 -20.230 -17.720   26.817 -13.322 - 4.250   28.368 - 6.027
   4.250   28.368 - 6.027   17.720   26.817 -13.322   17.720   25.349 -20.230

LOWER FRONT PANEL (CENTER)   04
 - 2.250   24.540 -21.011 - 2.250   25.349 -20.230   2.250   25.349 -20.230
   2.250   24.540 -21.011

LOWER FRONT PANEL (LEFT CENTER)   05
 -16.770   23.101 -22.401 -16.770   25.349 -20.230 - 2.250   25.349 -20.230
 - 2.250   24.540 -21.011 - 4.500   23.101 -22.401

LOWER FRONT PANEL (RIGHT CENTER)   05
   4.500   23.101 -22.401   2.250   24.540 -21.011   2.250   25.349 -20.230
  16.770   25.349 -20.230   16.770   23.101 -22.401

LOWER FRONT PANEL (LEFT WING)   05
 -17.132   17.808 -27.859 -21.897   22.971 -22.537 -18.080   25.349 -20.230
 -16.770   25.349 -20.230 -16.770   17.808 -27.859

LOWER FRONT PANEL (RIGHT WING)   05
  16.770   17.808 -27.859   16.770   25.349 -20.230   18.080   25.349 -20.230
  21.897   22.971 -22.537   17.132   17.808 -27.859

LOWER FRONT PANEL (LOWER LEFT CENTER)   04
 -16.770   20.314 -25.093 -16.770   23.101 -22.401 -12.020   23.101 -22.401
 -12.020   20.314 -25.093

LOWER FRONT PANEL (LOWER RIGHT CENTER)   04
  12.020   20.314 -25.093   12.020   23.101 -22.401   16.770   23.101 -22.401
  16.770   20.314 -25.093

CIRCULAR SCOPE   06
   0.000   19.769 -32.343 - 4.630   21.376 -30.210 - 4.630   24.602 -25.930
   0.000   26.209 -23.797 - 4.630   24.602 -25.930   4.630   21.376 -30.210

HEAD UP DISPLAY PANEL (FRONT)   04
 - 2.000   18.188 - 8.229 - 2.000   19.071 - 4.071   2.000   19.071 - 4.071
   2.000   18.188 - 8.229

HEAD UP DISPLAY PANEL (LEFT SIDE)   04
 - 2.750   27.480 -10.204 - 2.750   28.368 - 6.027 - 2.000   19.071 - 4.071
 - 2.000   18.188 - 8.229

HEAD UP DISPLAY PANEL (TOP SIDE)   04
 - 2.000   19.071 - 4.071   2.000   19.071 - 4.071   2.750   28.368 - 6.027
 - 2.750   28.368 - 6.027

HEAD UP DISPLAY PANEL (RIGHT SIDE)   04
   2.000   19.071 - 4.071   2.750   28.368 - 6.027   2.750   27.480 -10.204
   2.000   18.188 - 8.229

HEAD UP DISPLAY PANEL (BOTTOM SIDE)   04
   2.000   18.188 - 8.229   2.750   27.480 -10.204 - 2.750   27.480 -10.204
 - 2.000   18.188 - 8.229

UPPER LEFT SIDE PANEL   04
 -17.125 -14.201 -32.230 -17.125   17.922 -28.001 -21.898   22.975 -22.522
 -21.899 -14.824 -27.498

LOWER LEFT SIDE PANEL   04
 -10.405 -14.080 -33.392 -10.426   16.904 -29.320 -17.074   17.887 -28.008
 -17.052 -14.238 -32.231

UPPER RIGHT SIDE PANEL   04
  17.125   17.711 -27.988   21.898   22.789 -22.505   21.898 -15.134 -27.498
  17.125 -14.511 -32.230

LOWER RIGHT SIDE PANEL   04
  10.431   16.604 -29.311   17.079   17.686 -27.987   17.058 -14.537 -32.222
  10.410 -14.320 -33.394

D162-10127-1

63

```
CONTROL STICK PLATFORM (TOP SIDE)              04
- 2.750   13.149 -28.230   2.750   13.149 -28.230   2.750   19.149 -28.230
- 2.750   19.149 -28.230
CONTROL STICK PLATFORM (LEFT SIDE)             04
- 2.750   13.149 -37.730 - 2.750   13.149 -28.230 - 2.750   19.149 -28.230
- 2.750   19.149 -37.730
CONTROL STICK PLATFORM (FRONT SIDE)            04
  2.750   13.149 -37.730   2.750   13.149 -28.230 - 2.750   13.149 -28.230
- 2.750   13.149 -37.730
CONTROL STICK PLATFORM (RIGHT SIDE)            04
  2.750   19.149 -37.730   2.750   19.149 -28.230   2.750   13.149 -28.230
  2.750   13.149 -37.730
CONTROL STICK PLATFORM (BACK SIDE)             04
- 2.750   19.149 -37.730 - 2.750   19.149 -28.230   2.750   19.149 -28.230
  2.750   19.149 -37.730
SEAT PLANE (LEFT SIDE)                         05
- 8.500 - 4.751 -37.730 - 8.500 - 4.751 -32.370 - 8.500   8.009 -31.090
- 8.500   8.609 -31.930 - 8.500   8.609 -37.730
SEAT PLANE (FRONT SIDE)                        04
- 8.500   8.609 -37.730 - 8.500   8.609 -31.930   8.500   8.609 -31.930
  8.500   8.609 -37.730
SEAT PLANE (RIGHT SIDE)                        05
  8.500   8.609 -37.730   8.500   8.609 -31.930   8.500   8.009 -31.090
  8.500 - 4.751 -32.370   8.500 - 4.751 -37.730
SEAT PLANE (FRONT-TOP BEVEL)                   04
- 8.500   8.609 -31.930 - 8.500   8.009 -31.090   8.500   8.009 -31.090
  8.500   8.609 -31.930
SEAT PLANE (TOP)                               04
- 8.500   8.009 -31.090 - 8.500 - 4.751 -32.370   8.500 - 4.751 -32.370
  8.500   8.009 -31.090
SEATBACK PLANE (LOWER FRONT)                   04
- 8.500 - 4.751 -32.370 - 8.500 -10.751 - 4.370   8.500 -10.751 - 4.370
  8.500 - 4.751 -32.370
SEATBACK PLANE (UPPER FRONT)                   04
- 8.500 -10.751 - 4.370 - 8.500 -10.751   3.630   8.500 -10.751   3.630
  8.500 -10.751 - 4.370
FLOOR                                          04
-21.897 -15.000 -37.730  21.897 -15.000 -37.730  21.897  30.000 -37.730
-21.897  30.000 -37.730
CEILING                                        04
-21.897 -15.000  10.000  21.897 -15.000  10.000  21.897  30.000  10.000
-21.897 -15.000  10.000
BACK WALL                                      04
-21.897 -15.000 -37.730 -21.897 -15.000  10.000  21.897 -15.000  10.000
 21.897 -15.000 -37.730
RIGHT SIDE WALL                                04
 21.897 -15.000 -37.730  21.897 -15.000  10.000  21.897  30.000  10.000
 21.897  30.000 -37.730
FRONT WALL                                     04
 21.897  30.000 -37.730  21.897  30.000  10.000 -21.897  30.000  10.000
-21.897  30.000 -37.730
LEFT SIDE WALL                                 04
-21.897  30.000 -37.730 -21.897  30.000  10.000 -21.897 -15.000  10.000
-21.897 -15.000 -37.730
```

SAMPLE ØUTPUT

The following listing gives the initial and final positions of BOEMAN-I during

Task 1. This task requires that he move from standard position to the

throttles. No interference is encountered and the task is deemed feasible

by the REACHA overlay.

COCKPIT GEOMETRY EVALUATION PROGRAM

BOEMAN   *   *   *   *   *   *   *   *   PHASE I

THIS RUN IS FOR THE MULTIMISSION SIMULATOR USING THE STANDARD LINK SURVEY AND
USING A TASK SEQUENCE CONSISTING OF 7 TASKS

1. LINK DIMENSIONS

LINK NAME	LINK PERCENTILE	LENGTH (INCHES)	PERCENTILE	MASS WEIGHT (POUNDS)	CENTROID DISTANCE FROM PROXIMAL END PERCENTAGE	LENGTH (INCHES)
1. LUMBAR LINK	5.0	3.3	5.0	9.3	50.0	1.6
2. DUMMY THORACIC	5.0	0.0	5.0	0.0	0.0	0.0
3. THORACIC LINK	5.0	14.2	5.0	29.3	50.0	7.1
4. NECK LINK,VERTICAL	5.0	2.0	5.0	0.0	50.0	1.0
5. NECK LINK,HORIZONTAL	5.0	1.5	5.0	2.6	50.0	.8
6. HEAD LINK	5.0	5.5	5.0	9.1	50.0	2.8
7. EYEMIDPOINT TO HEAD LINK	5.0	5.5	5.0	0.0	50.0	2.8
8. UNIT VECTOR FROM EYE	5.0	1.0	5.0	0.0	0.0	0.0
9. LEFT EYEBALL TO HEAD	5.0	5.5	5.0	0.0	50.0	2.8
10. RIGHT EYEBALL TO HEAD	5.0	5.5	5.0	0.0	50.0	2.8
11. INTERCLAVICULAR LINK,LEFT	5.0	1.0	5.0	0.0	50.0	.5
12. INTERCLAVICULAR LINK,RIGHT	5.0	1.0	5.0	0.0	50.0	.5
13. CLAVICULAR LINK,LEFT	5.0	6.2	5.0	3.9	50.0	3.1
14. CLAVICULAR LINK,RIGHT	5.0	6.2	5.0	3.9	50.0	3.1
15. DUMMY SHOULDER LINK,LEFT	5.0	0.0	5.0	0.0	50.0	0.0
16. DUMMY SHOULDER LINK,RIGHT	5.0	0.0	5.0	0.0	50.0	0.0
17. HUMERAL LINK,LEFT	5.0	11.2	5.0	4.1	46.1	5.2
18. HUMERAL LINK,RIGHT	5.0	11.2	5.0	4.1	46.1	5.2

19	RADICAL LINK,LEFT	5.0	10.1	5.0	2.3	5.0	42.5	4.3
20	RADICAL LINK,RIGHT	5.0	10.1	5.0	2.3	5.0	42.5	4.3
21	DUMMY HAND LINK,LEFT	5.0	0.0	5.0	0.0	5.0	0.0	0.0
22	DUMMY HAND LINK,RIGHT	5.0	0.0	5.0	0.0	5.0	0.0	0.0
23	HAND LINK (EXTENDED),LEFT	5.0	6.9	5.0	1.0	5.0	47.6	3.3
24	HAND LINK (EXTENDED),RIGHT	5.0	5.9	5.0	1.0	5.0	47.4	3.3
25	HAND LINK (CLENCHED),LEFT	5.0	2.7	5.0	0.1	5.0	39.2	0.1
26	HAND LINK (CLENCHED),RIGHT	5.0	2.7	5.0	1.0	5.0	39.2	1.0
27	PELVIC LINK,LATERAL,LEFT	5.0	3.5	5.0	7.3	5.0	50.0	1.7
28	PELVIC LINK,LATERAL,RIGHT	5.0	3.5	5.0	7.3	5.0	50.0	1.7
29	FEMORAL LINK,LEFT	5.0	16.0	5.0	13.0	5.0	42.7	6.8
30	FEMORAL LINK,RIGHT	5.0	15.0	5.0	13.0	5.0	42.7	6.8
31	TIBIAL LINK,LEFT	5.0	14.9	5.0	5.9	5.0	40.4	6.0
32	TIBIAL LINK,RIGHT	5.0	14.9	5.0	5.9	5.0	40.4	6.0
33	FOOT LINK,LEFT	5.0	3.2	5.0	2.1	5.0	58.1	1.8
34	FOOT LINK,RIGHT	5.0	3.2	5.0	2.1	5.0	58.1	1.8
35	HEEL-TOE LINK,LEFT	5.0	9.8	5.0	0.0	5.0	50.0	4.9
36	HEEL-TOE LINK,RIGHT	5.0	9.8	5.0	0.0	5.0	50.0	4.9

2. TASK SEQUENCE

TASK NO.	RIGHT HAND CONTROL	LEFT HAND CONTROL	EYE AIMING POINT	TASK DURATION (SEC)	DURATION BETWEEN TASKS (SEC)
1	FCCSC	FCCSC	FCCSC	1.000	1.000
2	FCCSC	FCT	FCT	1.000	10.000
3	FCCSC	FCCSC	FCCSC	1.000	.001
4	ACAMC	FCCSC	ACAMC	.900	.100
5	MSHP	FCCSC	MSHP	1.500	.500
6	FCCSC	FCCSC	FCCSC	1.000	.001
7	FCCSC	FCT	FCT	1.000	10.000

1

RIGHT HAND ORIENTATION   LEFT HAND ORIENTATION

(EULER ANGLES)   (EULER ANGLES)

(	80.0,	85.0,	180.0)	(	100.0,	95.0,	-180.0)
(	80.0,	85.0,	180.0)	(	90.0,	110.0,	-90.0)
(	80.0,	85.0,	180.0)	(	100.0,	95.0,	-180.0)
(	100.0,	90.0,	110.0)	(	100.0,	95.0,	-180.0)
(	100.0,	80.0,	170.0)	(	100.0,	95.0,	-180.0)
(	80.0,	85.0,	180.0)	(	100.0,	95.0,	-180.0)
(	100.0,	85.0,	180.0)	(	90.0,	110.0,	-90.0)

2

3. STANDARD POSITION

JOINT NO.	ORIENTATION ANGLES (EULER ANGLES)
1	(0.0, 90.0, 0.0)
2	(20.0, 90.0, 0.0)
3	(-20.0, 90.0, 0.0)
4	(0.0, 0.0, 0.0)
5	(90.0, -90.0, 0.0)
6	(90.0, 90.0, 0.0)
7	(90.0, 90.0, 0.0)
8	(0.0, 90.0, 0.0)
9	(90.0, 120.0, 0.0)
10	(90.0, 60.0, 0.0)
11	(90.0, 180.0, 0.0)
12	(90.0, 0.0, 0.0)
13	(0.0, 90.0, 0.0)
14	(0.0, 90.0, 0.0)
15	(-30.0, 90.0, 0.0)
16	(-30.0, 90.0, 0.0)
17	(-90.0, 0.0, 30.0)
18	(90.0, 0.0, -30.0)
19	(90.0, 90.0, 0.0)
20	(90.0, 90.0, 0.0)
21	(10.0, 90.0, 0.0)
22	(10.0, 90.0, 0.0)
23	(-10.0, 90.0, 0.0)
24	(-10.0, 90.0, 0.0)
25	(-10.0, 90.0, 0.0)
26	(-10.0, 90.0, 0.0)
27	(118.2, 180.0, 0.0)
28	(118.2, 0.0, 0.0)
29	(90.0, 90.0, -61.8)
30	(90.0, 90.0, 61.8)
31	(90.0, -90.0, 0.0)
32	(90.0, -90.0, 0.0)
33	(0.0, 90.0, 0.0)
34	(0.0, 90.0, 0.0)
35	(90.0, 90.0, 0.0)
36	(90.0, 90.0, 0.0)

4. JOINT ANGULAR LIMITS

JOINT NO.	THETA MIN	THETA MAX	PHI MIN	PHI MAX	PSI MIN	PSI MAX
1	-10.0	10.0	90.0	90.0	0.0	0.0
2	20.0	20.0	90.0	90.0	0.0	0.0
3	-40.0	40.0	0.0	180.0	-35.0	35.0
4	0.0	0.0	0.0	0.0	0.0	0.0
5	90.0	90.0	-90.0	-90.0	0.0	0.0
6	30.0	150.0	0.0	180.0	-73.0	73.0
7	90.0	90.0	90.0	90.0	0.0	0.0
8	-61.0	61.0	0.0	180.0	0.0	0.0
9	90.0	90.0	120.0	120.0	0.0	0.0
10	90.0	90.0	60.0	60.0	0.0	0.0
11	90.0	90.0	180.0	180.0	0.0	0.0
12	90.0	90.0	0.0	0.0	0.0	0.0
13	-10.0	10.0	0.0	180.0	0.0	0.0
14	-10.0	10.0	0.0	180.0	0.0	0.0
15	-30.0	-30.0	90.0	90.0	0.0	0.0
16	-30.0	-30.0	90.0	90.0	0.0	0.0
17	-180.0	0.0	0.0	180.0	-34.0	97.0
18	0.0	180.0	0.0	180.0	-97.0	34.0
19	0.0	142.0	90.0	90.0	-113.0	77.0
20	0.0	142.0	90.0	90.0	-77.0	113.0
21	10.0	10.0	90.0	90.0	0.0	0.0
22	10.0	10.0	90.0	90.0	0.0	0.0
23	-37.0	37.0	0.0	180.0	0.0	0.0
24	-37.0	37.0	0.0	180.0	0.0	0.0
25	-37.0	37.0	0.0	180.0	0.0	0.0
26	-37.0	37.0	0.0	180.0	0.0	0.0
27	118.2	118.2	180.0	180.0	0.0	0.0
28	118.2	118.2	0.0	0.0	0.0	0.0
29	48.0	132.0	0.0	180.0	-95.8	-22.8
30	48.0	132.0	0.0	180.0	22.8	95.8
31	0.0	113.0	-90.0	-90.0	-35.0	43.0
32	0.0	113.0	-90.0	-90.0	-43.0	35.0
33	-23.0	23.0	0.0	180.0	0.0	0.0
34	-23.0	23.0	0.0	180.0	0.0	0.0
35	90.0	90.0	90.0	90.0	0.0	0.0
36	90.0	90.0	90.0	90.0	0.0	0.0

5. COCKPIT GEOMETRY

COCKPIT DESIGN USED IS THE MULTI-MISSION SIMULATOR.
CONTROL LOCATION COORDINATES ARE WITH RESPECT TO THE
EYE REFERENCE POINT AND ARE IN INCHES.

A. COCKPIT CONTROL CODES

CONTROL NAME	LOCATION OF CONTROL		
FCFWS	(-13.600,	.250,	-31.200)
FCEPS	(-11.450,	-4.350,	-31.200)
FCRT	(-15.800,	3.900,	-30.450)
FCEPT	(-15.700,	7.500,	-29.600)
FCCSC	(-.660,	17.900,	-23.200)
FCSPS	(-11.600,	6.100,	-30.400)
FCI	(-13.600,	6.680,	-30.200)
FIRPML	(-11.400,	26.000,	-17.000)
FIRPMR	(-9.000,	26.000,	-17.000)
FIPT	(-.240,	16.800,	-23.200)
FIAT	(-.240,	16.800,	-23.200)
FIWSFP	(-13.600,	.250,	-31.200)
FIFT	(-10.600,	27.300,	-11.000)
FIFI	(-10.600,	26.820,	-13.300)
FITSI	(-2.600,	25.600,	-19.200)
FIAAT	(-6.500,	27.800,	-8.600)
FIAAA	(-9.450,	26.800,	-8.940)
FIHUC	(0.000,	18.538,	-6.150)
FIHUD	(0.000,	28.000,	-6.150)
FIAI	(-6.910,	26.000,	-12.440)
FIAS	(-6.460,	26.300,	-16.300)
FIMM	(-6.000,	25.600,	-18.900)
FIVSD	(0.000,	26.360,	-15.200)
FIVSDSS	(-3.500,	26.900,	-12.400)
FIVSDH	(3.500,	26.900,	-12.400)
FIVSDI	(3.500,	25.770,	-18.250)
FIVSDC	(-3.500,	25.770,	-18.250)
FIC	(6.100,	27.650,	-10.280)
FIBA	(6.480,	26.800,	-13.100)
FIHA	(6.550,	26.300,	-15.800)
FIMC	(10.200,	26.800,	-8.940)
FIRC	(9.500,	26.800,	-13.100)
FIGM	(9.000,	26.320,	-15.400)
FIRMI	(8.860,	25.720,	-18.400)
FIHSD1	(0.000,	23.000,	-28.050)
FIHSD2	(0.000,	20.900,	-31.900)
FIHSDGC	(-5.900,	24.600,	-20.840)
FIHSDCI	(5.900,	24.600,	-20.840)
AFCSCP	(13.750,	13.700,	-29.150)

```
AFCSSP          (  10.600,   19.300,  -30.280)
AFCSAP          (  14.250,   19.300,  -29.830)
AFCADC          (  15.100,    4.510,  -31.100)

MSESP           ( -12.500,    5.130,  -30.500)
MSEPS           ( -15.700,    8.400,  -29.900)
MSLCI           ( -13.900,  -12.480,  -32.600)
MSLCF           ( -13.940,   -9.200,  -32.160)
MSLGC           ( -18.020,   20.587,  -24.829)
MSFMP           ( -14.500,    4.350,  -31.600)
MSECP           (  18.300,   -9.000,  -30.000)
MSCPI           (  18.250,   20.500,  -25.630)
MSLOQ           (  18.250,   23.300,  -22.755)
MSHP            (  20.450,   -8.000,  -28.030)
MSAHC           (  11.900,   26.753,  -13.573)
MSRAT           (  15.400,   22.600,  -24.080)
MSERT           (  15.780,   24.600,  -21.500)
MSMLS           ( -11.500,    8.430,  -30.200)
MSAS            ( -13.500,    8.380,  -29.900)
DCMSCP          (  20.840,   -4.600,  -29.100)
DCMRHAWI        (   9.440,   27.450,  -10.200)
DCMLWI          (  10.660,   27.160,  -11.400)
DCMEDSP         (  14.560,   26.480,  -14.800)
ACAMC           (   3.533,   27.288,  -11.108)
ACAEFL          (  -4.000,   27.900,  -16.140)
ACAEFR          (   4.000,   27.900,  -16.140)
ACALP           (  14.130,   25.700,  -18.380)
ACALFW          (  -8.400,   26.600,   -9.420)
ACAWW           (  -3.180,   27.250,  -11.040)
ACALAW          (   3.800,   25.000,  -19.200)
WSATS           ( -18.500,   -9.540,  -30.200)
WSMP            ( -19.000,   -4.600,  -29.100)
WSASP           ( -19.500,   16.750,  -25.750)
WSMAS           ( -19.080,   22.100,  -22.300)
WSGS            ( -19.900,   19.200,  -25.050)
WSWS            ( -15.560,   22.480,  -23.000)
WSSPS           ( -18.000,  -12.700,  -31.100)
WSSPAI          ( -18.030,  -13.820,  -31.150)
WSEJ            ( -19.080,   23.781,  -21.745)
WSLLLTV         (  14.000,   21.000,  -26.880)
CNIMPS          ( -18.375,    6.250,  -28.600)
CNIMSS          ( -21.375,    3.220,  -25.700)
CNITS           ( -20.800,   11.950,  -25.000)
CNIICSP         ( -19.500,   15.200,  -25.950)
CNIRBC          (  19.480,   16.560,  -25.400)
CNIDLR          (  12.650,   24.600,  -21.470)
CNIDLC          (  13.300,   24.000,  -23.720)
CNINMS          (  18.200,   14.040,  -27.200)
CNILOS          (  20.840,    5.220,  -25.700)
CNIDSS          (  22.300,    5.200,  -29.500)
CNINDP          (  13.900,    9.800,  -29.640)
CNIHTTD         (   6.050,   25.700,  -18.578)
CNIMDS          (  12.750,   -9.080,  -32.300)
```

B. COCKPIT PLANES

THE PLANE VERTICES ARE WITH RESPECT TO THE EYE REFERENCE
POINT AND ARE IN (X,Y,Z) COORDINATES.

PLANE	VERTICES
UPPER FRONT PANEL	
	(-17.720, 25.349, -20.230)
	(-17.720, 26.817, -13.322)
	(-4.250, 28.368, -6.027)
	(4.250, 28.368, -6.027)
	(17.720, 26.817, -13.322)
	(17.720, 25.349, -20.230)
LOWER FRONT PANEL (CENTER)	
	(-2.250, 24.540, -21.011)
	(-2.250, 25.349, -20.230)
	(2.250, 25.349, -20.230)
	(2.250, 24.540, -21.011)
LOWER FRONT PANEL (LEFT CENTER)	
	(-16.770, 23.101, -22.401)
	(-16.770, 25.349, -20.230)
	(-2.250, 25.349, -20.230)
	(-2.250, 24.540, -21.011)
	(-4.500, 23.101, -22.401)
LOWER FRONT PANEL (RIGHT CENTER)	
	(4.500, 23.101, -22.401)
	(2.250, 24.540, -21.011)
	(2.250, 25.349, -20.230)
	(16.770, 25.349, -20.230)
	(16.770, 23.101, -22.401)
LOWER FRONT PANEL (LEFT WING)	
	(-17.132, 17.808, -27.859)
	(-21.897, 22.971, -22.537)
	(-18.080, 25.349, -20.230)
	(-16.770, 25.349, -20.230)
	(-16.770, 17.808, -27.859)
LOWER FRONT PANEL (RIGHT WING)	
	(16.770, 17.808, -27.859)
	(16.770, 25.349, -20.230)
	(18.080, 25.349, -20.230)
	(21.897, 22.971, -22.537)
	(17.132, 17.808, -27.859)
LOWER FRONT PANEL (LOWER LEFT CENTER)	
	(-16.770, 20.314, -25.093)
	(-16.770, 23.101, -22.401)
	(-12.020, 23.101, -22.401)
	(-12.020, 20.314, -25.093)

LOWER FRONT PANEL (LOWER RIGHT CENTER)

```
(   12.020,   20.314,  -25.093)
(   12.020,   23.101,  -22.401)

(   16.770,   23.101,  -22.401)
(   16.770,   20.314,  -25.093)
```

CIRCULAR SCOPE

```
(    0.000,   19.769,  -32.343)
(   -4.630,   21.376,  -30.210)
(   -4.630,   24.602,  -25.930)
(    0.000,   26.209,  -23.797)
(   -4.630,   24.602,  -25.930)
(    4.630,   21.376,  -30.210)
```

HEAD UP DISPLAY PANEL (FRONT)

```
(   -2.000,   18.188,   -8.229)
(   -2.000,   19.071,   -4.071)
(    2.000,   19.071,   -4.071)
(    2.000,   18.188,   -8.229)
```

HEAD UP DISPLAY PANEL (LEFT SIDE)

```
(   -2.750,   27.480,  -10.204)
(   -2.750,   28.368,   -6.027)
(   -2.000,   19.071,   -4.071)
(   -2.000,   18.188,   -8.229)
```

HEAD UP DISPLAY PANEL (TOP SIDE)

```
(   -2.000,   19.071,   -4.071)
(    2.000,   19.071,   -4.071)
(    2.750,   28.368,   -6.027)
(   -2.750,   28.368,   -6.027)
```

HEAD UP DISPLAY PANEL (RIGHT SIDE)

```
(    2.000,   19.071,   -4.071)
(    2.750,   28.368,   -6.027)
(    2.750,   27.480,  -10.204)
(    2.000,   18.188,   -8.229)
```

HEAD UP DISPLAY PANEL (BOTTOM SIDE)

```
(    2.000,   18.188,   -8.229)
(    2.750,   27.480,  -10.204)
(   -2.750,   27.480,  -10.204)
(   -2.000,   18.188,   -8.229)
```

UPPER LEFT SIDE PANEL

```
(  -17.125,  -14.201,  -32.230)
(  -17.125,   17.922,  -28.001)
(  -21.898,   22.975,  -22.522)
(  -21.899,  -14.824,  -27.498)
```

LOWER LEFT SIDE PANEL

```
(  -10.405,  -14.080,  -33.392)
(  -10.426,   16.904,  -29.320)
(  -17.074,   17.887,  -28.008)
(  -17.052,  -14.238,  -32.231)
```

UPPER RIGHT SIDE PANEL

```
(   17.125,   17.711,  -27.988)
(   21.898,   22.789,  -22.505)
(   21.898,  -15.134,  -27.498)
(   17.125,  -14.511,  -32.230)
```

LOWER RIGHT SIDE PANEL

```
(  10.431,   16.604,  -29.311)
(  17.079,   17.686,  -27.987)
(  17.058,  -14.537,  -32.222)
(  10.410,  -14.380,  -33.384)
```

CONTROL STICK PLATFORM (TOP SIDE)

```
(  -2.750,   13.149,  -28.230)
(   2.750,   13.149,  -28.230)

(   2.750,   19.149,  -28.230)
(  -2.750,   19.149,  -28.230)
```

CONTROL STICK PLATFORM (LEFT SIDE)

```
(  -2.750,   13.149,  -37.730)
(  -2.750,   13.149,  -28.230)
(  -2.750,   19.149,  -28.230)
(  -2.750,   19.149,  -37.730)
```

CONTROL STICK PLATFORM (FRONT SIDE)

```
(   2.750,   13.149,  -37.730)
(   2.750,   13.149,  -28.230)
(  -2.750,   13.149,  -28.230)
(  -2.750,   13.149,  -37.730)
```

CONTROL STICK PLATFORM (RIGHT SIDE)

```
(   2.750,   19.149,  -37.730)
(   2.750,   19.149,  -28.230)
(   2.750,   13.149,  -28.230)
(   2.750,   13.149,  -37.730)
```

CONTROL STICK PLATFORM (BACK SIDE)

```
(  -2.750,   19.149,  -37.730)
(  -2.750,   19.149,  -28.230)
(   2.750,   19.149,  -28.230)
(   2.750,   19.149,  -37.730)
```

SEAT PLANE (LEFT SIDE)

```
(  -8.500,   -4.751,  -37.730)
(  -8.500,   -4.751,  -32.370)
(  -8.500,    8.009,  -31.090)
(  -8.500,    8.609,  -31.930)
(  -8.500,    8.609,  -37.730)
```

SEAT PLANE (FRONT SIDE)

```
(  -8.500,    8.609,  -37.730)
(  -8.500,    8.609,  -31.930)
(   8.500,    8.609,  -31.930)
(   8.500,    8.609,  -37.730)
```

SEAT PLANE (RIGHT SIDE)

```
(   8.500,    8.609,  -37.730)
(   8.500,    8.609,  -31.930)
(   8.500,    8.009,  -31.090)
(   8.500,   -4.751,  -32.370)
(   8.500,   -4.751,  -37.730)
```

SEAT PLANE (FRONT-TOP BEVEL)

```
(  -8.500,    8.609,  -31.930)
(  -8.500,    8.009,  -31.090)
(   8.500,    8.009,  -31.090)
(   8.500,    8.609,  -31.930)
```

SEAT PLANE (TOP)

```
(  -8.500,    8.009,  -31.090)
(  -8.500,   -4.751,  -32.370)
(   8.500,   -4.751,  -32.370)
(   8.500,    8.009,  -31.090)
```

SEATBACK PLANE (LOWER FRONT)

```
(  -8.500,   -4.751,  -35.370)
(  -8.500,  -10.751,   -4.370)
(   8.500,  -10.751,   -4.370)
(   8.500,   -4.751,  -35.370)
```

SEATBACK PLANE (UPPER FRONT)

```
(  -8.500,  -10.751,   -4.370)

(  -8.500,  -10.751,    3.630)
(   8.500,  -10.751,    3.630)
(   8.500,  -10.751,   -4.370)
```

FLOOR

```
( -21.897,  -15.000,  -37.730)
(  21.897,  -15.000,  -37.730)
(  21.897,   30.000,  -37.730)
( -21.897,   30.000,  -37.730)
```

CEILING

```
( -21.897,  -15.000,   10.000)
(  21.897,  -15.000,   10.000)
(  21.897,   30.000,   10.000)
( -21.897,  -15.000,   10.000)
```

BACK WALL

```
( -21.897,  -15.000,  -37.730)
( -21.897,  -15.000,   10.000)
(  21.897,  -15.000,   10.000)
(  21.897,  -15.000,  -37.730)
```

RIGHT SIDE WALL

```
(  21.897,  -15.000,  -37.730)
(  21.897,  -15.000,   10.000)
(  21.897,   30.000,   10.000)
(  21.897,   30.000,  -37.730)
```

FRONT WALL

```
(  21.897,   30.000,  -37.730)
(  21.897,   30.000,   10.000)
( -21.897,   30.000,   10.000)
( -21.897,   30.000,  -37.730)
```

LEFT SIDE WALL

```
( -21.897,   30.000,  -37.730)
( -21.897,   30.000,   10.000)
( -21.897,  -15.000,   10.000)
( -21.897,  -15.000,  -37.730)
```

INPUT DATA
CONTINUED

THE SEAT REFERENCE POINT IS LOCATED AT (0.000,   -4.000, -25.054)
WITH RESPECT TO THE EYE REFERENCE POINT.

6. STANDARD POSITION

JOINT NO.	(X,Y,Z) COORDINATES
1	(0.0, 0.0, 3.3)
2	(0.0, 0.0, 3.3)
3	(-0.0, 0.0, 17.5)
4	(-0.0, 0.0, 19.5)
5	(-0.0, -1.5, 19.5)
6	(-0.0, -1.5, 25.0)
7	(-0.0, 4.0, 25.0)
8	(-0.0, 5.0, 25.0)
9	(-2.8, 4.0, 25.0)
10	(2.8, 4.0, 25.0)
11	(-1.0, 0.0, 17.5)
12	(1.0, 0.0, 17.5)
13	(-7.2, 0.0, 17.5)
14	(7.2, 0.0, 17.5)
15	(-7.2, 0.0, 17.5)
16	(7.2, 0.0, 17.5)
17	(-7.2, 0.0, 6.2)
18	(7.2, 0.0, 6.2)
19	(-7.2, 10.1, 6.2)
20	(7.2, 10.1, 6.2)
21	(-7.2, 10.1, 6.2)
22	(7.2, 10.1, 6.2)
23	(-7.2, 17.1, 6.2)
24	(7.2, 17.1, 6.2)
25	(-7.2, 12.8, 6.2)
26	(7.2, 12.8, 6.2)
27	(-3.0, -0.0, -1.6)
28	(3.0, 0.0, -1.6)
29	(-3.0, 16.0, -1.6)
30	(3.0, 16.0, -1.6)
31	(-3.0, 16.0, -16.6)
32	(3.0, 16.0, -16.6)
33	(-3.0, 16.0, -19.7)
34	(3.0, 16.0, -16.6)
35	(-3.0, 25.8, -19.7)
36	(3.0, 16.0, -16.6)

INPUT DATA
CONTINUED

7. COCKPIT GEOMETRY

CONTROL LOCATION COORDINATES ARE WITH RESPECT TO THE SEAT
REFERENCE POINT AND ARE IN INCHES.

A. COCKPIT CONTROL CODES

CONTROL NAME (CODE)	LOCATION OF CONTROL (X,Y,Z) COORDINATES		
FCFWS	(-13.600,	4.250,	-6.146)
FCEPS	(-11.450,	-.350,	-6.146)
FCRT	(-15.800,	7.900,	-5.396)
FCEPT	(-15.700,	11.500,	-4.546)
FCCSC	(-.660,	21.900,	1.854)
FCSPS	(-11.600,	10.100,	-5.346)
FCT	(-13.600,	10.680,	-5.146)
FIRPML	(-11.400,	30.000,	8.054)
FIRPMR	(-9.000,	30.000,	8.054)
FIPT	(-.240,	20.800,	1.854)
FIAT	(-.240,	20.800,	1.854)
FIWSFP	(-13.600,	4.250,	-6.146)
FIFT	(-10.600,	31.300,	14.054)
FIFI	(-10.600,	30.820,	11.754)
FITSI	(-2.600,	29.600,	5.854)
FIAAT	(-6.500,	31.800,	16.454)
FIAAA	(-9.450,	30.800,	16.114)
FIHUC	(0.000,	22.538,	18.904)
FIHUD	(0.000,	32.000,	18.904)
FIAI	(-6.910,	30.000,	12.614)
FIAS	(-6.460,	30.300,	8.754)
FIMM	(-6.000,	29.600,	6.154)
FIVSD	(0.000,	30.360,	9.854)
FIVSDSS	(-3.500,	30.900,	12.654)
FIVSDH	(3.500,	30.900,	12.654)
FIVSDI	(3.500,	29.770,	6.804)
FIVSDC	(-3.500,	29.770,	6.804)
FIC	(6.100,	31.650,	14.774)
FIBA	(6.480,	30.800,	11.954)
FIRA	(6.550,	30.300,	9.254)
FIMC	(10.200,	30.800,	16.114)
FIRC	(9.500,	30.800,	11.954)
FIGM	(9.000,	30.320,	9.654)
FIRMI	(8.860,	29.720,	6.654)
FIHSD1	(0.000,	27.000,	-2.996)
FIHSD2	(0.000,	24.900,	-6.846)
FIHSDGC	(-5.900,	28.600,	4.214)
FIHSDCI	(5.900,	28.600,	4.214)
AFCSCP	(13.750,	17.700,	-4.096)
AFCSSP	(10.600,	23.300,	-5.226)
AFCSAP	(14.250,	23.300,	-4.776)
AFCADC	(15.100,	8.510,	-6.046)
MSESP	(-12.500,	9.130,	-5.446)

```
MSEPS          ( -15.700,   12.400,   -4.846)
MSLCI          ( -13.900,   -8.480,   -7.546)
MSLCE          ( -13.940,   -5.200,   -7.106)
MSLGC          ( -18.020,   24.587,    .225)
MSFMP          ( -14.500,    8.350,   -5.546)
MSECP          (  18.300,   -5.000,   -4.946)
MSCPI          (  18.250,   24.500,    -.576)
MSLOQ          (  18.250,   27.300,   2.299)
MSHP           (  20.450,   -4.000,   -2.976)
MSAHC          (  11.900,   30.753,   11.481)
MSRAT          (  15.400,   26.600,    .974)
MSEBT          (  15.780,   28.600,   3.554)
MSMLS          ( -11.500,   12.430,   -5.146)
MSAS           ( -13.500,   12.380,   -4.846)
DCMSCP         (  20.840,    -.600,   -4.046)
DCMRHAWI       (   9.440,   31.450,   14.854)
DCMLWI         (  10.660,   31.160,   13.654)
DCMEDSP        (  14.560,   30.480,   10.254)
ACAMC          (   3.533,   31.288,   13.046)
ACAEFL         (  -4.000,   31.900,   8.914)
ACAEFR         (   4.000,   31.900,   8.914)
ACALP          (  14.130,   29.700,   6.674)
ACALFW         (  -8.400,   30.600,   15.634)

ACAWW          (  -3.180,   31.250,   14.014)
ACALAW         (   3.800,   29.000,   5.854)
WSAIS          ( -18.500,   -5.540,   -5.146)
WSMP           ( -19.000,    -.600,   -4.046)
WSASP          ( -19.500,   20.750,    -.696)
WSMAS          ( -19.080,   26.100,   2.754)
WSGS           ( -19.900,   23.200,    .004)
WSWS           ( -15.560,   26.480,   2.054)
WSSPS          ( -18.000,   -8.700,   -6.046)
WSSPAI         ( -18.030,   -9.820,   -6.096)
WSEJ           ( -19.080,   27.781,   3.309)
WSLLLTV        (  14.000,   25.000,   -1.826)
CNIMPS         ( -18.375,   10.250,   -3.546)
CNIMSS         ( -21.375,    7.220,    -.646)
CNITS          ( -20.800,   15.950,    .054)
CNIICSP        ( -19.500,   19.200,    -.896)
CNIRRC         (  19.480,   20.560,    -.346)
CNIDLR         (  12.650,   28.600,   3.584)
CNIDLC         (  13.300,   28.000,   1.334)
CNINMS         (  18.200,   18.040,   -2.146)
CNILOS         (  20.840,    9.220,    -.646)
CNIDSS         (  22.300,    9.200,   -4.446)
CNINDP         (  13.900,   13.800,   -4.586)
CNIHTTD        (   6.050,   29.700,   6.476)
CNIMDS         (  12.750,   -5.080,   -7.246)
```

B. COCKPIT PLANES

THE PLANE VERTICES ARE WITH RESPECT TO THE SEAT REFERENCE POINT
AND ARE IN (X,Y,Z) COORDINATES,

PLANE	VERTICES
UPPER FRONT PANEL	(-17.720, 29.349, 4.824)
	(-17.720, 30.817, 11.732)
	(-4.250, 32.368, 19.027)
	(4.250, 32.368, 19.027)
	(17.720, 30.817, 11.732)
	(17.720, 29.349, 4.824)
LOWER FRONT PANEL (CENTER)	(-2.250, 28.540, 4.043)
	(-2.250, 29.349, 4.824)
	(2.250, 29.349, 4.824)
	(2.250, 28.540, 4.043)
LOWER FRONT PANEL (LEFT CENTER)	(-16.770, 27.101, 2.653)
	(-16.770, 29.349, 4.824)
	(-2.250, 29.349, 4.824)
	(-2.250, 28.540, 4.043)
	(-4.500, 27.101, 2.653)
LOWER FRONT PANEL (RIGHT CENTER)	(4.500, 27.101, 2.653)
	(2.250, 28.540, 4.043)
	(2.250, 29.349, 4.824)
	(16.770, 29.349, 4.824)
	(16.770, 27.101, 2.653)
LOWER FRONT PANEL (LEFT WING)	(-17.132, 21.808, -2.805)
	(-21.897, 26.971, 2.517)
	(-18.080, 29.349, 4.824)
	(-16.770, 29.349, 4.824)
	(-16.770, 21.808, -2.805)
LOWER FRONT PANEL (RIGHT WING)	(16.770, 21.808, -2.805)
	(16.770, 29.349, 4.824)
	(18.080, 29.349, 4.824)
	(21.897, 26.971, 2.517)
	(17.132, 21.808, -2.805)
LOWER FRONT PANEL (LOWER LEFT CENTER)	(-16.770, 24.314, .039)
	(-16.770, 27.101, 2.653)
	(-12.020, 27.101, 2.653)
	(-12.020, 24.314, .039)

LOWER FRONT PANEL (LOWER RIGHT CENTER)

```
(   12.020,   24.314,    -.039)
(   12.020,   27.101,    5.653)
(   16.770,   27.101,    5.653)
(   16.770,   24.314,    -.039)
```

CIRCULAR SCOPE

```
(    0.000,   23.769,   -7.289)
(   -4.630,   25.376,   -5.156)
(   -4.630,   28.602,    -.876)
(    0.000,   30.209,    1.257)
(   -4.630,   28.602,    -.876)
(    4.630,   25.376,   -5.156)
```

HEAD UP DISPLAY PANEL (FRONT)

```
(   -2.000,   22.188,   16.825)
(   -2.000,   23.071,   20.983)
(    2.000,   23.071,   20.983)
(    2.000,   22.188,   16.825)
```

HEAD UP DISPLAY PANEL (LEFT SIDE)

```
(   -2.750,   31.480,   14.850)
(   -2.750,   32.368,   19.027)
(   -2.000,   23.071,   20.983)
(   -2.000,   22.188,   16.825)
```

HEAD UP DISPLAY PANEL (TOP SIDE)

```
(   -2.000,   23.071,   20.983)
(    2.000,   23.071,   20.983)
(    2.750,   32.368,   19.027)
(   -2.750,   32.368,   19.027)
```

HEAD UP DISPLAY PANEL (RIGHT SIDE)

```
(    2.000,   23.071,   20.983)
(    2.750,   32.368,   19.027)
(    2.750,   31.480,   14.850)
(    2.000,   22.188,   16.825)
```

HEAD UP DISPLAY PANEL (BOTTOM SIDE)

```
(    2.000,   22.188,   16.825)
(    2.750,   31.480,   14.850)
(   -2.750,   31.480,   14.850)
(   -2.000,   22.188,   16.825)
```

UPPER LEFT SIDE PANEL

```
(  -17.125,  -10.201,   -7.176)
(  -17.125,   21.922,   -2.947)
(  -21.898,   26.975,    2.532)
(  -21.899,  -10.824,   -2.444)
```

LOWER LEFT SIDE PANEL

```
(  -10.405,  -10.080,   -8.338)
(  -10.426,   20.904,   -4.266)
(  -17.074,   21.887,   -2.954)
(  -17.052,  -10.238,   -7.177)
```

UPPER RIGHT SIDE PANEL

```
(   17.125,   21.711,   -2.934)
(   21.898,   26.789,    2.549)
(   21.898,  -11.134,   -2.444)
(   17.125,  -10.511,   -7.176)
```

LOWER RIGHT SIDE PANEL

```
(   10.431,   20.604,   -4.257)
(   17.079,   21.686,   -2.933)
(   17.058,  -10.537,   -7.168)
(   10.410,  -10.380,   -8.330)
```

CONTROL STICK PLATFORM (TOP SIDE)

```
(  -2.750,   17.149,   -3.176)
(   2.750,   17.149,   -3.176)
(   2.750,   23.149,   -3.176)

(  -2.750,   23.149,   -3.176)
```

CONTROL STICK PLATFORM (LEFT SIDE)

```
(  -2.750,   17.149,  -12.676)
(  -2.750,   17.149,   -3.176)
(  -2.750,   23.149,   -3.176)
(  -2.750,   23.149,  -12.676)
```

CONTROL STICK PLATFORM (FRONT SIDE)

```
(   2.750,   17.149,  -12.676)
(   2.750,   17.149,   -3.176)
(  -2.750,   17.149,   -3.176)
(  -2.750,   17.149,  -12.676)
```

CONTROL STICK PLATFORM (RIGHT SIDE)

```
(   2.750,   23.149,  -12.676)
(   2.750,   23.149,   -3.176)
(   2.750,   17.149,   -3.176)
(   2.750,   17.149,  -12.676)
```

CONTROL STICK PLATFORM (BACK SIDE)

```
(  -2.750,   23.149,  -12.676)
(  -2.750,   23.149,   -3.176)
(   2.750,   23.149,   -3.176)
(   2.750,   23.149,  -12.676)
```

SEAT PLANE (LEFT SIDE)

```
(  -8.500,    -.751,  -12.676)
(  -8.500,    -.751,   -7.316)
(  -8.500,   12.009,   -6.036)
(  -8.500,   12.609,   -6.876)
(  -8.500,   12.609,  -12.676)
```

SEAT PLANE (FRONT SIDE)

```
(  -8.500,   12.609,  -12.676)
(  -8.500,   12.609,   -6.876)
(   8.500,   12.609,   -6.876)
(   8.500,   12.609,  -12.676)
```

SEAT PLANE (RIGHT SIDE)

```
(   8.500,   12.609,  -12.676)
(   8.500,   12.609,   -6.876)
(   8.500,   12.009,   -6.036)
(   8.500,    -.751,   -7.316)
(   8.500,    -.751,  -12.676)
```

SEAT PLANE (FRONT-TOP BEVEL)

```
(  -8.500,   12.609,   -6.876)
(  -8.500,   12.009,   -6.036)
(   8.500,   12.009,   -6.036)
(   8.500,   12.609,   -6.876)
```

SEAT PLANE (TOP)

```
(  -8.500,   12.009,   -6.036)
(  -8.500,    -.751,   -7.316)
(   8.500,    -.751,   -7.316)
(   8.500,   12.009,   -6.036)
```

SEATBACK PLANE (LOWER FRONT)

```
(  -8.500,   -.751,  -7.316)
(  -8.500,  -6.751,  20.684)
(   8.500,  -6.751,  20.684)
(   8.500,   -.751,  -7.316)
```

SEATBACK PLANE (UPPER FRONT)

```
(  -8.500,  -6.751,  20.684)
(  -8.500,  -6.751,  28.684)
(   8.500,  -6.751,  28.684)
(   8.500,  -6.751,  20.684)
```

FLOOR

```
( -21.897, -11.000, -12.676)
(  21.897, -11.000, -12.676)
(  21.897,  34.000, -12.676)
( -21.897,  34.000, -12.676)
```

CEILING

```
( -21.897, -11.000,  35.054)
(  21.897, -11.000,  35.054)
(  21.897,  34.000,  35.054)
( -21.897, -11.000,  35.054)
```

BACK WALL

```
( -21.897, -11.000, -12.676)
( -21.897, -11.000,  35.054)
(  21.897, -11.000,  35.054)
(  21.897, -11.000, -12.676)
```

RIGHT SIDE WALL

```
(  21.897, -11.000, -12.676)
(  21.897, -11.000,  35.054)
(  21.897,  34.000,  35.054)
(  21.897,  34.000, -12.676)
```

FRONT WALL

```
(  21.897,  34.000, -12.676)
(  21.897,  34.000,  35.054)
( -21.897,  34.000,  35.054)
( -21.897,  34.000, -12.676)
```

LEFT SIDE WALL

```
( -21.897,  34.000, -12.676)
( -21.897,  34.000,  35.054)
( -21.897, -11.000,  35.054)
( -21.897, -11.000, -12.676)
```

PART 2

PROCESSING TASK   1

POSITION   1

JOINT NO.	EUCLIDEAN COORDINATES (INCHES)			ANGULAR COORDINATES (DEGREES)		
1	(0.000,	0.000,	3.290)	(0.000,	90.000,	0.000)
2	(0.000,	0.000,	3.290)	(20.000,	90.000,	0.000)
3	(-.000,	.000,	17.460)	(-20.000,	90.000,	0.000)
4	(-.000,	.000,	19.513)	(0.000,	0.000,	0.000)
5	(-.000,	-1.500,	19.513)	(90.000,	-90.000,	0.000)
6	(-.000,	-1.500,	25.054)	(90.000,	90.000,	0.000)
7	(-.000,	4.000,	25.054)	(90.000,	90.000,	0.000)
8	(-.000,	5.000,	25.054)	(0.000,	90.000,	0.000)
9	(-2.847,	4.000,	25.054)	(90.000,	120.000,	0.000)
10	(2.847,	4.000,	25.054)	(90.000,	60.000,	0.000)
11	(-1.000,	.000,	17.460)	(90.000,	180.000,	0.000)
12	(1.000,	.000,	17.460)	(90.000,	0.000,	0.000)
13	(-7.190,	.000,	17.460)	(0.000,	90.000,	0.000)
14	(7.190,	.000,	17.460)	(0.000,	90.000,	0.000)
15	(-7.190,	.000,	17.460)	(-30.000,	90.000,	0.000)
16	(7.190,	.000,	17.460)	(-30.000,	90.000,	0.000)
17	(-7.190,	.000,	6.233)	(-90.000,	0.000,	30.000)
18	(7.190,	.000,	6.233)	(90.000,	0.000,	-30.000)
19	(-7.190,	10.142,	6.233)	(90.000,	90.000,	0.000)
20	(7.190,	10.142,	6.233)	(90.000,	90.000,	0.000)
21	(-7.190,	10.142,	6.233)	(10.000,	90.000,	0.000)
22	(7.190,	10.142,	6.233)	(10.000,	90.000,	0.000)
23	(-7.190,	17.085,	6.233)	(-10.000,	90.000,	0.000)
24	(7.190,	17.085,	6.233)	(-10.000,	90.000,	0.000)
25	(-7.190,	12.811,	6.233)	(-10.000,	90.000,	0.000)
26	(7.190,	12.811,	6.233)	(-10.000,	90.000,	0.000)
27	(-3.050,	-0.000,	-1.630)	(118.145,	180.000,	0.000)
28	(3.050,	0.000,	-1.630)	(118.165,	0.000,	0.000)
29	(-3.050,	16.020,	-1.630)	(90.000,	90.000,	-61.835)
30	(3.050,	16.020,	-1.630)	(90.000,	90.000,	61.835)
31	(-3.050,	16.020,	-16.570)	(90.000,	-90.000,	0.000)
32	(3.050,	16.020,	-16.570)	(90.000,	-90.000,	0.000)
33	(-3.050,	16.020,	-19.720)	(0.000,	90.000,	0.000)
34	(3.050,	16.020,	-16.570)	(0.000,	90.000,	0.000)
35	(-3.050,	25.780,	-19.720)	(90.000,	90.000,	0.000)
36	(3.050,	16.020,	-16.570)	(90.000,	90.000,	0.000)

JOINT NO. EUCLIDEAN COORDINATES (INCHES)   ANGULAR COORDINATES (DEGREES)

JOINT NO.	EUCLIDEAN COORDINATES (INCHES)			ANGULAR COORDINATES (DEGREES)		
1	(.000,	-.527,	3.247)	(-9.224,	90.000,	0.000)
2	(.000,	-.527,	3.247)	(20.000,	90.000,	0.000)
3	(-.471,	4.202,	16.597)	(-8.934,	282.356,	-7.210)
4	(-.539,	4.887,	18.531)	(0.000,	0.000,	0.000)
5	(-.731,	3.462,	19.021)	(90.000,	-90.000,	0.000)
6	(-.210,	7.523,	22.776)	(118.241,	85.878,	2.603)
7	(-.331,	11.274,	18.756)	(90.000,	90.000,	0.000)
8	(-.363,	11.889,	17.968)	(5.124,	91.301,	0.000)
9	(-3.057,	4.676,	19.929)	(90.000,	120.000,	0.000)
10	(2.637,	10.370,	25.623)	(90.000,	60.000,	0.000)
11	(-1.462,	4.311,	16.523)	(90.000,	180.000,	0.000)
12	(.520,	4.092,	16.671)	(90.000,	0.000,	0.000)
13	(-7.558,	5.169,	15.874)	(2.461,	117.654,	0.000)
14	(6.695,	4.220,	16.252)	(-11.046,	241.967,	0.000)
15	(-7.558,	5.169,	15.874)	(-30.000,	90.000,	0.000)
16	(6.695,	4.220,	16.252)	(-30.000,	90.000,	0.000)
17	(-6.544,	6.402,	4.760)	(-101.101,	-23.691,	14.716)
18	(4.688,	7.163,	5.604)	(112.143,	28.047,	-1.106)
19	(-3.272,	15.513,	1.735)	(64.976,	90.000,	-14.257)
20	(3.066,	16.081,	1.054)	(46.810,	90.000,	18.892)
21	(-3.272,	15.513,	1.735)	(10.000,	90.000,	0.000)
22	(3.066,	16.081,	1.054)	(10.000,	90.000,	0.000)
23	(-.676,	21.951,	1.834)	(10.030,	137.821,	0.000)
24	(-.624,	21.906,	1.859)	(-36.306,	192.918,	0.000)
25	(-1.674,	19.476,	1.796)	(-10.000,	90.000,	0.000)
26	(.794,	19.667,	1.550)	(-10.000,	90.000,	0.000)
27	(-3.050,	-0.000,	-1.630)	(118.165,	180.000,	0.000)
28	(3.050,	0.000,	-1.630)	(118.165,	0.000,	0.000)
29	(-3.050,	16.020,	-1.630)	(90.000,	90.000,	-61.835)
30	(3.050,	16.020,	-1.630)	(90.000,	90.000,	61.835)
31	(-3.050,	16.020,	-16.570)	(90.000,	-90.000,	0.000)
32	(3.050,	16.020,	-16.570)	(90.000,	-90.000,	0.000)
33	(-3.050,	16.020,	-19.720)	(0.000,	90.000,	0.000)
34	(3.050,	16.020,	-16.570)	(0.000,	90.000,	0.000)
35	(-3.050,	25.780,	-19.720)	(90.000,	90.000,	0.000)
36	(3.050,	16.020,	-16.570)	(90.000,	90.000,	0.000)

SUMMATION OF TASK 1

JOINT AND CENTROID DISPLACEMENTS

	TASK DISPLACEMENT	CUMULATIVE DISPLACEMENT
JOINT 1	.529	.529
CENTROID 1	.265	.265
JOINT 2	.529	.529
CENTROID 2	.529	.529
JOINT 3	4.315	4.315
CENTROID 3	1.907	1.907
JOINT 4	5.013	5.013
CENTROID 4	4.664	4.664
JOINT 5	5.059	5.059
CENTROID 5	5.029	5.029
JOINT 6	9.308	9.308
CENTROID 6	7.154	7.154
JOINT 7	9.627	9.627
CENTROID 7	9.212	9.212
JOINT 8	9.889	9.889
CENTROID 8	9.627	9.627
JOINT 9	5.173	5.173
CENTROID 9	0.000	0.000
JOINT 10	6.399	6.399
CENTROID 10	0.000	0.000
JOINT 11	4.436	4.436
CENTROID 11	4.376	4.376
JOINT 12	4.195	4.195
CENTROID 12	4.255	4.255
JOINT 13	5.419	5.419
CENTROID 13	4.923	4.923

JOINT 14	4.417	4.417
CENTROID 14	4.302	4.302
JOINT 15	5.419	5.419
CENTROID 15	5.419	5.419
JOINT 16	4.417	4.417
CENTROID 16	4.417	4.417
JOINT 17	6.601	6.601
CENTROID 17	5.940	5.940
JOINT 18	7.613	7.613
CENTROID 18	5.831	5.831
JOINT 19	8.026	8.026
CENTROID 19	6.879	6.879
JOINT 20	8.894	8.894
CENTROID 20	7.802	7.802
JOINT 21	8.026	8.026
CENTROID 21	8.026	8.026
JOINT 22	8.894	8.894
CENTROID 22	8.894	8.894
JOINT 23	9.245	9.245
CENTROID 23	8.524	8.524
JOINT 24	10.170	10.170
CENTROID 24	9.315	9.315
JOINT 25	9.723	9.723
CENTROID 25	0.000	0.000
JOINT 26	10.480	10.480
CENTROID 26	0.000	0.000
JOINT 27	0.000	0.000
CENTROID 27	0.000	0.000
JOINT 28	0.000	0.000
CENTROID 28	0.000	0.000

JOINT 29	0.000	0.000
CENTROID 29	0.000	0.000
JOINT 30	0.000	0.000
CENTROID 30	0.000	0.000
JOINT 31	0.000	0.000
CENTROID 31	0.000	0.000
JOINT 32	0.000	0.000
CENTROID 32	0.000	0.000
JOINT 33	0.000	0.000
CENTROID 33	0.000	0.000
JOINT 34	0.000	0.000
CENTROID 34	0.000	0.000
JOINT 35	0.000	0.000
CENTROID 35	0.000	0.000
JOINT 36	0.000	0.000
CENTROID 36	0.000	0.000

PART 3

SUMMATION OF TASK 1

MASS, CENTROID PRODUCTS, FOR EACH LINK

	MASS (POUNDS)	CENTROID DISPLACEMENT (INCHES)	WORK (LB-IN)	CUMULATIVE WORK (LB-IN)
LINK 1	9.340	.265	2.471	2.471
LINK 2	0.000	.529	0.000	0.000
LINK 3	29.276	1.907	55.821	55.821
LINK 4	0.000	4.664	0.000	0.000
LINK 5	2.611	5.029	13.133	13.133
LINK 6	9.106	7.154	65.139	65.139
LINK 7	0.000	9.412	0.000	0.000
LINK 8	0.000	9.627	0.000	0.000
LINK 9	0.000	0.000	0.000	0.000
LINK 10	0.000	0.000	0.000	0.000
LINK 11	0.000	4.376	0.000	0.000
LINK 12	0.000	4.255	0.000	0.000
LINK 13	3.867	4.923	19.035	19.035
LINK 14	3.867	4.302	16.635	16.635
LINK 15	0.000	5.419	0.000	0.000
LINK 16	0.000	4.417	0.000	0.000
LINK 17	4.101	5.940	24.361	24.361
LINK 18	4.101	5.831	23.915	23.915
LINK 19	2.277	6.879	15.662	15.662
LINK 20	2.277	7.802	17.764	17.764
LINK 21	0.000	8.026	0.000	0.000
LINK 22	0.000	8.694	0.000	0.000
LINK 23	1.021	8.524	8.704	8.704
LINK 24	1.021	9.315	9.512	9.512
LINK 25	1.021	0.000	0.000	0.000
LINK 26	1.021	0.000	0.000	0.000
LINK 27	7.281	0.000	0.000	0.000
LINK 28	7.281	0.000	0.000	0.000
LINK 29	12.972	0.000	0.000	0.000
LINK 30	12.972	0.000	0.000	0.000
LINK 31	5.926	0.000	0.000	0.000
LINK 32	5.926	0.000	0.000	0.000
LINK 33	2.059	0.000	0.000	0.000
LINK 34	2.059	0.000	0.000	0.000
LINK 35	0.000	0.000	0.000	0.000
LINK 36	0.000	0.000	0.000	0.000
SUMMED TOTAL	131.3828	127.2893	272.1510	

SUMMATION OF TASK 1

JOINT ANGULAR DEFLECTIONS (PSI)

	TASK DEFLECTION (DEGREES)	CUMULATIVE DEFLECTION (DEGREES)
JOINT 1	0.000	0.000
JOINT 2	0.000	0.000
JOINT 3	-7.210	-7.210
JOINT 4	0.000	0.000
JOINT 5	0.000	0.000
JOINT 6	2.603	2.603
JOINT 7	0.000	0.000
JOINT 8	0.000	0.000
JOINT 9	0.000	0.000
JOINT 10	0.000	0.000
JOINT 11	0.000	0.000
JOINT 12	0.000	0.000
JOINT 13	0.000	0.000
JOINT 14	0.000	0.000
JOINT 15	0.000	0.000
JOINT 16	0.000	0.000
JOINT 17	-15.284	-15.284
JOINT 18	28.804	28.804
JOINT 19	-14.257	-14.257
JOINT 20	18.892	18.892
JOINT 21	0.000	0.000
JOINT 22	0.000	0.000
JOINT 23	0.000	0.000
JOINT 24	0.000	0.000
JOINT 25	0.000	0.000
JOINT 26	0.000	0.000
JOINT 27	0.000	0.000
JOINT 28	0.000	0.000
JOINT 29	61.835	61.835
JOINT 30	-61.835	-61.835
JOINT 31	0.000	0.000
JOINT 32	0.000	0.000
JOINT 33	0.000	0.000
JOINT 34	0.000	0.000
JOINT 35	0.000	0.000
JOINT 36	0.000	0.000
HEAD DEFLECTION	47.338	47.338
EYE DEFLECTION	52.098	52.098

APPENDIX I

OVERLAY	SUBROUTINES USED
CGE	POOL*
INTRAN	ANGLET, SETUP, RØT3, TRANSF, DTPF, AFF, MAB, KØMPAR*
REACHA	PREANL, MINUM, RAND*, FUN1+, FUN2+, FUN3+, FUN4+, FUN5+, FUN6+, FUN7+, REDEF
MAN1	RYTE, TASK, LINE, ROT3, POSE, TRANSF, DTPF, AFF, MAB⊗, LYNX, RAKE, SPRING (+FUN), SPRINX (+DER), POC (+PENF), POXC (+PENFX), PENLTY, TRAC, HBAD, REPLCE, CTERP, FTERP, RØ3°
INTERF	INSECT, DET2, DET3, DET4
SUMM	CMLØC
OUTGO	None

*System Subroutine

+Function subroutine

⊗With MATB, MABT entry points

°With entry point RØ3F

APPENDIX II

BLOCK COMMON NAMES IN EACH OVERLAY

APPENDIX II

OVERLAY	BLOCK COMMON USED
CGE	INTERF, TASKS, LINKS, STDPØS, CONTRL, ØUT, MAN, SUM, ITRANS, RØTATE, PROD, POINTS, FCON, INDEX, TRANS, IPØC, BDRY, PRØBLM, INØUT, KØNTRL, CIRCLE
INTRAN	INTERF, TASKS, LINKS, STDPØS, CONTRL, ØUT, SUM, ITRANS, RØTATE, PRØD, POINTS, FCØN, INDEX, TRANS, IPØC, BDRY, PROBLM, INØUT, KØNTRL, CIRCLE
REACHA	TASKS, LINKS, STDPØS, CONTRL, OUT, MAN
MAN1	STDPØS, MAN, ITRANS, ROTATE, PROD, PØINTS, FCØN, INDEX, TRANS, IPØC, BDRY, PRØBLM, INØUT, KØNTRL, CIRCLE, PØCTRA*, WDER*, RMINK*, EYE*, FUN*, KRAKE*, TRAP*, MUZZLE*, INTERP*, FANG*, ICØMM*, FTERP*
INTERF	INTERF, TASKS, LINKS, STDPØS, CONTRL, ØUT, MAN
SUMM	SUM, LINKS, STDPØS, ØUT, IPØC
OUTGO	INTERF, TASKS, LINKS, STDPØS, CØNTRL, ØUT, MAN, SUM

*Local to overlay

Appendix III

Variables In Block Common

Appendix III

Block Common	Variables	Length
(Global Blocks)		
BDRY	BL(42, BU(42), ACB(42)	126
CIRCLE	DR, PI, PII	3
CONTRL	KT1,KT2,KT3,KT4,KT5,KT6,KT7,KT8,KT9,KT10, KT11,KT12,KT13,KT14,KT15,KT16,KT17,KT18, KT19,KT20,KT21(20),KT22(20)	60
ICØN	CCØN(15),SCØN(15)	30
INDEX	IP,N,M,KØUNT,KKØUNT,LM	6
INØUT	PTI(15),PTF(15),STEPS	31
INTERF	NPL,NVERT(36),PDES(4,36), PPT(3,6,36),NCC	830
IPØC	IA,IB,IC,IJOIN,IUB,NUB,LA,LAI, LB,LJOIN,ISPRA(10),LBL,LLA(10), IH(8),IJP1	40
ITRANS	IQ(42),IPAR(42)	84
KONTRL	ISKIP,MAN,IPOSE,NSTEPS,NMJ,MAXMA, ICØN,NCASE,KPØS	9
LINKS	NL,LINKS(36),MASS(36),CNTLOC(36), LVOC(3,36),PCT(36),LN(36),OPCT(36), CPCT(36)	361
MAN	IREDEF,X3,Y3,Z3,INF,FROM,SWITCH,NSTEP	8
ØUT	BLANK,ØSERV,IM,DGCV,JTASK,IN,IN1,IN2, IN3	9
PØINTS	P(3,36),DP(3,42,36)	4644
PRØBLM	CØNST(100)	100
PROD	TP(3,3),DTP(3,3,42),TIP(3,3) DTIP(3,3,42),TRNSLT(36)	810

RØTATE	T(3,3,36),DT(3,9,36),F(2,3),JDER	1303
STDPØS	STDPØS(3,36),ANGLIM(6,36),R(3,36), RF(3,36),ANGLE1(3,36),ANGLE2(3,36)	756
SUM	CTD(36),CTMD(36),CWØRK(36), CDJA(36),CDI,CDHD	146
TASKS	NT,RHTPT(3,20),LHTPT(3,20),ETPT(3,20), TDUR(20),THØLD(20),TDES(7,20),RHØRT(3,20), LHØRT(3,20),	481
TRANS	ERR,FX,PXN,PK,SCALE,X(27),GX(27), PF(30),GSXTST,ERC	91

$$23310_8 = 9928_{10}$$

Block Common	Variables	Length
(Local to MAN1 Overlay)		
EYE	ALEPH(3),BETH(3),GIMEL(3),BETHN, BETHNS	11
FANG	H(30,30),A(30,30),B(30,30)	2700
FTERP	FI,FII,XI(30),XII(30),KILL	63
FUN	A(30),G(30)	60
ICØMM	NN,NNN,MM,MMM	4
INTERP	EP1,EP2,SIG(30),S(30),R	63
KRAKE	KCALL	1
MUZZLE	V(30),P(30),GV(30),PENX(30),PV(30,30) VLAMBD(30)	1050
PØCTRA	DRIGHT(3,15),DLEFT(3,15),PT(20)	110
RMINK	FV,PEN,ELACS,PK2,FY,FZ,GSX,GSZ,ETA, Z,Ø,FRAC,AZAHA,YHDEN,SIGY	15
TRAP	PX(30,30),PXNX(30),XLAMBD(30)	960
WDER	GW(27)	27

$$11710_8 = 5064_{10}$$

Appendix IV

Definition of Variables

VARIABLE	TYPE	DIMENSION	1st ENCOUNTER PROGRAM, SUBROUTINE	COMMON BLOCK	G:
ARRAY	REAL	(100)	CGE,		
FROM	INTEGER		CGE,	MAN	
INF	INTEGER		CGE,	MAN	
IXX*	INTEGER		CGE,		
JTASK	INTEGER		CGE,	ØUT	
NT*	INTEGER		CGE,	TASKS	
PSEUD	REAL		CGE,		
SWITCH	INTEGER		CGE,	MAN	
ZATRIX	REAL	(3)	CGE,		
ANGLE1	REAL	(3,36)	INTRAN,	STDPØS	
ANGLIM	REAL	(6,36)	INTRAN,	STDPØS	
BLANK	REAL		INTRAN,	OUT	
CDHD	REAL		INTRAN,	SUM	
CDI	REAL		INTRAN,	SUM	
CDJA	REAL	(36)	INTRAN,	SUM	
CNTLØC	REAL	(36)	INTRAN,	LINKS	
COMENT*	REAL	(24)	INTRAN,		

* Denotes Card or Tape

APPENDIX IV

COMMON BLOCK	LOCAL OR GLOBAL	DEFINITION
	L	Storage array used in Subroutine POOL
MAN	G	$\text{FROM} = \begin{cases} 0 & \text{Program MAN1 called after INTERF} \\ 1 & \text{Program MAN1 called after REACHA} \end{cases}$
MAN	G	$\text{INF} = \begin{cases} 0 & \text{Task is feasible} \\ 1 & \text{Task infeasible} \end{cases}$
	L	Used to read EØF to get abnormal exit with file and
ØUT	G	Stores current task number
TASKS	G	Stores total number of tasks in task sequence
	L	References the Hollerith word "CGE"
MAN	G	$\text{Switch} = \begin{cases} 0 & \text{No visual interference, continue} \\ 1 & \text{Program MAN1 recalled} \end{cases}$
	L	Stores logical file names, used in Subroutine POOL
STDPØS	G	References Euler angle values at each joint of BOEM Angle1(I,J): I = TYPE of angle (θ, φ, ψ) J = Joint
STDPØS	G	References lower and upper angles bounds on joint ANGLIM(I,J): $I = 2i$ Upper Bound $\quad\quad\quad\quad\quad\quad\quad 2i-1$ Lower Bound $i = 1,2,3$; = Type of angle (θ, φ, ψ) J = Joint Numb
OUT	G	Stores a 10 character blank word for the output ove
SUM	G	Stores the cumulative head deflection angle
SUM	G	Stores the cumulative eye deflection angles
SUM	G	Stores the cumulative joint twist angle for each jo
LINKS	G	Stores the relative centroid distance from a link e for each link
	L	References a description of the evaluation run for lay

* Denotes Card or Tape Input Variable

2

DEFINITION

Storage array used in Subroutine POOL

$$\text{FROM} = \begin{cases} 0 & \text{Program MAN1 called after INTERF} \\ 1 & \text{Program MAN1 called after REACHA} \end{cases}$$

$$\text{INF} = \begin{cases} 0 & \text{Task is feasible} \\ 1 & \text{Task infeasible} \end{cases}$$

Used to read EØF to get abnormal exit with file and core dumps

Stores current task number

Stores total number of tasks in task sequence

References the Hollerith word "CGE"

$$\text{Switch} = \begin{cases} 0 & \text{No visual interference, continue} \\ 1 & \text{Program MAN1 recalled} \end{cases}$$

Stores logical file names, used in Subroutine POOL

References Euler angle values at each joint of BOEMAN-I,

Angle1(I,J): I = TYPE of angle (θ, φ, ψ) J = Joint Number

References lower and upper angles bounds on joint

ANGLIM(I,J): I = $2i$ Upper Bound

                     $2i-1$ Lower Bound

i = 1,2,3; = Type of angle (θ, φ, ψ) J = Joint Number

Stores a 10 character blank word for the output overlay

Stores the cumulative head deflection angle

Stores the cumulative eye deflection angles

Stores the cumulative joint twist angle for each joint

Stores the relative centroid distance from a link endpoint
for each link

References a description of the evaluation run for output over-
lay

3

CØNST*	REAL	(100)	INTRAN,	PRØBLN
CTD	REAL	(36)	INTRAN,	SUM
CTMD	REAL	(36)	INTRAN,	SUM
CWØRK	REAL	(36)	INTRAN,	SUM
DGCV	REAL		INTRAN,	ØUT
DUM1*				
DUM2*				
DUM3*	REAL		INTRAN,	
DUM4*				
ERC*	REAL		INTRAN,	TRANS
ERR*	REAL		INTRAN,	TRANS
ETC*	REAL	(20)	INTRAN,	
ETPT	REAL	(3,20)	INTRAN,	TASKS
IA	INTEGER		INTRAN,	IPØC
IB	INTEGER		INTRAN,	ZPOC
IC	INTEGER		INTRAN,	IPOC
IM	INTEGER		INTRAN,	OUT
IP*	INTEGER		INTRAN,	INDEX
IPOSE*	INTEGER		INTRAN,	KONTRL
ISKIP*	INTEGER		INTRAN,	KONTRL
IUB	INTEGER		INTRAN,	IPOC

1

* Denotes card or tap

AN,	PROBIN	G	A set of weighting constants for penalty functio Euler angles for objective function in MAN1
AN,	SUM	G	Stores the cumulative displacement of each joint
AN,	SUM	G	Stores cumulative centroid displacement of each
AN,	SUM	G	Stores cumulative work done for each link
AN,	ϕUT	G	$= \frac{\pi}{180}$; conversion factor for degrees and radians
AN,		L	Stores superfluous descriptions of retrieved dat
AN,	TRAMS	G	Used in MAN1; <u>Allowed error</u> in each equality cor
AN,	TRAMS	G	Used in MAN1; <u>Allowed minimization</u> error for LYI
AN,		L	<u>Control code to be viewed by the eye</u> midpoint f
AN,	TASKS	G	Control point (corresponding to code) to be viev eye midpoint, $\text{ETPT}(I,J)$: I = type of Euclidian coord (x,y,z)
AN,	IPϕC	G	= 24; Used in MAN1; Index of right arm terminal
AN,	ZPOC	G	= 23; Used in MAN1; Index of left arm terminal jc
AN,	IPOC	G	= 7 ; Used in MAN1; index of eye midpoint
AN,	OUT	G	= 12; Contains intermediate output file no.
AN,	INDEX	G	Contains MAN1 error output option value (=1,Yes)
AN,	KONTRL	G	Used in MAN1; specifies whether BOEMAN's position lated from input angles.
AN,	KONTRL	G	Used in MAN1; specifies whether first position o be bypassed.
AN,	IPOC	G	= 10; Used in MAN1; number of links in spine - ri

* **Denotes card** or tape input variable

2

A set of weighting constants for penalty function and preferred
Euler angles for objective function in MAN1

Stores the cumulative displacement of each joint

Stores cumulative centroid displacement of each link

Stores cumulative work done for each link

$= \frac{\pi}{180}$; conversion factor for degrees and radians

Stores superfluous descriptions of retrieved data sets

Used in MAN1; <u>Allowed error</u> in each equality constraint.

Used in MAN1; <u>Allowed minimization</u> error for LYNX

<u>Control code to be viewed by the eye</u> midpoint for each task

Control point (corresponding to code) to be viewed by the
eye midpoint,
ETPT(I,J): I = type of Euclidian coord (x,y,z) J = Task Number.

= 24; Used in MAN1; Index of right arm terminal joint

= 23; Used in MAN1; Index of left arm terminal joint

= 7 ; Used in MAN1; index of eye midpoint

= 12; Contains intermediate output file no.

Contains MAN1 error output option value (=1,Yes)

Used in MAN1; specifies whether BOEMAN's position is to be calcu-
lated from input angles.

Used in MAN1; specifies whether first position optimization is to
be bypassed.

= 10; Used in MAN1; number of links in spine - right arm system

3

KOM	INTEGER	INTRAN,	
KT1*	INTEGER	INTRAN,	CØNTRL
KT2*	INTEGER	INTRAN,	CØNTRL
KT3*	INTEGER	INTRAN,	CØNTRL
KT4*	INTEGER	INTRAN,	CØNTRL
KT5*	INTEGER	INTRAN,	CØNTRL
KT6*	INTEGER	INTRAN,	CØNTRL
KT7*	INTEGER	INTRAN,	CØNTRL
KT8*	INTEGER	INTRAN,	CØNTRL
KT9*	INTEGER	INTRAN,	CØNTRL
KT10*	INTEGER	INTRAN,	CØNTRL
KT11	INTEGER	INTRAN,	CØNTRL
KT12*	INTEGER	INTRAN,	CØNTRL
KT13*	INTEGER	INTRAN,	CØNTRL
KT14	INTEGER	INTRAN,	CØNTRL
KT15*	INTEGER	INTRAN,	CØNTRL
KT16	INTEGER	INTRAN,	CØNTRL
KT17*	INTEGER	INTRAN,	CONTRL

*Denotes

INTRAN,		L	=	$\begin{cases} 0 & \text{Task control code matches dictionar} \\ 1 & \text{No match (output variable from subr} \end{cases}$
INTRAN,	CØNTRL	G	=	$\begin{cases} 0 & \text{Input data to be printed out} \\ 1 & \text{Input data not to be printed} \end{cases}$
INTRAN,	CØNTRL	G	=	$\begin{cases} 0 & \text{Do not print link dimensions} \\ 1 & \text{Print out link dimensions (Outgo)} \end{cases}$
INTRAN,	CØNTRL	G	=	$\begin{cases} 0 & \text{Do not print task information} \\ 1 & \text{Print task information (Outgo)} \end{cases}$
INTRAN,	CØNTRL	G	=	$\begin{cases} 0 & \text{Do not print standard angular posit} \\ 1 & \text{Print standard angular position (o} \end{cases}$
INTRAN,	CØNTRL	G	=	$\begin{cases} 0 & \text{Do not print joint angle limits} \\ 1 & \text{Print joint angle limits (o} \end{cases}$
INTRAN,	CØNTRL	G	=	$\begin{cases} 0 & \text{Do not print cockpit geometry or co} \\ 1 & \text{Print cockpit geometry and codes (} \end{cases}$
INTRAN,	CØNTRL	G	=	$\begin{cases} 0 & \text{Standard link survey used} \\ 1 & \text{Non-standard link survey used (} \end{cases}$
INTRAN,	CØNTRL	G	=	$\begin{cases} 0 & \text{Do not print standard position join} \\ 1 & \text{Print standard position joint coord} \end{cases}$
INTRAN,	CØNTRL	G	=	$\begin{cases} 0 & \text{Do not print (SRP) control location} \\ 1 & \text{print (SRP) control locations (} \end{cases}$
INTRAN,	CØNTRL	G	=	$\begin{cases} 0 & \text{Extended hand position tested} \\ 1 & \text{Clenched hand position tested (} \end{cases}$
INTRAN,	CØNTRL	G	=	$\begin{cases} 0 & \text{Task initially feasible} \\ 1 & \text{Task redefined (} \end{cases}$
INTRAN,	CØNTRL	G	=	$\begin{cases} 0 & \text{Do not print reach analysis output} \\ 1 & \text{Print reach analysis output (} \end{cases}$
INTRAN,	CØNTRL	G		Not used currently
INTRAN,	CØNTRL	G	=	$\begin{cases} 0 & \text{No visual interference} \\ 1 & \text{Visual interference exists} \end{cases}$
INTRAN,	CØNTRL	G	=	$\begin{cases} 0 & \text{Determine new position correspondin} \\ 1 & \text{Do not calculate new position} \end{cases}$
INTRAN,	CØNTRL	G	=	$\begin{cases} 0 & \text{No physical interference} \\ 1 & \text{Physical interference exists} \end{cases}$
INTRAN,	CONTRL	G		

*Denotes card or tape input variable

2

$=\begin{cases}0 & \text{Task control code matches dictionary entry} \\ 1 & \text{No match (output variable from subroutine KØMPAR)}\end{cases}$

$=\begin{cases}0 & \text{Input data to be printed out} \\ 1 & \text{Input data not to be printed}\end{cases}$

$=\begin{cases}0 & \text{Do not print link dimensions} \\ 1 & \text{Print out link dimensions (Outgo)}\end{cases}$

$=\begin{cases}0 & \text{Do not print task information} \\ 1 & \text{Print task information   (Outgo)}\end{cases}$

$=\begin{cases}0 & \text{Do not print standard angular position} \\ 1 & \text{Print standard angular position (outgo)}\end{cases}$

$=\begin{cases}0 & \text{Do not print joint angle limits} \\ 1 & \text{Print joint angle limits   (outgo)}\end{cases}$

$=\begin{cases}0 & \text{Do not print cockpit geometry or codes} \\ 1 & \text{Print cockpit geometry and codes (outgo)}\end{cases}$

$=\begin{cases}0 & \text{Standard link survey used} \\ 1 & \text{Non-standard link survey used   (outgo)}\end{cases}$

$=\begin{cases}0 & \text{Do not print standard position joint coordinates} \\ 1 & \text{Print standard position joint coordinates   (outgo)}\end{cases}$

$=\begin{cases}0 & \text{Do not print (SRP) control locations} \\ 1 & \text{print (SRP) control locations   (outgo)}\end{cases}$

$=\begin{cases}0 & \text{Extended hand position tested} \\ 1 & \text{Clenched hand position tested   (reacha)}\end{cases}$

$=\begin{cases}0 & \text{Task initially feasible} \\ 1 & \text{Task redefined   (reacha)}\end{cases}$

$=\begin{cases}0 & \text{Do not print reach analysis output} \\ 1 & \text{Print reach analysis output   (outgo)}\end{cases}$

Not used currently

$=\begin{cases}0 & \text{No visual interference} \\ 1 & \text{Visual interference exists}\end{cases}$

$=\begin{cases}0 & \text{Determine new position corresponding to new line of sight} \\ 1 & \text{Do not calculate new position}\end{cases}$

$=\begin{cases}0 & \text{No physical interference} \\ 1 & \text{Physical interference exists}\end{cases}$

e input variable

3

KT18*	INTEGER		INTRAN,	CØNTRL
KT19*	INTEGER		INTRAN,	CØNTRL
KT20*	INTEGER		INTRAN,	CONTRL
LA	INTEGER		INTRAN,	IPOC
LB	INTEGER		INTRAN,	IPOC
LHORT*	REAL	(3,20)	INTRAN,	TASKS
LHTC*	REAL	(20)	INTRAN,	
LHTPT	REAL	(3,20)	INTRAN,	TASKS
LINKS	REAL	(36)	INTRAN,	LINKS
LMEAN*	REAL	(36)	INTRAN,	
LN*	INTEGER	(36)	INTRAN,	LINKS
LSTD*	REAL	(36)	INTRAN,	
LFTY	REAL		INTRAN,	
LFTZ	REAL		INTRAN,	
LVØC	INTEGER	(3,36)	INTRAN,	LINKS
MAN*	INTEGER		INTRAN,	KØNTRL
MASS	REAL	(36)	INTRAN,	LINKS
MCPCT*	REAL	(36)	INTRAN,	LINKS
MMEAN*	REAL	(36)	INTRAN,	
MSTD*	REAL	(36)	INTRAN,	

\

* Denotes card

APPENDIX IV

N,	CØNTRL	G	Not Used currently
N,	CØNTRL	G	Not Used Currently
N,	CONTRL	G	Not Used Currently
N,	IPOC	G	= 13; Used in MAN1; total number of angles in spi
N,	IPOC	G	= 22; Used in MAN1; LA + number of angles in left
N,	TASKS	G	Task defined left hand orientation angles LHORT(I J = TYPE of angle. (θ, φ, ψ) J = Task No.
N,		L	Task defined control code for left hand, each tas
N,	TASKS	G	Control point for left hand, each task LHTPT(I,J I = coordinates (x,y,z) J = task number.
N,	LINKS	G	Stores BOEMANS link lengths, each link
N,		L	Contains link length means, each link
N,	LINKS	G	Contains link numbers, each link
N,		L	Contains link length standard deviations, each l
N,		L	Contains differences in y coordinate of seat and reference points
N,		L	Contains difference in z coordinate of seat and reference points
N,	LINKS	G	Contains 28 character link name, each link I = 1,3;10 characters each; J = link no.
N,	KØNTRL	G	$= \begin{cases} 0 & \text{If no output from MAN1 is desired} \\ 1 & \text{otherwise} \end{cases}$
N,	LINKS	G	Stores BOEMANS link mass, each link
N,	LINKS	G	References % distance of centroid from proxima
N,		L	Contains means of link mass, per link
N,		L	Contains link mass standard deviations, per link

2

* Denotes card or tape input variable

sed currently

ged Currently

ged Currently

Used in MAN1; total number of angles in spine and right arm

Used in MAN1; LA + number of angles in left arm

defined left hand orientation angles LHORT(I,J);
YPE of angle. (θ, φ, ψ) J = Task No.

defined control code for left hand, each task

ol point for left hand, each task LHTPT(I,J)
oordinates (x,y,z) J = task number.

s BOEMANS link lengths, each link

ins link length means, each link

ins link numbers, each link

ins link length standard deviations, each link

ins differences in y coordinate of seat and eye
ence points

ins difference in z coordinate of seat and eye
ence points

ins 28 character link name, each link
,3;10 characters each; J = link no.
If no output from MAN1 is desired
otherwise

s BOEMANS link mass, each link

ences % distance of centroid from proximal end

ins means of link mass, per link

ins link mass standard deviations, per link

3

D162-10127-1

109

NAF*	INTEGER		INTRAN,	
NCC*	INTEGER		INTRAN,	INTERF
NCF*	INTEGER		INTRAN,	
NL*	INTEGER		INTRAN,	LINKS
NLF*	INTEGER		INTRAN,	
NMJ*	INTEGER		INTRAN,	KONTRL
NNF*	INTEGER		INTRAN,	
NØF*	INTEGER		INTRAN,	
NPCT	INTEGER		INTRAN,	
NPF*	INTEGER		INTRAN,	
NPL*	INTEGER		INTRAN,	INTERF
NSF*	INTEGER		INTRAN,	
NSTEPS*	INTEGER		INTRAN,	KØNTRL
NT*	INTEGER		INTRAN,	TASKS
NTF*	INTEGER		INTRAN,	
NUB	INTEGER		INTRAN,	IPØC
NVERT	INTEGER	(36)	INTRAN,	INTERF
NVT*	INTEGER		INTRAN,	
OPCT*	REAL	(36)	INTRAN,	LINKS
OSERV*	REAL		INTRAN,	
P	REAL	(3,36)	INTRAN,	POINTS
PCT*	REAL	(36)	INTRAN,	LINKS
PDES*	REAL	(4,36)	INTRAN,	INTERF
PPT*	REAL	(3,6,36)	INTRAN,	INTERF

APPENDIX IV

RAN,		L	= 11; File Number for joint angular limits data
RAN,	INTERF	G	Number of control codes
RAN,		L	= 11; File Number for control code data
RAN,	LINKS	G	Number of links of BOEMAN-I
RAN,		L	= 11 File Number for link data
RAN,	KONTRL	G	Number of moveable joints
RAN,		L	= 11 File Number for normal table distribution
RAN,		L	= 11 Not currently used
RAN,		L	References each link percentile PCT (I)
RAN,		L	= 11 File Number for cockpit planes data
RAN,	INTERF	G	Number of cockpit planes
RAN,		L	= 11 File Number for standard position data
RAN,	KØNTRL	G	Size of each step in a task
RAN,	TASKS	G	Number of tasks
RAN,		L	= 11 File Number for task data
RAN,	IPØC	G	Number of joints in the spine head system
RAN,	INTERF	G	Number of vertices of each cockpit planes
RAN,		L	Number of vertices of a cockpit plane
RAN,	LINKS	G	Percentiles for non-standard survey
RAN,		L	Non-standard survey name (up to 10 characters)
RAN,	POINTS	G	Joint position array in Euclidean coordinates
RAN,	LINKS	G	Percentile values per link $P(I,J)$, $I=1,3$ (x,y,z) $J = 1$, No. of joints
RAN,	INTERF	G	40 character cockpit plane identifiers
RAN,	INTERF	G	Cockpit plane vertex coordinates $PPT(I,J,K)$ J $J=1$, no. vertices, $K=1$, no. of planes

* Denotes card or tape input variable   2

= 11; File Number for joint angular limits data

Number of control codes

= 11; File Number for control code data

Number of links of BOEMAN-I

= 11 File Number for link data

Number of moveable joints

= 11 File Number for normal table distribution

= 11 Not currently used

References each link percentile PCT (I)

= 11 File Number for cockpit planes data

Number of cockpit planes

= 11 File Number for standard position data

Size of each step in a task

Number of tasks

= 11 File Number for task data

Number of joints in the spine head system

Number of vertices of each cockpit planes

Number of vertices of a cockpit plane

Percentiles for non-standard survey

Non-standard survey name (up to 10 characters)

Joint position array in Euclidean coordinates

Percentile values per link $P(I,J)$, I=1,3 (x,y,z)
J = 1, No. of joints

40 character cockpit plane identifiers

Cockpit plane vertex coordinates PPT(I,J,K) J=1,3 (x,y,z)
J=1, no. vertices, K=1, no. of planes

3

)le

Q	REAL		INTRAN,	
R	REAL	(3,36)	INTRAN,	STDPØS
RF	REAL	(3,36)	INTRAN,	STDPØS
RHØRT*	REAL	(3,20)	INTRAN,	TASKS
RHTC*	REAL	(50)	INTRAN,	
RHTPT	REAL	(3,20)	INTRAN,	TASKS
SCALE*	REAL		INTRAN,	TRANS
SP	REAL	(3)	INTRAN,	
STDPØS*	REAL	(3,36)	INTRAN,	STDPØS
TASKNØ*	INTEGER	(50)	INTRAN,	
TCLØC	REAL	(3,91)	INTRAN,	
TCVØC	REAL	(91)	INTRAN,	
TDES	REAL	(7,20)	INTRAN,	TASKS
TDUR	REAL	(20)	INTRAN,	TASKS
THØLD	REAL	(20)	INTRAN,	TASKS
U	REAL	(51)	INTRAN,	

\
* Denotes card o

APPENDIX IV

INTRAN,		L	Denotes the normal deviate value correspondi[ng] link percentile
INTRAN,	STDPØS	G	Initial joint position array in Euclidean co[ord] (same as P)
INTRAN,	STDPØS	G	Final position array at end of task in Eucli[dean]
INTRAN,	TASKS	G	Task defined right hand orientation angles RHØRT(I,J) I=type of angle (θ, φ, ψ,); J=tas[k]
INTRAN,		L	Task defined control code for right hand, ea[ch]
INTRAN,	TASKS	G	Control location for right hand, each task RHTPT(I,J) I=coordinates (x,y,z) J=task numb[er]
INTRAN,	TRANS	G	Penalty coefficient scale factor
INTRAN,		L	Lumbar joint coordinates with respect to eye
INTRAN,	STDPØS	G	Standard angular position, each joint
INTRAN,		L	Task number of task sequence
INTRAN,		L	Task defined control locations TCLØC(I,J) I=(x,y,z) coordinates, J=task num[ber]
INTRAN,		L	10 character cockpit control codes
INTRAN,	TASKS	G	70 character task description, each task
INTRAN,	TASKS	G	Task duration in seconds, each task
INTRAN,	TASKS	G	Task hold time in seconds, each task
INTRAN,		L	Normal deviate values (positive half of table)

* Denotes card or tape input variable   2

:notes the normal deviate value corresponding to the

lnk percentile

litial joint position array in Euclidean coordinates

;ame as P)

lnal position array at end of task in Euclidean coordinates

isk defined right hand orientation angles
I∮RT(I,J) I=type of angle (θ, ϕ, ψ,); J=task number

isk defined control code for right hand, each task

introl location for right hand, each task
TPT(I,J) I=coordinates (x,y,z) J=task number

inalty coefficient scale factor

mbar joint coordinates with respect to eye reference point

:andard angular position, each joint

isk number of task sequence

isk defined control locations
:L∮C(I,J) I=(x,y,z) coordinates, J=task number

) character cockpit control codes

) character task description, each task

isk duration in seconds, each task

isk hold time in seconds, each task

irmal deviate values (positive half of table)

3

Appendix V

Storage and Retrieval
Program and Subroutine
Descriptions

Appendix V

Storage and Retrieval Subroutines

Program CAPSIS	Program RAPSIS
Subroutines Used:	Subroutines Used:
INITMF	NVNTRY
SAVTBS	LSTSTS
NVNTRY	GETSET
LSTSTS	DFST
DFST	SVTBS
SVTBS	IØRC
IØRC	TØC
TØC	KMPVLW
KMPVLW	KWDS
KWDS	RECUT2
RECUT2	MFWØRK
MFWØRK	NVNTR
NVNTR	GTST
GTST	DPØUT
DPØUT	DPIN
DPIN	TØXIC
TØXIC	

Program CAPSIS

Program CAPSIS is the creation section of the Storage and Retrieval program.
The data pool tape is created by first reading a card specifying the number
of data sets to be read from the input file. Refer to the usage section
(4.1) for the data set formats for various types of data. Each data set
read is delimited by two cards. The first specifies the set name, and the
second, specifies the end of the data. The data pool tape is created on
file BPØØL.

SUBROUTINES CALLED are: INITMF, SAVTBS, NVNTRY, LSTSTS

Figure 10 shows the Program CAPSIS flow diagram.
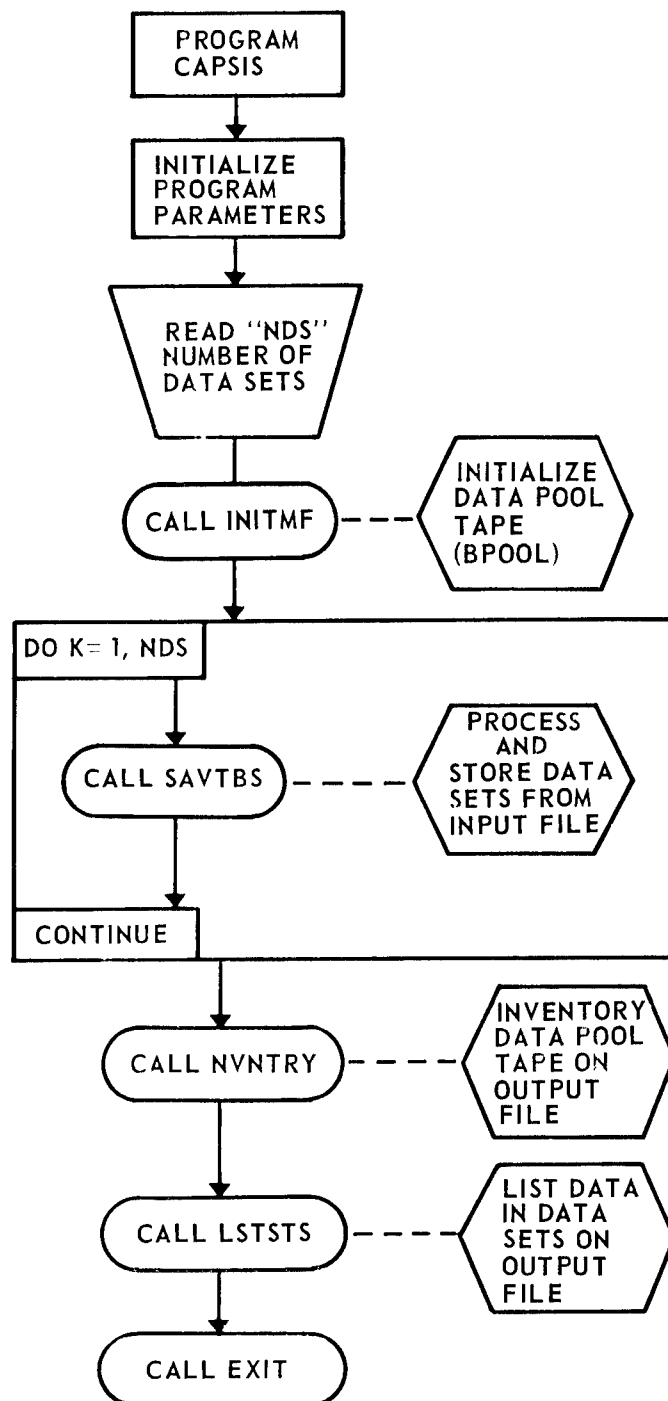
Figure 10. Program CAPSIS-Flow Diagram

Program RAPSIS

Program RAPSIS is the retrieval section of the Storage and Retrieval program. Data sets are retrieved from the data pool tape by reading a card specifying the total number of sets to be retrieved and then reading a card specifying the data set name for each set requested. The data pool tape must be mounted on file BPØØL and the output tape on which the retrieved data are stored is on file WØRK. The order of the sets retrieved is established by the order in which the data set names were read.

SUBROUTINES CALLED are: GETSET, NVNTRY, and LSTSTS.

Figure 11 shows the Program RAPSIS flow diagram.
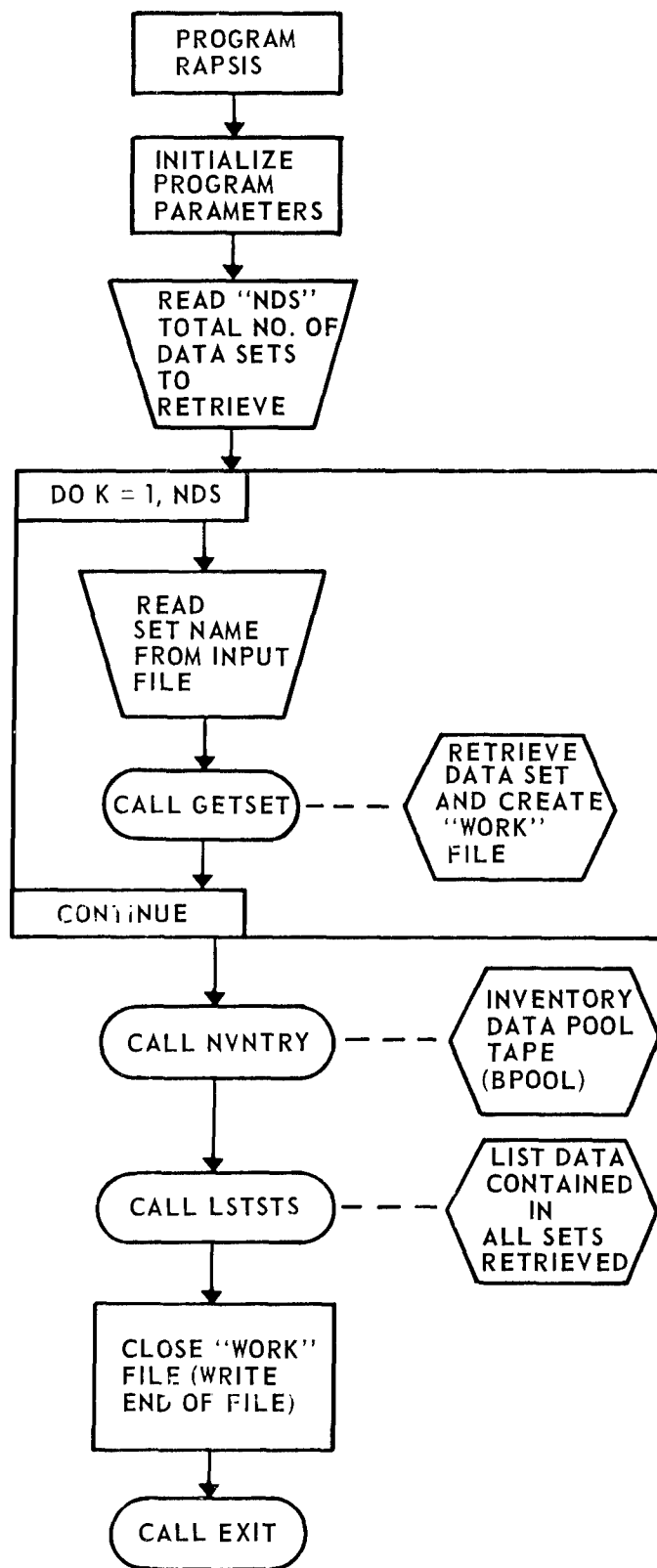
Figure 11. Program RAPSIS—Flow Diagram

Subroutine INITMF

Subroutine INITMF is called to initialize parameter when creating a data pool tape.

No Subroutines are called.

Subroutine DEFSET

This subroutine is not used in the present program configuration, but maybe utilized in later development of storage and retrieval.

Subroutines Called are: DPIN, TØC, DFST, TØCØØC, and DPØUT.

Subroutine REMSET

This subroutine is not used in the present program configuration, but maybe utilized in later development of a more sophisticated Storage and Retrieval system.

Subroutine Called are: DPIN, TØC, RMST, TØCØØC, and DPØUT.

Subroutine DFST

This subroutine is not used in the present program.

The subroutine called is: IØRC.

Subroutine RMST

This subroutine is not used in the present program.

The subroutine called is: IØRC.

Subroutine GETSET

Subroutine GETSET controls the retrieval processing of specified data sets to the user file WØRK. That is, the contexts of the data set specified is retrieved from File BPØØL and restored on user file WØRK.

Subroutine Called are: GTST, DPIN, TØXIC, and DPØUT

Subroutine GTST

Subroutine GTST is called from GETSET and performs the retrieval of particular data sets. After the set name has been read, GTST searches a table of data set names and retrieves the set from the pool. It is then written on the user WØRK file.

Subroutine Called are: KMPVLW, and KWDS.

Subroutine SVTBS

Subroutine SVTBS is called from program SAVTBS and performs the addition of data sets to the data pool tape. The data set cards are read, the names processed, and the data is written onto tape. The program checks all input set names for uniqueness.

Subroutine Called are: MFWØRK, DFST, and TØC.

Subroutine SAVTBS

Subroutine SAVTBS control the addition of the data sets to the data pool.

Subroutine Called are: DPIN, SVTBS, and DPØUT

Subroutine REMTAB

This subroutine is not used in present program configuration.

Subroutine Called are: INITMF, DPIN, RECØUT, TØC, RMST, MFWØRK, SVTBS, and DPØUT

Subroutine IØRC

Subroutine IØRC is called from DPST.

Subroutine NVNTRY

Subroutine NVNTRY provides an inventory of the data pool tape, listing all the set names, their location and length on the tape. This is written on the output file for later printing.

Subroutines called are: DPIN, TØXIC, MFWØRK, and DPØUT.

Subroutine LSTSTS

Subroutine LSTSTS provides a listing of the data contained in all the data sets stored on the data pool tape. This is written on the output file for later printing.

Subroutines called are: DPIN, DPØUT, TØXIC, GTST, and MFWØRK.

Subroutine MFWØRK

This subroutine is called from SVTBS for adding data sets, called from NVNTRY to obtain a file summary, and also from LSTSTS for full list of data.

(System) Subroutines KØMSTR, STRMØV

Two subroutines used by Storage and Retrieval programs are machine oriented so that equivalent subroutines must be written on what ever machine is used.

These routines are:

1. KØMSTR - Function compares two strings of characters

        I = KOMSTR (S1,K1,N,S2,K2)

            I = -1   String 1 ∠ String 2
                0   "   =   "
                1   "   >   "

        S1, S2 - Arrays containing strings to compare
        K1, K2 - Integer stating location in S1, S2 of
                   1st character to compare
        N   - Number of characters.

2. STRMØV - Moves a string of characters from one array to another.

        Call STRMØV (S1,C1,N,S2,C2)

        S1   - Array to be moved
        S2   - Receiving array
        C1   - Integer specifying position of first character
                to be moved relative to first position of array
        N   - Number of characters to be moved
        C2   - Integer specifying position of 1st character
                in S2 which is to receive characters from S1

APPENDIX VI

CGE PROGRAM AND SUBROUTINE DESCRIPTIONS

SUBROUTINE POOL (A,B,C)

Program CGE calls a system routine providing for file sharing in the buffer.

The calling sequence is CALL POOL (A,B,C) where:

    A = An array consisting of logical file names (of size "B")

    (Input)

    B = Number of binary files to be shared in the buffer minus one

    (Input)

    C = A local storage array for the subroutine (of size 100)

A more detailed description of the system routine POOL may be found in

References (1) and (2).

PROGRAM INTRAN

This program stores all input necessary for the evaluation whether it be user specified data on punched cards or Anthropometric or Cockpit data on magnetic tape. In each case, data is read into memory from the card reader and the retrieval tape. These data are written on an intermediate output file after being transformed. All of these files are declared in the main overlay (PROGRAM CGE). The program calls both a system routine and subroutine ANGLET (see below) which calls the remainder of the subroutines of this program.

SUBROUTINE ANGLET

The purpose of SUBROUTINE ANGLET is to calculate the initial joint locations of BOEMAN-I in the standard angular position, to initialize the palm locations and orientations, to store sine and cosines of fixed angles of joints, and to initialize the variable angles and the associated rotation matrices

Input variables required by ANGLET include:

N^* = NL = number of links

$(TRNSLT(I))$ = LINKS(I) = link lengths, I = 1, NL

$(ANG(I,J))$ = ANGLIM(I,J) = angular limit on angle I,

$$I = \begin{array}{l} 1,\ 3,\ 5\ (\text{Minimum } \theta, \varphi, \psi) \\ 2,\ 4,\ 6\ \text{at joint } J,\ J = 1,\ NL \\ (\text{maximum } \theta, \varphi, \psi) \end{array}$$

* Near the end of the routine, N is reset to the number of variable angles.

$(SPOS(I,J)) = STDPOS(I,J)$ = angular value for Θ, φ, ψ
(I = 1, 3) at joint J, J=1, NL

Output variables calculated by ANGLET include

	ISPRA(I)	= array of joint numbers for right arm and spine systems
	ILA(I)	= array of joint numbers for left arm system
	IH(I)	= array of joint numbers for head system
	IRL(I)	= array of joint numbers for right leg system
	ILL(I)	= array of joint numbers for left leg system
	WR(I)	= variable joint angle initial values (in radians) I = 1, K1 (number of variable angles)
	CR(I)	= constant angles of joints I = 1, K2 (number of constant angles)
	SCON(I)	= sin (CR(I))
	CCON(I)	= cos (CR(I))
P(I,J)	= R(I,J)	= initial joint locations I = 1,3 (coordinate x, y, z) J = joint number
	N	= number of variable joint angles
	X(I)	= initial values of variable joint angles, I = 1, N (used in MAN1)
	PTI(J)	= initial palm position (J = 1, 3), (J = 7, 9) and orientation (J = 4, 6), (J = 10, 12) right and left hands respectively and eye viewing point (J = 13, 15)

This subroutine (ANGLET) calls subroutines SETUP and TRANSF each time a link system is to be referenced. A joint angle designation table, given below, describes each of the link systems, their numbering system, and whether each angle of any joint is a variable or a constant.

SUBROUTINE SETUP

The basic function of subroutine SETUP is to sort the $ANGLIM(I,J)(=ANG(I,J))$ array into three categories

        (1)   a variable angle in a joint (θ, φ, or ψ)

        (2)   a constant angle in a joint having at least one
               other variable angle

        (3)   a constant angle in a joint having all constant angles

$ANGLIM(I,J)$ gives the lower and upper bounds of each angle in each joint. If an angle is constant, both of these bounds have the same value.

JOINT ANGLE DESIGNATION TABLE

SPINE SYSTEM

JOINT NO./ LINK NO.	JOINT NAME/ LINK NAME	VARIABLE ANGLE NO.	CONSTANT ANGLES	FIXED ANGLES
1	LUMBAR	1 (THETA)	1 (PHI) 2 (PSI)	--
2	DUMMY THORACIC			1 (THETA) 2 (PHI) 3 (PSI)
3	THORACIC	2 (THETA) 3 (PHI) 4 (PSI)		--

HEAD SYSTEM

4	VERTICAL NECK	--	--	25 (THETA) 26 (PHI) 27 (PSI)
5	HORIZONTAL NECK	--	--	28 (THETA) 29 (PHI) 30 (PSI)
6	VERTICAL	23 (THETA) 24 (PHI) 25 (PSI)	--	--
7	HORIZONTAL HEAD	--	--	31 (THETA) 32 (PHI) 33 (PSI)
8	EYE MIDPOINT VECTOR	26 (THETA) 27 (PHI)	11 (PSI)	--
9	HEAD-LEFT EYE			31 (THETA) 32 (PHI) 33 (PSI)
10	HEAD-RIGHT EYE			31 (THETA) 32 (PHI) 33 (PSI)

RIGHT ARM SYSTEM

JOINT NO./ LINK NO.	JOINT NAME LINK NAME	VARIABLE ANGLE NO.	CONSTANT ANGLES	FIXED ANGLES
12	INTERCLAVICULAR	--	--	4 (THETA) 5 (PHI) 6 (PSI)
14	CLAVICLE	5 (THETA) 6 (PHI)	3 (PSI)	--
16	DUMMY SHOULDER	--	--	7 (THETA) 8 (PHI) 9 (PSI)
18	HUMERAL	7 (THETA) 8 (PHI) 9 (PSI)	--	--
20	RADIAL	10 (THETA) 11 (PSI)	4 (PHI)	--
22	DUMMY WRIST	--	--	10 (THETA) 11 (PHI) 12 (PSI)
24	PALM (HAND EXTENDED)	12 (THETA) 13 (PHI)	5 (PSI)	
26	PALM (HAND CLENCHED)	12 (THETA) 13 (PHI)	6 (PSI)	

LEFT ARM SYSTEM

JOINT NO./ LINK NO.	JOINT NAME/ LINK NAME	VARIABLE ANGLE NO.	CONSTANT ANGLES	FIXED ANGLES
11	INTERCLAVICULAR	--	--	13 (THETA) 14 (PHI) 15 (PSI)
13	CLAVICLE	14 (THETA) 15 (PHI)	7 (PSI)	16 (THETA) 17 (PHI) 18 (PSI)
15	DUMMY SHOULDER	--	--	19 (THETA) 20 (PHI) 21 (PSI)
17	HUMERAL	16 (THETA) 17 (PHI) 18 (PSI)	--	--
19	RADIAL	19 (THETA) 20 (PSI)	8 (PHI)	--
21	DUMMY WRIST	--	--	22 (THETA) 23 (PHI) 24 (PSI)
23	HAND (HAND EXTENDED)	21 (THETA) 22 (PHI)	9 (PSI)	--
25	HAND (HAND CLENCHED)	21 (THETA) 22 (PHI)	10 (PSI)	--

RIGHT LEG SYSTEM

JOINT NO./ LINK NO.	JOINT NAME/ LINK NAME	VARIABLE ANGLE NO.	CONSTANT ANGLES	FIXED ANGLES
28	PELVIC LATERAL	--	--	34 (THETA) 35 (PHI) 36 (PSI)
30	FEMORAL	28 (THETA) 29 (PHI) 30 (PSI)	--	--
32	TIBIAL	31 (THETA) 32 (PSI)	12 (PHI)	--
34	FOOT	33 (THETA) 34 (PHI)	13 (PSI)	--
36	HEEL-TOE	--	--	37 (THETA) 38 (PHI) 39 (PSI)

LEFT LEG SYSTEM

JOINT NO./ LINK NO.	JOINT NAME/ LINK NAME	VARIABLE ANGLE NO.	CONSTANT ANGLES	FIXED ANGLES
27	PELVIC LATERAL	--	--	40 (THETA) 41 (PHI) 42 (PSI)
29	FEMORAL	35 (THETA) 36 (PHI) 37 (PSI)	--	--
31	TIBIAL	38 (THETA) 39 (PSI)	14 (PHI)	--
33	FOOT	40 (THETA) 41 (PHI)	15 (PSI)	--
35	HEEL-TOE	--	--	43 (THETA) 44 (PHI) 45 (PSI)

In addition, these bounds are stored more compactly in
bound arrays (upper and lower) and the angle numbers
and joint numbers for variable angles are initialized in
separate arrays.

The calling sequence for SETUP is

(J, IND, TEMP, ANG, SPOS, W, IQ, IPAR, K1, K2, K3, C, FL,

FLR, FLR1, F, DR, II) where

J = joint number in a link system (input)

IND(I) = a control array used to differentiate between

fixed angles in variable joints,; I = 1, 2, 3

(output)

TEMP(I) = temporary storage array for two joint angular

values (output)

ANG(I,J) = lower and upper bounds of all joint angles (input)

SPOS(I,J) = standard angular values for all joints (input)

W(I) = variable angle values used in MAN1 calculations

I = 1, K1 (output)

IQ(I) = array preserving the place in a joint number

sequence to which variable angle I belongs

I = 1, K1 (output)

IPAR(I) = array preserving the kind of angle (θ, φ, or ψ)

to which variable angle I belongs (I = 1, K1)

(output)

K1 = number of variable angles (output)

K2 = number of constant angles in variable joints (output)

K3 = number of constant angles in fixed joints (output)

C(I) = constant angle value array (I = 1, K2) (output)

FLR(I) = FL(I) = constant angle value array (I = 1, K3)
        (output)

FLR1(I) = difference of two constant angles in a joint
        I = 1, K3 (output)

F(I,J) = matrix of constant trigonometric functions of
        angles used for calculation in TRANSF I = 1, 2
                                          J = 1, 3, (output)
DR = 1.0 constant (input)

II = position number in a link system (input)

SETUP calls ROT3 to calculate the value of the rotation
        matrix F.

SUBROUTINE ROT3

This subroutine initializes a rotation matrix T (see
Section 2.4 of Math Model Document) used later in MAN1
overlay. For general use of this subroutine, see page
in the MAN1 subroutine section.

The calling sequence for ROT3 is CALL ROT3(J) where
J = joint number in a link system (input); in addition
the variables JDER and F, transmitted through COMMON/
ROTATE/, must be preset. F(I,J) is the matrix of trigono-
metric functions of the constant angles given above,
and JDER = 4 allows for the calculation of rotation matrix T.

Output from this routine is the rotation matrix

$T(I,I1,I2)$ of angles in a joint ($I = 1, 3$; $I1 = 1, 3$; $I2 =$ joint number, $K = 1,$ NL).

SUBROUTINE TRANSF

This subroutine is used to calculate a sequence of joint locations for a link system, given the link lengths, corresponding rotation matrices, and an initial joint location (either the lumbar joint or top of spine point).

Subroutine TRANSF is used predominantly in the MAN1 overlay for joint position calculation and evaluation of joint derivatives with respect to the Euler angles. A description for derivative calculation is given in the MAN1 subroutine section.

The calling sequence for TRANSF is

CALL TRANSF (V, IT, LB, LUB, K, L, MNA, MNP, LJOIN, JABOS, IDER) where

$V(I)$ = variable joint angle array = $WR(I)$, (input)

     $I = 1$, $K1$ (number of variable angles)

$IT(I)$ = array of joint numbers for a specified link system

     (i.e., ISPRA, ILA, IH, ILL, IRL) (input) $I = 1$, 10 maximum

LB = lower bound on the subscript I of $IT(I)$ (input)

LUB = upper bound on the subscript I of $IT(I)$ (input)

K = integer denoting the first constant angle of a variable

     joint (C, SCON, CCON) that the subroutine encounters in

     a link system.

L = integer denoting the first variable angle (of IQ array)

     that the subroutine encounters in a link system (input)

MNA = maximum number of angles BOEMAN-I (input)

MNP = maximum number of points (joints) of BOEMAN-I (input)

LJOIN &= number of Euler angles in the spine that are variable in V(I) (input)

JABOS &= denotes the joint succeeding the initial reference joint number (top or bottom of spine) upon which link system calculatinns are based (input)

IDER &= control variable determining whether or not joint derivatives are to be calculated

$$\text{IDER} = \begin{cases} 0 & \text{no derivatives} \\ 1 & \text{calculate derivatives} \end{cases} \quad \text{(input)}$$

Additional input required by this routine, transmitted via block common includes:

IQ(I), IPAR(I) arrays referencing joint and angle number subscripts, T(I, I1, I2), the joint rotation matrix (I2 = 1, NL; I = 1, 3, I1 = 1, 3) F(I,J), (I = 1, 2; J = 1, 3), the array of trigonometric functions of the Euler angles defining T, TIP (I,J) (I = 1, 3; J = 1, 3) references the transformation matrix at the initial joint of the sequence (bottom or top of the spine), TRNSLT(I), (I = 1, NL) the lengths of the 36 links, CCON (I), SCON(I), the sines and cosines of the constant angles in rotations T having at least one variable Euler angle V(J).

JDER, the control variable for subroutine ROT3, TP(I, J) (I = 1, 3; J = 1, 3), the current transformation matrix at a joint, and P(I, J) (I = 1, 3; J = 1, NL), the joint location array in (x, y, z) coordinates are direct output.

TP(I,J) stores the current value of the product of rotation matrices T when tracing a sequential link system such as ISPRA.

SUBROUTINE MAB

This subroutine performs the matrix multiplication $C = AB$

where A is L x M, B is M x N.

It has two entry points MATB and MABT where one calculates

$C = A^T B$ and $C = AB^T$ respectively.

The calling sequence for MAB is

CALL MAB (A, B, C, L, M, N, NRA, NRB, NRC)

A = matrix to be premultiplied, (L x M), (input)

B = matrix to be post multiplied, (M x N), (Input)

C = resultant matrix, (L x N), (output)

L = number of rows of A, Input)

M = number of columns of A and rows of B, (input)

N = number of columns of B, (input)

NRA = number of rows of A, (input)

NRB = number of rows of B, (input)

NRC = number of rows of C, (input)

SUBROUTINE AFF

This subroutine finds the first derivative of the position
vector $P(J, IT(I))$, with respect to the LLth variable
angle used in transforming from system $IT(M)$ to $IT(M+1)$,
where $M < I$. (See Section 2.4 of Math Model Document).

The calling sequence is

CALL AFF (NCALL, IT, I, LL, DP, NC) where

NCALL = control variable

$$= \begin{cases} 1 & \text{calculate derivative of initial position} \\ & \text{vector} \\ 2 & \text{calculate derivative of any other position} \\ & \text{vector} \\ 3 & \text{calculate position vector only} \end{cases}$$

IT(I) = array of joint numbers for a specified link

system (input)

I = integer designating the Ith place in the joint number

sequence (input)

LL = the derivative number to be calculated (input)

DP = the angle derivative of the joint position vector

(used in this case as $P(J, IT(I))$ itself), (output)

NC = number of columns of derivative array DP (maximum

number of angles)

The routine utilizes the following additional input from

Block Common

TP (I1, I2) - transformation matrix at a joint (I1 = 1, 3;

I2 = 1, 3).

DTP(I1, I2, I3) - Joint derivative array of the trans-
formation matrix with respect to each joint angle
(I1 = 1, 3, I2 = 1, 3, I3 = 1, maximum number of
angles).

TRNSLT(I1) = link lengths (I1 = 1, number of links)

SUBROUTINE DTPF

This subroutine calculates the first derivative of the
product array TP of rotation matrices T(I, J, IT(M))
with respect to the LLth variable angle , where M is the
MAN1 overlay but must be included whenever TRANSF is
called (since INTRAN does not require derivative
calculations).

The calling sequence is

CALL DTPF (NCALL, JT, LL) where

NCALL = $\begin{cases} 1 & \text{calculate product array TP} \\ 2 & \text{calculate angle derivative of product array} \\ & \text{TP, (input)} \end{cases}$

JT = joint number (set from IT(I)), (input)

LL = derivative number of product array TP (input)

Input variables transmitted from block common include

IPAR(I1) = array of joint angle specifiers (I1 = 1,
number of variable angles)

TP (I1, I2) = transformation matrix at a joint
(I1 = 1, 3; I2 = 1, 3)

DTP(I1, I2, I3) = angle derivatives of array TP
(I1 = 1, 3; I2 = 1, 3; I3 = 1, Maximum number
of angles)

DT(I1, I2, I3) = angle derivative matrix of rotation matrix T
(I1 = 1, 3; I2 = 1,9; I3 = 1, number of joints)

T(I1, I2, I3) = rotation matrix at joint I3 (I1, I2 = 1, 3;
I3= 1, number of joints)

SUBROUTINE KOMPAR

Subroutine KOMPAR is a system routine which tests whether
two variables store identical values. Its calling
sequence is

CALL KOMPAR (A, B, C) where

A = first variable to be compared, bit by bit (input)

B = second variable to be compared, bit by bit (input)

$C = \begin{cases} 0 & A \text{ and } B \text{ are identical} \\ \neq 0 & A \text{ and } B \text{ differ} \end{cases}$

A more detailed discussion of KOMPAR may be found in
Reference 3.

PROGRAM REACHA

This program initializes all data needed for Subroutine PREANL, which determines the gross feasibility of the task that BOEMAN-I is currently undertaking. It also presents array PTF, used in the MAN1 overlay to the terminal eye and hand joint locations, and final hand orientations of the task. When the preanalysis has been completed, the task number and feasible top of spine position are optionally written on the intermediate output tape. If the task is deemed infeasible, the final palm positions are redefined and their distances from the original task-defined positions are calculated and all pertinent information is written on the intermediate output tape. This program calls Subroutine PREANL which in turn calls the remainder of the subroutines of this overlay.

SUBROUTINE PREANL

This subroutine calculates a feasible top of spine position given the palm positions of the hands (task defined) and the location of the lumbar joint according to an optimization procedure described in the Math Model Document (Section 2.3). The optimization routine utilized is called MINUM.

The calling sequence for PREANL is given by

CALL PREANL (AA, BB, ALPHA, BETA, X0, Y0, Z0, X1, Y1, Z1, RHØ1, RHØ2, X2, Y2, Z2, SIGMA1, SIGMA2, X3, Y3, Z3, IREDEF, X11, Y11, Z11, X22, Y22, Z22, L1, L2, L3, L4, L5) where

AA = Lumbar link length (= LINKS(1)), (input)

BB = thoracic link length (= LINKS(3)), (input)

ALPHA = angle ψ limit on lumbar joint (= ANGLIM(2,1)), (input)

BETA = angle φ limit on thoracic joint (= ANGLIM(2,3)), (input)

X0 = x coordinate ⎫
Y0 = y coordinate ⎬ of initial top of the spine position (= P(I, 3), I = 1, 3), (input)
Z0 = z coordinate ⎭

X1 = x coordinate ⎫
Y1 = y coordinate ⎬ of right hand control point (= RHTPT(I,JTASK), I= 1, 3), (input)
Z1 = z coordinate ⎭

RHØ1 = radius **from right** hand control point to top of

  spine with arm in contracted attitude, (input)

RHØ2 = radius from right hand control point to top of

  spine with arm in extended attitude (straight

  arm)

X2 = x coordinate
Y2 = y coordinate } of left hand control point
  $(=$ LHTPT$(I,$JTASK$),$ I $=$ 1, 3$)$, (input)
Z2 = z coordinate

SIGMA1 = radius from left hand control to top of spine,

  arm in contracted attitude (input)

SIGMA2 = radius from left hand control to top of spine

  arm in extended attitude, (input)

X3 = x coordinate
Y3 = y coordinate } of feasible top of spine position
  (input)
Z3 = z coordinate

IREDEF $= \begin{cases} 1 & \text{Task is feasible} \\ 2 & \text{Task controls have been redefined (output)} \end{cases}$

X11 = x coordinate
Y11 = y coordinate } of redefined right palm position
  (output)
Z11 = z coordinate

X22 = x coordinate
Y22 = y coordinate } of redefined left palm position
  (output)
Z22 = z coordinate

L1 = one-half of the interclavicular link length

(= LINKS(12)), (input)

L2 = clavicular link length (= LINKS(14)), (input)

L3 = humeral link length (= LINKS(18)), (input)

L4 = radial link length (= LINKS(20)), (input)

L5 = wrist to palm link length (= LINKS(24) or LINKS(26)),

(input)

This subroutine also initializes variables used in
Subroutine MINUM, which it calls repeatedly with each of
the FUNCTIONS FUN1, FUN2, FUN3, FUN4, FUN5, FUN6, and
FUN7 and Subroutine REDEF.

SUBROUTINE MINUM

This subroutine calculates the minimum value of a function of N parameters whose values range from 0 to 1, using techniques of random direction gradient direction, average direction and jump steps in a user-prescribed mixture.

The calling sequence for this routine is

CALL MINUM (N, PARM, A, FUN, IU, IX, ISTP, IPRINT, JR, JG, JA, JJ, AA, BB, ALPHA, BETA, X0, Y0, Z0, QQ, Y1, Z1, RHØ1, RHØ2, X2, Y2, Z2, SIGMA1, SIGMA2, R, RR, DELTA) where

N = number of parameters = 3, (coordinates of top of spine location), (input)

PARM(I) = array of given starting values for parameters (mapped to [0, 1] coordinate system) and final values for parameters (input/output)

A(I,J) = working storage array dimensioned at least 7 x N, (input)

FUN = name of function to be minimized (i.e., FUNi, i = 1, 7) (input)

IU = start multiplier for random number generator subroutine RAND (IU must be odd; IU = 3746531 currently), (input)

IX = fixed multiplier for RAND (currently IX = 16777213), (input)

ISTP = total number of steps to be tried (currently ISTP = 1000), (input)

$$IPRINT = \begin{cases} -1 & \text{suppress printing} \\ +i & \text{print out at every } i^{th} \text{ step} \end{cases} \quad \text{(input)}$$

JR = unnormalized frequency of use of random step

       (currently JR = 1), (input)

JG = unnormalized frequency of use of gradient step

       (currently JG = 20), (input)

JA = unnormalized frequency of use of average step

       (currently JA = 1), (input)

JJ = unnormalized frequency of use of jump step

       (currently JJ = 1), (input)

AA = lumbar link length, (input)

BB = thoracic link length, (input)

ALPHA = angle φ limit on lumbar joint, (input)

BETA = angle φ limit on thoracic joint, (input)

$$\left.\begin{matrix} X0 \\ Y0 \\ Z0 \end{matrix}\right\} = \text{coordinates of initial top of spine position, (input)}$$

$$\left.\begin{matrix} QQ \\ Y1 \\ Z1 \end{matrix}\right\} = \text{coordinates of right hand control point, (input)}$$

$$\left.\begin{matrix} RH\emptyset1 \\ RH\emptyset2 \end{matrix}\right\} = \text{Radii from right hand control point to top of spine, in contracted and extended arm attitudes respectively, (input)}$$

$$\left.\begin{matrix} X2 \\ Y2 \\ Z2 \end{matrix}\right\} = \text{coordinates of left hand control point, (input)}$$

SIGMA1   radii from left hand control point to top of
          spine in contracted and extended arm attitudes,
SIGMA2   respectively, (input)

R = minimum distance between top and bottom of spine, (input)

RR = maximum distance between top and bottom of spine,

        (input)

DELTA = angle between X-Y plane (horizontal) **and line from**

        bottom of spine to top of spine with links at

        maximum angular deviation   (input)

<u>FUNCTIONS FUNi</u> (i = 1, ---, 7)

Functions FUN1, ---, FUN7 are used by MINUM and PREANL

to evaluate objective functions with a penalty function

representing constraints. Each is of the form

FUNi = OBJ + (PENALTY) where

$$OBJ = \begin{cases} (X-X0)^2 + (Y-Y0)^2 + (Z-Z0)^2 & i = 1, 2, 3 \\ (X5-X)^2 + (Y5-Y)^2 + (Z5-Z)^2 & i = 4 \\ (X-XM)^2 + (Y-YM)^2 + (Z-ZM)^2 & i = 5, 6, 7 \end{cases}$$

$$PENALTY = \sum_{j=1}^{6} C_j \quad \text{for all } i = 1, 7$$

For each i, however, the content of the constraint set

differs. The C_j (j = 1, --, 6) are the problem constraints, as

specified in Section 3.2 of the Math Model Document.

The calling sequence is

V = FUNi (PARAM, AA, BB, ALPHA, BETA, X0, Y0, Z0, XI, YI,

ZI, RHØ1, RHØ2, X2, Y2, Z2, SIGMA1, SIGMA2, R, RR, DELTA,

C1, C2, C3, C4, C5, C6, OBJ, X, Y, Z) where

PARAM = normalized coordinates of a feasible top-of-spine

       position, (input)

and variables AA through DELTA are as described in the

sequences of PREANL and MINUM, and

C_j, j = 1, ---, 6) are constraint equations evaluated at

      coordinates (X, Y, Z), (functions of PARAM), (output)

OBJ = value of objective function at (X, Y, Z), (output)

$$\left.\begin{array}{r} X \\ Y \\ Z \end{array}\right\} = \begin{array}{l} \text{coordinates (unnormalized) of top-of-spine} \\ \text{(evaluation point), (output)} \end{array}$$

V = value of FUNi evaluated at PARAM(J), J = 1, 3

SUBROUTINE REDEF

This subroutine redefines the task specified control points for each hand for an initially infeasible task.

The calling sequence is

CALL REDEF (X1, Y1, Z1, X2, Y2, Z2, X3, Y3, Z3, X11, Y11, Z11, X22, Y22, Z22, L1, L2, L3, L4, L5).

These variables are defined in PREANL and all but X11 through Z22 are input variables.

MINUM invokes the use of a system subroutine, called
RAND which generates random numbers for use in the
calculation of steps and search directions in the
optimization technique.

The calling sequence is

CALL RAND (IU, IX, K, N, U) where

IU = variable start multiplier, an odd-number:
$2^{24} <$ IU $< 2^{25}$, (input/output)

IX = fixed start multiplier, $2^{23} <$ IX $< 2^{25}$, IX = $8n \pm 3$
where $1048537 < n < 4194307$, (input)

K = an indicator specifying normal or uniform distribution
(K = 1, K = 0 respectively) (K = 0 currently),
(input)

N = number of random numbers desired (N = 1 currently),
(input)

U = an array of N uniformly or normally distributed
variables. If uniform, then $0 < U(J) < 1$,
J = 1, ---, N.

Program MAN1

MAN1 is the main program for the motion-model overlay which generates step-by-step task motion for a seated human pilot. It requires that the initial position of BOEMAN-I (with Euler angles) be completely specified, and the control points to be reached and viewed, along with final hand orientations, also be given. It calls the subroutines RYTE, TASK, POSE and LYNX. RYTE through POSE represent utility routines for writing, position calculation and final step constraints for the hands and eyes respectively. LYNX is the main routine enbodying the optimization procedure. If at any time during the task an infeasibility condition occurs, checks for violation of the task defining constraints and body-restriction constraints are made and violations are written on the intermediate output file. An infeasibility condition occurs if

    (1)   The completed number of minimizations to generate a BOEMAN-I position for a step exceeds 5

    (2)   The value of the penalty function exceeds the maximum allowable constraint error value EP2=1 inch

    (3)   The distances between the final hand positions and the hand control points exceed EP3=1 inch

If the task is feasible, position locations (R) and orientation angles (ANGLE1) for each step are written on the intermediate output file. Furthermore, these position locations (RG) are preserved on a binary file for the interference overlay as it tests for physical interference.

Subroutine RYTE

RYTE lists the input to MAN1 and writes out BOEMAN-I joint locations for each step, control points and constraint violations, preferred angles, objective function weights and angular bounds.

The calling sequence is

CALL RYTE (NCALL)

where NCALL = an integer specifying the kind of printout desired by the calling routines. (input)

i.e.,

$$NCALL = \begin{cases} 1 & \text{write out input data to MAN1} \\ 2 & \text{write out reinitializing man message} \\ 3 & \text{write out go onto next task message} \\ 4 & \text{write out position vectors and rotation matrices for} \\ & \text{steps between } 2^{nd} \text{ and next to last step} \\ 5 & \text{write out position vectors and rotation matrices for} \\ & \text{final step} \end{cases}$$

Other input from block common includes

$$MAN \begin{cases} \leq 0 & \text{suppress all printout from this routine and return} \\ > 0 & \text{continue as specified by NCALL} \end{cases}$$

MAN is a fixed control parameter, specified by the user at the beginning of a run.

Subroutine TASK

TASK specifies "control points" along the palm joint straight line paths corresponding to a given step during a task. It also determines the number of steps comprising a task. In the final position (step), hand orientation and directional (line of sight) constraints are also set up. This routine calls: Subroutine LINE and RØ3.

The calling sequence is:

      CALL TASK (NCALL)

where

NCALL	=	as input, a positive integer specifying the entry for TASK; as output, it specifies whether or not optimization is to be called.

Other input from block common includes:

DR	=	1.
ISKIP	$\begin{cases} = 1 \\ \\ \neq 1 \end{cases}$	if optimization is not to be called for the first position calculation of a task any other integer otherwise
KPOS	=	the number of the current position in a task
N	=	the number of parameters (variable Euler angles) in the stick-man for a task.
NSTEPS	=	as input, the maximum distance to be moved by either hand from one position to the next, in integral inches if a positive integer; if negative, its absolute value is the reciprocal of the above quantity.
PTI(I)(I=1,12)	=	the vector of equality constraints at the beginning of a task.
PTF(I)(I=1,15)	=	the same vector for the end of a task; elements 13, 14, and 15 are the coordinates of the eye control point
STEPS	=	the number of positions for the current task

Output variables from block common include

M	=	the number of equality contraints for a call to optimization routine LYNX
NSTEPS	=	the number of steps in which the next task is to be executed.
PT(I)(I=1,15)	=	the array in which the equality constraints for a call to LYNX are stored.

Subroutine RØ3

RØ3 calculates the three-space rotation matrix given Euler angles THETA, PHI, PSI. There is an entry point denoted RØ3F if the sine and cosine functions of the Euler angles are specified.

The calling sequences are

CALL RØ3(THETA,PHI,PSI,RØT)

where THETA, PHI, PSI are Euler angles (input) and RØT (output) is a 3-space rotation matrix calculated from these angles.

CALL RØ3F (THETA,PHI,BSI,ROT)

where

$$
\left.
\begin{array}{l}
\text{THETA} \\
\text{PHI} \\
\text{PSI}
\end{array}
\right\}
\quad \text{Dummy variables}
$$

$$
\text{ROT(I,J)} =
\begin{bmatrix}
\sin(\text{THETA}) & \sin(\text{PHI}) & \sin(\text{PHI-PSI}) \\
\cos(\text{THETA}) & \cos(\text{PHI}) & \cos(\text{PHI-PSI}) \\
0 & 0 & 0
\end{bmatrix}
\quad \text{(input/output)}
$$

as input and is a 3 space rotation matrix calculated from these angle functions.

Subroutine LINE

LINE performs linear interpolation for a line, given its end points as vectors in Euclidean three space.

The calling sequence is

    CALL LINE (P1,P2,P,T,I,N)

where P1 and P2 are points in N-space (input) and P (output) is a point on the line through P1 and P2, to be found using the information supplied by I and the scalar T (input).

If I = 0, T is used to scale the difference of P2 and P1 and thus determine P on the straight line joining them.

If I = 1,2,3, then a point P on the straight line joining P,P2 is found, having T as its Ith coordinate.

Subroutine PØSE

PØSE calculates the position vectors for all of the joints of BOEMAN-I given the Euler angles for each joint. It utilizes subroutine TRANSF for these calculations. This routine is similar in intent to subroutine ANGLET of the INTRAN overlay which calculates the standard starting position.

The calling sequence is

    CALL PØSE

Input is transmitted entirely through block common and includes

DR	=	previously defined
IC	=	the label of eye midpoint of the stickman
IH(I)(I=1,8)	=	the same for the spine - head system (see ISPRA)
IJOIN	=	the label of the top of the spine
ILA(I)(I=1,10)	=	the same for the spine - left arm system (see ISPRA)
ISPRA(I)(I=1,10)	=	the link-connecting point (joint) labels in the spine and right arm, starting with the top of the lumbar link and ending with the tip of the right hand
IUB	=	the number of link - connecting points (or joints) in the spine and either arm of the stickman
LJOIN	=	the number of variable Euler angles in the spine link - system
N	=	number of variable Euler angles
NUB	=	the number of link - connecting point (joints) in the spine and head system, including eye midpoint and an extra point one inch beyond the eye midpoint
TRNSLT(I)(I=1,36)	=	the link lengths for the stickman, indexed by the labels of corresponding link - connecting points, such as the positive integer elements of ISPRA.

D162-10127-1

X(I)(I=1,N)   =   the vector of variable Euler angles (parameter vector for optimization) that define a position of the stickman

Output from block common includes

ALEPH(I)(I=1,3) =   a unit vector based at the eye midpoint of the stick-man, showing the direction in which the stick-man is actually looking

BETH(I)(I=1,3) =   a unit vector giving the line-of-sight direction from eye midpoint to the eye control point in the cockpit

BETHN   =   the distance from the eye midpoint to the eye control point

BETHNS   =   the square of BETHN

P(I,J)(I=1,3;J=1,36) = the coordinates of all 36 link - connecting points and terminal points (e.g., finger-tips, eye midpoint) on the stickman

PF(I)(I=4,5,6,10,11,12) two 3-dimensional unit vectors representing right hand and left hand orientation, respectively

TIP(I,J)(I=1,3;J=1,3)= the product of the 3-space rotation matrices T(I,J,K) where I,J, and K vary from 1 to 3, K increasing from the left in the product

TP(I,J)(I=1,3;J=1,3) = the product of rotation matrices T(I,J,K) starting with the bottom of spine joint and proceeding (from left to right) as K varies over the indices of a sequential system on the stickman. TIP(I,J) is a special case, the sequential system being the spine link-connector whose labels (indices) are 1,2, and 3.

SUBROUTINE SPRING

SPRING computes the value of the objective function to be minimized. Currently the sum of squares of the differences between the orientation angles and the preferred angle values with weight coefficients comprise the objective function, i.e.,

$$FX = \sum_{i=1}^{N} G(I) * A(I)$$

where $G(I) = C\emptyset NST(I) * A(I)$

$A(I) = (X(I) - C\emptyset NST(I+N))$

The calling sequence is:

    CALL SPRING(X,FX)

where $\acute{X}(I)$ (I=1,N)   = the current value of the parameter vector (input)

FX   = the objective function value (output)

Input variables from block common are:

CONST(I) (I=1,2*N) = the constants needed for the objective
                           function

IPAR(I) (I=1,N)   = previously defined (see TRANSF)

M   = see TASK

N   = see P∅SE

Output variables from block common:

A(I) (I=1,N)   = differences between the variable angles
                       and its preferred values stored to avoid
                       repetition for the gradient calculation
                       performed in SPRINX

SUBROUTINE SPRING- Continued

G(I) (I=1,N)   = weighted differences between the variable

                        angles and its preferred values

<u>SUBROUTINE SPRINX</u>

SPRINX calculates the Gradient of the objective function (generated by
Subroutine SPRING) with respect to each variable Euler angle. i.e.,

$$GX(I) = 2 * G(I)$$

The calling sequence is

    CALL SPRINX(X,GX)

where   X(I) (I=1,N) is as defined for SPRING (input)

    GX(I) (I=1,N) = the gradient value of the objective function at

                    X (output)

Input from block common includes

    A(I) (I=1,N)   see SPRING

    G(I) (I=1,N)   see SPRING

    IPAR(I) (I=1,N) see TRANSF

    N   see PØSE

There is no output through block common.

SUBROUTINE PØC

PØC calculates the equality constraints (link constraints) given a BOEMAN-I position at each step and the hand orientations to be assumed at the final step. The line of sight constraint for the eye is also calculated for the final step. To accomplish these items, PØC calls subroutine TRANSF, obtaining position vectors and PT derivatives of position vectors with respect to the variable angles.

The calling sequence is

    CALL PØC(X,PF)

where X(I) (I=1,N) is as defined in SPRING (input)

    PF(I) (I=1,M) = the value of the constraint vector at X (output)

Input from block common includes

DR	= 1
IA	= the label of the right arm terminal point
IB	= the same for the left arm
IC	see PØSE
IH(I) (I=1,8)	see PØSE
IJØIN	see PØSE
IJP1	= IJØIN+1
ILA	see PØSE
ISPRA	see PØSE
IUB	see PØSE
KCALL	= information from RAKE, which handles all function calls to LYNX, telling PØC whether or not to call for derivatives for the gradient calculation in PØCX.

LA	= the number of variable Euler angles in the spine right arm (ISPRA) system
LAL	= LA + 1, the index of the first angle in the left arm system (ILA(I), I=4,10)
LB	= LA + (number of angles in the left arm system)
LJØIN	see PØSE
M	see TASK
N	see PØSE
NUB	see PØSE
PT(I) (I=1,15)	see TASK
TRNSLT(I) (I=1,36)	see TRANSF

Output from block common includes

ALEPH(I) (I=1,3)	see PØSE	
BETH(I) (I=1,3)	see PØSE	
BETHN	see PØSE	
BETHNS	see PØSE	
DLEFT(I,J) (I=1,3; J=1, LB-LA)		= the DP array for the left arm
DRIGHT(I,J) (I=1,3; J=1, LA)		= the DP array for the right arm
DP(I,J,K)	see TRANSF	
DTIP(I,J,K) (I=1,3; J=1,3; K=1, LJØIN)	see TRANSF	
DTP(I,J,K) (I=1,3; J=1,3; K=1,42)	see TRANSF	
P(I,J) (I=1,3; J=1,36)	see PØSE	
TIP(I,J) (I=1,3; J=1,3)	see PØSE	
TP(I,J) (I=2,3; J=1,3)	see PØSE	

SUBROUTINE PØCX

PØCX calculates the gradient matrix (PX) of the constraint vector (PF), using the first derivatives (DP) of the position vector (P) obtained by subroutine PØC. This routine calls no other subroutines.

The calling sequence is

    CALL PØCX (X,PX)

where X(I) (I=1,N) is as defined in SPRING (input)

        PX(I,J) (I=1,N; J=1,M) = the current value of the Jacobian

                             matrix of the vector function PF

                             at X (output)

Input from block common includes

ALEPH(I) (I=1,3)	see PØSE
BETH(I) (I=1,3)	see PØSE
BETHN	see PØSE
BETHNS	see PØSE
DLEFT(I,J) (I=1,3; J = 1, LB-LA)	see PØC
DRIGHT(I,J) (I=1,3; J=1,LA)	see PØC
DP(I,J,K) (I=1,3; J=1,42; K=1,36)	see TRANSF
IA	see PØC
IB	see PØC
IC	see PØSE
LA	see PØC
LAL	see PØC
LB	see PØC
LBL	see PØC
LJØIN	see PØSE

SUBROUTINE PØCX - Continued

M		see TASK
N		see PØSE

There is no output through block common.

SUBROUTINE TRANSF

TRANSF calculates joint locations and/or their first derivatives with respect to the variable Euler angles. They are generated for a sequential array of points connected by links. The Euler angles, together with arrays defining the structure of the link system are input. TRANSF calls subroutines AFF, DTPF and RØT3 for its calculation.

This routine is also present in the INTRAN overlay, where calculation of joint locations is described without derivative calculations. (IDER = 0)

To calculate joint locations and their first derivatives requires that the variable IDER be set to 1.

The calling sequence is

    CALL TRANSF(V,IT,LB,LUB,K,L,MNA,MNP,LJØIN,JABØS,IDER)

where V(I) (I=1,N)   = the variable Euler angles, from the bottom-
                             of-spine joint to terminal joint, for the
                             spine-right arm system (ISPRA), plus the
                             angles for the other sequential systems,
                             each set of angles beginning with the top
                             of spine and proceeding to the terminal
                             joint for that system. At present, the
                             systems appear in the order ISPRA, ILA, IH
                             (input)

       IT(I) (I=LB,LUB)   = one of ISPRA, ILA, or IH, giving the indices
                             K for the link-connectors $P(I,K)$ and rotation
                             matrices $T(I,J,K)$ of a sequential system

D162-10127-1
169

on the stickman. LB defines the location in the IT (e.g., ILA) array at which the system for this call to TRANSF joins on to the body (e.g., where the left arm joins the top of the spine), and LUB defines the end of the system (e.g., tip of left hand)

LB   defined above (input)

LUB   defined above (input)

K   = the current value of the index of the arrays SCØN and CCØN (see below) (input/output)

L   = the current value of the index of V (input/output)

MNA   = dimensioning information for calls to utility routines used by TRANSF. e.g., DTP should be dimensioned DTP (3,3,MNA). (input)

MNP   = the same information for the dimension P(3,MNP) (input)

LJØIN   = the number of variable angles V(I) at which the system IT joins other systems, counting from the origin of all systems on the stickman (e.g., see definition in PØSE writeup) (input)

JABØS   = the index (label) of the link-connector P(I,JABØS) (I=1,3) in the IT system that joins to the bottom of the spine; in systems ISPRA,ILA, and IH, JABØS = 1. (input)

IDER   = 1 if derivatives with respect to Euler angles of the points P(I,K) in the IT system are desired; IDER = 0 if only the coordinates of the points themselves are wanted. (input)

Input from block common includes

CC∅N(I)   = the cosines of the constant Euler angles in variable transformations T(I,J,...), corresponding to movable link-connectors (joints). I ranges from the input value of K to the output value during a call to TRANSF.

DTIP(I,J,K)   = the derivatives with respect to the variable Euler angles V(K) of the array TIP(I,J), defined in the P∅SE writeup

IQ(I)   = the value of the index of the array IT , which gives the index of the transformation T(J,K,IT) for which V(I) is a variable Euler angle

IPAR(I)   = = 1 if V(I) is the Euler angle Θ

                = 2 if V(I) is the Euler angle φ

                = 3 if V(I) is the Euler angle ψ

                in transformation T(J,K,IT(IQ(I)))

SC∅N(I)   = the sines corresponding to the cosines in CC∅N(I)

TIP(I,J)   = defined in P∅SE writeup

TRNSLT(I)   = the link lengths of the links in the stickman

Output from block common includes

DP(I,J,K) (I=1,3; J=1,42; K=1,36)

> = the derivatives with respect to variable Euler
>
> angles V(J) of link-connecting points P(I,K)

DT(I,J,K) (I=1,3; J=1,9; K=1,36)

> = the derivatives with respect to the variable
>
> Euler angles in transformation T(I,J,K) (where
>
> J=1,3) of T(I,J,K)

DTP(I,J,K) (I=1,3; J=1,3; K=1,42)

> = the derivatives with respect to Euler angles V(K)
>
> of TP(I,J), which is defined in the PØSE writeup

F(I,J) (I=1,2; J=1,3)

> = the sines and cosines of all Euler angles for a
>
> transformation T or its derivative DT ; F
>
> is used to transfer these functions to RØT3 ,
>
> which calculates T and/or DT

JDER

> = the variable specifying whether T or one of
>
> the three derivatives DT is to be calculated
>
> in RØT3 . If JDER = 4, T is calculated.
>
> JDER = 1,2,3 means calculate the derivative of
>
> T with respect to Θ, φ, or ψ.

T(I,J,K) (I=1,3; J=1,3; K=1,36)

> = the 3X3 rotation matrix for the rotation needed
>
> to transform for the local coordinate system
>
> aligned with link K-1 to that aligned with link K.

SUBROUTINE TRANSF - Continued

P(I,J) (I=1,3; J=1,36)

      = the coordinates of the link-connectors J ,

       some of which are movable (i.e., are joints),

       all expressed in the cockpit reference system

       based at the bottom of the stickman spine.

TP(I,J) (I=1,3; J=1,3)

      = defined in PØSE writeup

SUBROUTINE AFF

AFF generates a rotation and translation of a point (position vector) in three-space and calculates the first derivatives of the point with respect to the variable Euler angles.

AFF appears and is described in the INTRAN overlay.

SUBROUTINE DTPF

DTPF computes the first derivative with respect to Euler angles of a product of rotation matrices based on the Euler angles. It calls Subroutine MAB to generate the necessary products.

DTPF appears and is described in the INTRAN overlay.

SUBROUTINE RØT3

RØT3 calculates the 3-space rotation matrices and their first derivatives with respect to their corresponding Euler angles.

RØT3 appears and is described in the INTRAN overlay.

SUBROUTINE LYNX

LYNX embodies optimization procedure and minimizes a non-linear objective
function of N variables subject to non-linear equality constraints. It
uses the Davidon variable metric method (where H is the variable metric
matrix). This method initially establishes a search direction along
the line of steepest descent and a relative minimum is bracketed in an
interval $(X(I),X(I)+ETA*S(I))$. If the objective function value is still
decreasing on the interval, then the interval is too small and a larger
interval is chosen. Once a good interval is established, the location
of the minimum is estimated on that interval using cubic interpolation.
If this is not sufficient to locate the minimum, an estimate is made using
the golden section technique. Once a relative minimum is established, the
H matrix is modified and the next search direction is specified on the
basis of H and the procedure is repeated. The procedure stops when the
absolute distance from the minimum does not exceed ERR.

At this point, a new objective function based on an increased penalty
function term embodying the constraints is set up and the optimization
procedure is tried again. When the components of the constraint vector
do not exceed the constraint error ERC, the algorithm is completed and
variable orientation angles for a single step have been generated.
This subroutine calls subroutines CTERP, FTERP, RAKE, PENLTY, REPLCE,
HBAD, MAB, and TRAC.

The calling sequence is

    CALL LYNX(FUN,DER,PENF,PENFX)

where FUN   = the formal name of the objective function to be

minimized (input)

DER   = the gradient of FUN with respect to the optimiza-

tion parameters (input)

PENF = the vector function that calculates the vector

of equality constraints (input)

PENFX = the gradient of PENF (input)

These parameters must be declared with an EXTERNAL statement in any

routine which calls LYNX.

Input from block common includes

IP   = the output option for the amount of information

to be printed out by the output routine TRAC

ERC   = the allowed error in satisfying each equality

constraint for a call to LYNX

ERR   = the allowed error in approaching the constrained

minimum of the objective function during a

Davidon minimization

LM   = the upper bound on the number of Davidon

minimizations during a call to LYNX. If the

optimization procedure has not converged by the

time the number of minimizations (KKØUNT)

reaches LM, LYNX is forced to return with the

current approximation to the solution.

M   = defined in Subroutine TASK

N   = defined in Subroutine PØSE

SUBROUTINE LYNX - Continued

SCALE	= the number by which the penalty coefficient PK is to be multiplied between Davidon minimizations.
X(I) (I=1,N)	= defined in the PØSE writeup; as output, X is the solution vector of variable Euler angles for a stickman position

Output from block common includes

FX	= the objective function value at the time LYNX returns to the calling program
GX(I) (I=1,N)	= the gradient at return
KKØUNT	= the number of Davidon minimizations completed during optimization
KØUNT	= the number of iterations performed during the final Davidon minimization in the optimization procedure
PF(I) (I=1,M)	= the final value of the constraint vector, with M equality constraints
PK	= the penalty function coefficient; PK is multiplied by SCALE every minimization
PXN	= the final value of the penalty function, including the factor PK .

SUBROUTINE CTERP

CTERP performs the one-dimensional minimization of a function of many variables using the method of cubic interpolation to estimate the location of a relative minimum on an interval. CTERP calls subroutine RAKE to check the objective function value at the estimated location.

The calling sequence is

    CALL CTERP(FUN,DER,PENF,PENFX)

where FUN, DER PENF, and PENFX are externals which have been previously defined.

Input from block common includes

ETA	= the estimated scale factor for the search vector S ; i.e., the search interval for a Davidon iteration lies between the starting point X and the point $V = X + ETA * S$
FX	= defined previously
FY	= the value of the objective function at $V = X + ETA * S$
GSX	= the one-dimensional derivative of the objective function in the direction of search, evaluated at X . GSX = -(GX-TRANSPOSE) * H * GX , where H = Davidon search matrix
GSY	= the same derivative evaluated at $V = X + ETA * S$
N	= previously defined in Subroutine TASK
S(I) (I=1,N)	= the Davidon search vector for an iteration . $S = -(H * GX)$
X(I) (I=1,N)	= defined previously

SUBROUTINE CTERP - Continued

Output from block common includes

ALPHA = the scale factor for the downhill step vector SIG

FV = the value of the objective function at the interpolated minimum V

FZ = the same quantity, in another storage location

GV(I) (I=1,N) = the gradient of the objective function at the interpolated minimum V

P(I) (I=1,M) = the constraint vector at V

PEN = the penalty function at V, including the penalty coefficient PK

PV(I,J) = the Jacobian of the constraint vector at V
(I=1,N; J=1,M)

SIG(I) (I=1,N) = ALPHA * S(I) , the step to the interpolated minimum

V(I) (I=1,N) = X(I) + SIG(I) , the interpolated minimum

SUBROUTINE FTERP

FTERP performs a one-dimensional minimization of a function of many variables using the Golden Section Technique to bracket a minimum on a given interval. It returns the interpolated minimum in the interval $(X, X + ETA \cdot S)$. FTERP is called only if CTERP fails to estimate the relative minimum as it is the slower of the two routines. FTERP repeatedly calls subroutine RAKE to determine the objective function value at a given location.

The calling sequence is

    CALL FTERP(FUN,DER,PENF,PENFX)

where FUN, DER, PENF, and PENFX have been defined previously (see CTERP).

Input from block common includes

EP1 = PE2	= the convergence limit for the algorithm
ETA	previously defined
FX	previously defined
FY	previously defined
GSX	previously defined
GSY	previously defined
N	previously defined
R	= the Fibonacci ratio to the desired decimal place, .618 ... .
S(I) (I=1,N)	previously defined
X(I) (I=1,N)	previously defined

SUBROUTINE FTERP - Continued

Output from block common includes

ALPHA   previously defined

FV   previously defined

FZ   previously defined

GV(I) (I=1,N)   previously defined

KILL   = the number of iterations of the golden section

                     procedure; $5 \leq \text{KILL} \leq 50$

P(I) (I=1,N)   previously defined

PEN   previously defined

PV(I,J) (I=1,N; J=1,M) previously defined

SIG(I) (I=1,N)   previously defined

V(I) (I=1,N)   previously defined

SUBROUTINE RAKE

This subroutine is used to supply all values of the objective function
and penalty function (and constraint vector) given the values of the
variable angles X . It also generates gradient values for the objective
function and penalty function. The subroutine is set up to allow for
function calculation using the variable names of the calling routine.
Subroutines called are FUN, DER, PENF, PENFX, and PENLTY.

The calling sequence is

    CALL RAKE(FUN,DER,PENF,PENFX,NCALL)

where RUN, DER, PENF, and PENFX have been defined (see $\emptyset$TERP) and

    NCALL = the entry point for RAKE. If = 1, this is the first

            call for function values in an optimization, and so

            the initial values of FX and PXN are found and the

            corresponding scale factors FSC and PK are calculated

            so that FX = PXN = 10., as long as FX $\geq$ 1. or

            PXN $\geq$ 1. FSC remains unchanged thereafter, and PK

            is multiplied by SCALE after each minimization.

Input from block common includes

    FX   = as input, the objective function value to be

                    modified for the beginning of the next minimization

                    to take into account the change in PK (the

                    total FX value = (unmodified objective function

                    FX) * FSC + PK * PXN)

    GX(I) (I=1,N)   = the same quantity for the gradient

    PK   previously defined

PK2	= .5 * PK , used to calculate FI and FII
PXN	= the input quantity corresponding to FX above
PXNX	= the input quantity corresponding to FX above
V(I) (I=1,N)	previously defined
X(I) (I=1,N)	previously defined
XI(I) (I=1,N)	= a point at which the objective function is evaluated during Fibonacci (golden section) search
XII(I) (I=1,N)	= a point at which the objective function is evaluated during Fibonacci (golden section) search

Output from block common includes

FI	= objective function value at XI
FII	= objective function value at XII
FV	previously defined
FX	previously defined
GV	previously defined
GX	previously defined
P	previously defined
PEN	previously defined
PENX	= the gradient of PEN
PF	= previously defined
PK	previously defined
PV	previously defined
PX	previously defined
PXN	previously defined
PXNX	= the gradient of PXN

SUBROUTINE RAKE – Continued

$\text{VLAMBD(I)} \ (I=1,M) \ . \ = PK * P(I)$

$\text{XLAMBD(I)} \ (I=1,M) \ = PK * PF(I)$, sheuld approach the Lagrange
multiplier vector for a problem with
equaltiy constraints $PF(I)$ but no
inequality constraints.

SUBROUTINE PENLTY

PENLTY calculates the penalty function value based on the constraint vector. It calls no subroutines.

The calling sequence is

    CALL PENLTY(PF,PX,XLAMBD,PXN,FX,PXNX,GX,PK)

where the parameters in the list correspond to definitions given previously. For some calls to PENLTY, the actual parameters may be P, PV, VLAMBD, PEN, FV, PENX, GV, and PK .

PF, PX, PK, and the unconstrained objective function and gradient FX and GX are input. FX and GX are output as (input FX) + PXN and (input GX) + PXNX, respectively, along with XLAMBD, PXN, and PXNX .

SUBROUTINE REPLCE

REPLCE stores a set of variables evaluated at the initial point of a search interval in the storage locations corresponding to the end point of the interval. It also performs the reverse of this procedure. It calls no subroutines.

The calling sequence is

    CALL REPLCE(NCALL)

where NCALL = 1 (input) means replace all variables from list 1 by those on list 2 and NCALL = 2 means replace all variables from list 2 by those on list 1 . All quantities are input/output through labelled common.

    List 1: FX, PXN, X, GX, PXNX, PX, PF

    List 2: FV, PEN, V, GV, PENX, PV, P

SUBROUTINE HBAD

HBAD modifies the Davidon variable metric matrix H for reinitialization
for the next minimization and may restore H to positive definiteness
by zeroing off-diagonal elements and replacing diagonal elements by
their absolute value. A positive definite matrix H satisfies

$$X^T H X > 0 \quad \text{for all non zero vectors } X.$$

The calling sequence is

CALL HBAD(NCALL)

where NCALL = the entry for HBAD; specifying the array in which H
is to be corrected. Presently, NCALL = 3 is used
exclusively; here, H is set to the identity.

Input from block common includes

GSX   previously defined

H(I,J) (I=1,N; J=1,N)   previously defined

N   previously defined

YHDEN   = a quantity used to store information about
                        the search direction at time of call to
                        HBAD; not used at present

Output from block common includes

N(I,J) (I=1,N; J=1,N)   previously defined

SUBROUTINE MAB

MAB performs the matrix multiplication AB of two matrices A and B.
It has two entry points MATB and MABT which calculate A^TB and AB^T
respectively. (T denotes the transpose operation on a matrix). MAB
is also utilized in the Intran overlay.

The calling sequences are

    CALL MAB(A,B,C,L,M,N,NRA,NRB,NRC)

or

    CALL MATB(...)

or

    CALL MABT(...)

where A(I,J) , B(I,J) (input) are arrays from which matrices are to be
multiplied and C(I,J) (output) is the array in which the result is to
be stored. A, B, and C have row dimensions NRS, NRB, and NRC, (input)
respectively, and must be so dimensioned in the calling program. L,
M, and N are the input dimensions of the matrices to be multiplied;
the rule is that an LXM matrix times and MXN matrix yields an LXN
matrix, regardless of whether these matrices are A or A- transpose
or B or B- transpose. However, NRA and NRB are the true
row-dimensions of the arrays A and B , respectively.

SUBROUTINE TRAC

TRAC provides optional output based on calculations made in Subroutine
LYNX. This output provides information regarding the progress of the
optimization procedure during the search for a constrained minimum
of the objective function.

The output option is dependent on the value of IP which is user specified.

IP =
0 no output

1 write X, FX at every iteration

2 in addition, write one dimensional minimization data

3 write X, FX, PXN, PF

4 in addition, write GX, PXNX and one dimensional
   minimization data

5 in addition, write H at every iteration

6 write X, FX, and H at every iteration

7 write H at every iteration

8 write X, FX every iteration and attempted refinement
   data if successful

9 in addition, write PF, PXN every iteration

One dimensional minimization data includes the endpoint

$V = X + ETA * S$ of the search interval

$FY = FV$ objective function value

GSX, GSV one dimensional derivatives at X and V and the cosines
          of the angles between gradients GV, GX and the search
          vector S

SUBROUTINE TRAC - Continued

The calling sequence is

    ·CALL TRAC(NCALL)

where NCALL = the output mode parameter for TRAC. TRAC will output

               most of the quantities in block common, hence all block

               common corresponding to LYNX should be in TRAC.

PROGRAM INTERF

The purpose of INTERF is to detect and correct for visual inter-
ference between BOEMAN-I and his surrounding environment. In
addition, physical interference between BOEMAN-I and the seatback
is detectable. The detection process consists of finding intersec-
tions of line segments and bounded cockpit planes. For visual
interference, the line segment used is that between the eye
midpoint and the task defined eye aiming point; for physical
interference, each link of BOEMAN-I plays the line segment role.
On the other hand, all cockpit planes are tested in visual
interference whereas only the seatback planes are considered
for physical interference.

Correcting for visual interference consists of redefining the
line of sight (with the eye aiming point fixed) so that it and
the offending plane no longer intersect. If the correction
procedure is not successful, the task is deemed infeasible.
Visual interference is tested at the end of each task; physical
interference with the seatback is tested at each step of the
task. A discussion of the mathematical procedures used for the
interference analysis appears in Section 3.4 of the Mathe-
matical Model Document (D6-53620-2).

The entire interference procedure is embodied in PROGRAM INTERF and subroutines called are only for subsidiary calculations. INTERF is only called if the task is feasible.

Input variables set in previous overlays and transmitted through block Common and stored in local variables include:

NPL = number of cockpit planes (NPLANE)

NVERT(I) = number of vertices of plane I, (N)

PPT(I,J,K) = coordinates I of plane K at vertex J,

(BOUND(I,J)) K = 1, NPLANE

*ETPT(I,J) = coordinates of eye aiming point (task defined)

(EFP(I)) I=1,3

RF(I,7) = eye midpoint (x, y, z) coordinates (EMP(I))

I = 1, 3

SWITCH = control indicator for whether visual interference or physical interference is to be checked

$= \begin{cases} 0 & \text{check visual interference} \\ 1 & \text{check physical interference only} \end{cases}$

*RG(I,J,K) = intermediate joint locations during the task

(SLINK(I,J)) I = 1, 3; J = 1, 26; (number of movable joints) K = position number

NSTEP = total number of steps in the task, (ISTEP)

PPT(I,J,K), K = 29, 30, seat back planes, (SBOUND(I,J))

K = 29, 30

* Output from program to intermediate output tape

*LVOC(I,J) = link name I = 1, 3; J = 1, NL (Number of links)

*PDES(I,J) = name of cockpit plane I = 1, 4; J = 1, NPL
(Number of planes)

Those quantities calculated during the interference
analysis or used as output are:
(Visual interference)

CENT(I) = centroid of plane K, I = 1, 3

LIGHT(I) = indicator of relationship between plane I and
line of sight

$$LIGHT(I) = \begin{cases} -1 & \text{intersection point is not on line of sight} \\ 0 & \text{set upon entry to program} \\ 1 & \text{line of sight embedded in plane I} \\ 2 & \text{line of sight parallel to plane I} \\ 3 & \text{line of sight intersects plane I outside bounded region} \\ 4 & \text{line of sight intersects plane I inside bounded region} \\ 5 & \text{line of sight intersects plane I at eye aiming point} \end{cases}$$

*SOLUT(I,J) = intersection point coordinates for plane J
(I = 1, 3)

NV(I) = number of vertices of plane K, K = 1, NPLANE

K7 = number of cockpit plane intersections at the end of
a task

LOSMP(I) = midpoint of line of sight (I = 1, 3)

* Output from program to intermediate output tape.

BBOUND(I,J,K) = storage array for planes K for which

        interference has occurred, I = 1, 3, J = 1,

        number of vertices (NV(K))

*Q(I) = coordinates of point on redefined line of sight

        I = 1, 3

(Physical interference)

CEN(I) = centroid of seatback plane K (I = 1, 3), K = 29, 30

IGHT(I) = indicator of relationship between link K and

        seatback plane

$$
\text{IGHT(I)} =
\begin{cases}
0 & \text{upon entry} \\
6 & \text{link I intersects seatback plane outside plane boundary} \\
7 & \text{link I intersects seatback plane inside plane boundary} \\
8 & \text{link I intersects seatback plane at link end point} \\
9 & \text{intersection point is beyond link I} \\
11 & \text{link I imbedded in seatback plane} \\
12 & \text{link I parallel to seatback plane}
\end{cases}
$$

*N8 = link number for seatback interference

*SOL(I,J) = intersection point between link J and

        seatback plane (I = 1, 3)

*   Output from program to intermediate output tape

SUBROUTINE INSECT

The purpose of subroutine INSECT is to discover if there is an intersection between the plane boundary line (an edge) and the centroid-intersection point line (lack of such an intersection implies the point of intersection is inside the bounded plane).

The calling sequence of INSECT is

CALL INSECT (S1, S2, S3, C, B11, B12, B13, B21, B22, B23, IND) where

$\left.\begin{array}{c} S1 \\ S2 \\ S3 \end{array}\right\} =$ are coordinates point of intersection (SOL(I,J) or SOLUT(I,J), I = 1, 3; J fixed)

C = an array (of dimension 3) storing the centroid coordinates

$\left.\begin{array}{c} B11 \\ B12 \\ B13 \end{array}\right\} =$ coordinates of a plane vertex (BOUND(I,J), SBOUND(I,J), I = 1, 3; J fixed)

$\left.\begin{array}{c} B21 \\ B22 \\ B23 \end{array}\right\} =$ coordinates of the adjacent plane vertex (BOUND(I,J + 1), SBOUND(I,J + 1), I = 1, 3, J fixed)

IND = $\begin{cases} 0 & \text{no intersection} \\ 1 & \text{intersection has occurred between the two lines} \end{cases}$

All variables except IND are input to the subroutine.

SUBROUTINE DETi (i = 2, 3, 4)

The purpose of DETi (i = 2, 3, 4) is to calculate the value of a determinant of order i. The method used is the repeated application of Laplace expansion. The general calling sequence for DETi (i = 2, 3, 4) is

CALL DETi (Ajk, F) where j = 1, i; k = 1, i; i = 2, 3, 4 and k varies most often, and

Ajk = element in j^{th} row and k^{th} column of determinant of order i, (input)

F = value of determinant of order i

PROGRAM SUMM

Program SUMM is called by the main overlay of the computer program for each task performed, that task is feasible. The following quantities are calculated:

1. TD   Linear displacement of each joint per task

2. TMD   Linear displacement of each mass centroid

3. CTD   Cumulative summation of linear displacement for each joint

4. CTMD   Cumulative summation of mass centroid displacement for each link

5. WORK   Work for each link per task

6. CWORK   Cumulative summation of work for each link

7. SMASS   Total mass of man

8. SCTD   Total joint displacement for task

9. SWORK   Total work of all link per task

10. DJA   PSI deflection (twist) per link per task

11. CDJA   Cumulative summation of PSI deflection for each link

12. DHD   Head Deflection

13. DI   Eye Deflection

14. CDHD   Cumulative summation of head deflection

15. CDI   Cumulative summation of eye deflection

The program accepts input through labeled common from other overlays. The necessary input includes initial and final positions of all joints, the per cent of the link length for centroid location, parameters defining link structure of man-model, mass values for each link, and Euler angles for each link. The output is written onto the intermediate output file for later processing in the output overlay.

The subroutine called by SUMM is Subroutine CM1ØC

Input to SUMM via block common includes

ANGLE1 - initial Euler angle array

ANGLE2 - terminal Euler angle array

R - array of initial joint locations

RF - array of terminal joint locations

MASS - array of masses for each link

LINKS - array of link lengths

CPCT - array of percentages of link length for centroid
       location

SUBROUTINE CMLØC

CMLØC calculates the initial and final mass centroid location given initial and final joint locations and the location of the centroid relative to the proximal end of the link.

The calling sequence for CMLØC is

CALL CMLØC (R, RF, J, JM1, CPCT, II, CLØC, CLØCF)

DIMENSION R(3, NL), RF(3, NL), CPCT(NL), CLOC(3, NL),

CLØCF(3, NL) where

R - array of initial joint locations, (input)

RF - array of final joint locations, (input)

J - particular link number (joint number, top of link), (input)

JM1 - previous joint number for particular link

CPCT - array of percentages referencing length along link

       where mass centroid is located, (input)

II - index to determine particular link, (input)

CLØC - initial mass centroid location, (output)

CLØCF - final mass centroid location, (output)

PROGRAM ØUTGØ

ØUTGØ is the output overlay. It accepts the intermediate output file written by previous overlays and a separate binary file which stores intermediate position and orientation data written by the MAN1 overlay. The output is in tabular form and covers three broad areas: input data, task processing information and summation. The tables which are largely self-explanatory, provide a history of BOEMAN-I's performance and are arranged by task. The summation area is given for each computed (feasible) task and provides cumulative totals from the starting position. For feasible tasks, BOEMAN-I's initial and final position are printed. If the task is infeasible, all intermediate positions are printed, up to the point of infeasibility.

All output is written onto the output file as well as a backup tape which stores duplicate printer information. This backup tape is used for chart and graph information as well as for duplication copies of the printout, if necessary.

Control parameters of the form KTi (i = 1, ---, 20) are used to provide various output options for the user which may abbreviate the total number of output pages or tables.

The program is self-contained and calls no subroutines.

Input variables to ØUTGØ transmitted via block Common include:

KTi, i = 1, 20 control parameters referencing printout
        options

NCC = number of control codes

NPL = number of cockpit planes

NVERT(I) = number of vertices of the Ith cockpit plane

All other input variables are read from the intermediate output tape and the MAN1 output tape. This overlay is called after all tasks have been processed and all of the input variables are written on the printer and backup tape.

Appendix VII

Validation Program and
Subroutine Descriptions

Program SVØMM

This program accepts joint location data ("N" coordinates) of humans at midpath during a task. Each task is repeated "NR" times and an average location results. This is then compared with the same sized BOEMAN-I for the same task at its corresponding midpath location. In each case, the joint coordinates are determined using a cutting plane through the path.

An F-statistic is used to test the hypothesis that there is no significant difference between the paths of motion of BOEMAN-I and humans in performance of the task (i.e they belong to the same population). Using the formula for the F-value

$$F = \frac{NR*(NR-N)}{N*(NR-1)} \sum_{\substack{I=1 \\ J=1}}^{N} (XM(I) - SXM(I)) \cdot \left[C\emptyset V[X] \right]_{IJ}^{-1}$$

where

$$\left[C\emptyset V[X] \right]_{IJ} = \frac{1}{NR-1} \left[(X(I,K) - XM(I))*(X(J,K)-XM(J)) \right]_{(IxJ)}$$

for all I=1,N, J=1,N, K=1,NR

And the degrees of freedom are given as

$\quad$ M = NR-N and N

Figure 11 contains a general flow diagram of Program SVØMM.

Input for the program includes

$\quad$ CØDE = a control to allow for sequential processing

$\quad$ =HYPØ $\begin{cases} = \text{PRØBLM} & \text{process following data} \\ \neq \text{PRØBLM} & \text{stop} \end{cases}$

$\quad$ PN $\quad$ = descriptive title of data set

$\quad$ N $\quad\;$ = number of joint coordinates being compared

$\quad$ NR $\quad$ = number of repetitions of the path

$\quad$ MX $\quad$ = maximum order of covariance matrix
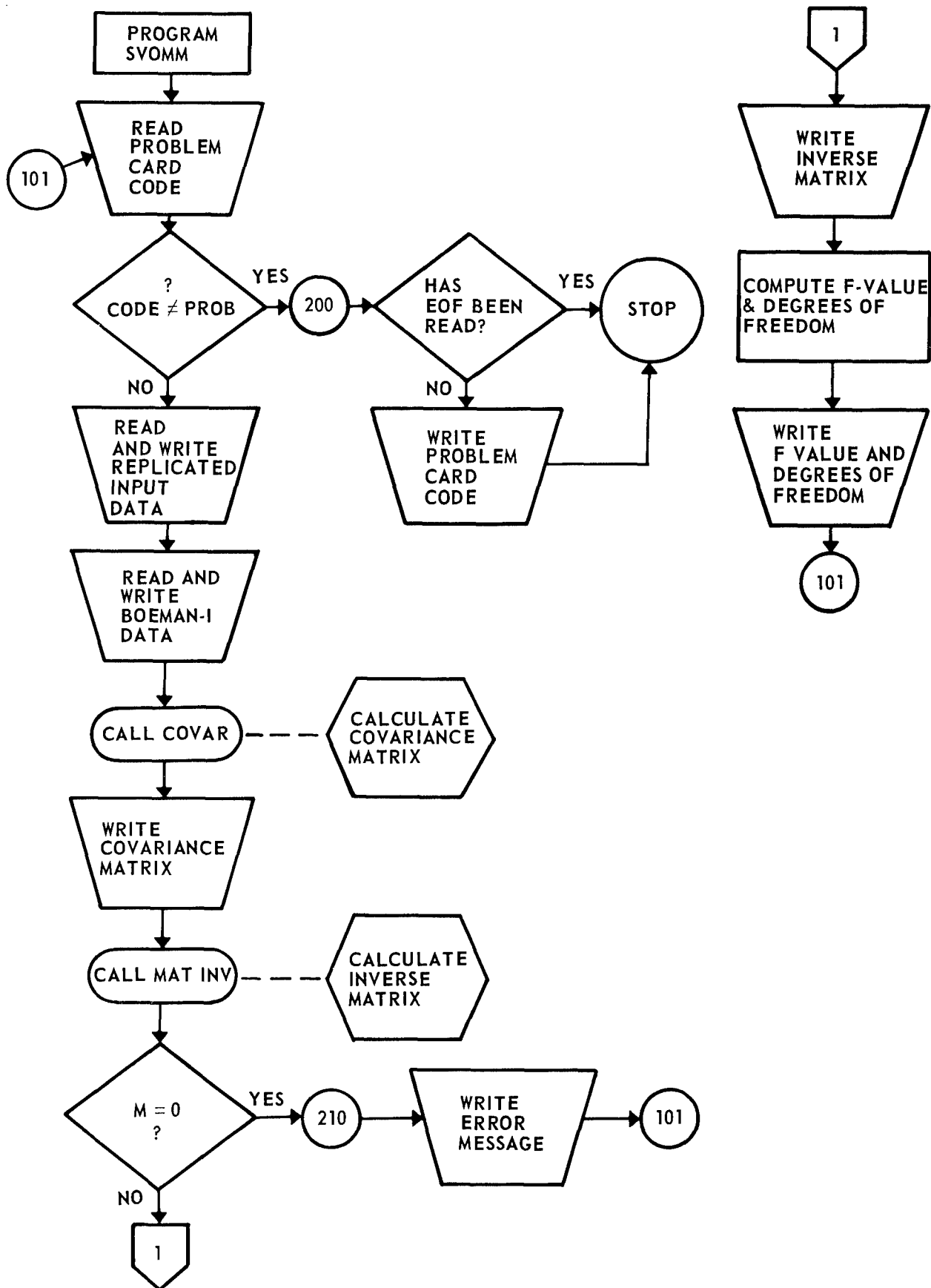
Figure 12. Program SVOMM—Flow Diagram

INP $\begin{cases} = & \text{output control to print out input data} \\ = & \text{1 printout data; } \neq \text{1 suppress data printout} \end{cases}$

FMT $\quad = \quad$ format statement number allowing for input of data of various accuracies

X(I,J) $\quad = \quad$ replicated joint location data (one coordinate only) I=1, N, J=1, NR

SXM(I) $\quad = \quad$ BOEMAN-I joint location data (one coordinate only) I=1, N

Output data includes

PN $\quad = \quad$ problem description

X(I,J) $\quad = \quad$ replicated joint location data (one coordinate only) J=1,N J=1, NR

SXM(I) $\quad = \quad$ BOEMAN-I joint location data (one coordinate only) I=1, N

A(I,J) $\quad = \quad$ covariance matrix of replicated data I,J=1,N

AI(I,J) $\quad = \quad$ inverse of covariance matrix I,J=1, N

FV $\quad = \quad$ F value (computed)

N $\quad = \quad$ number of joint coordinates being compared; (degree of freedom)

M $\quad = \quad$ difference between number of replications and number of compared coordinates

A check is made to insure that computation of the inverse matrix does not yield singularity.

Subroutine CØVAR

The purpose of CØVAR is to calculate a covariance matrix using the input data of replicated joint locations for each path or task. The matrix is calculated using

$$XM(I) = \sum_{J=1}^{NR} X(I,J)/NR$$

$$RNN = \frac{1}{(NR-1)}$$

$$SA(I,J) = RNN*\sum_{K=1}^{NR} (X(I,K)-XM(I))*(X(J,K)-XM(J)); \quad I,J=1,N$$

where XM is the input data means; RNN is the reciprocal of the number of replications minus one; and SA is the covariance matrix.

The calling sequence for CØVAR is

CALL CØVAR(NR,NT,X,XM,SA)

where

NR	=	number of repetitions of path (input)
NT	=	number of joint coordinates being compared (input)
X(I,J)	=	array storing replicated joint locations (one coordinate only) (input) I=1,NR J=1,NR
XM(I)	=	means of χ array I=1,NT (output)
SA(I,J)	=	covariance array I,J=1,NT (output)

Subroutine MATINV

This subroutine finds the inverse of a non-singular NXN matrix. The method uses the Gaussian elimination technique with pivoting of the diagonal elements on both the original matrix and the identity matrix. A check for singularity is also performed.

The calling sequence for MATINV is

CALL MATINV (SA,AI,N,NR,M)

where

SA(I,J)	=	matrix to be inverted (I,J=1,N) (input)
AI(I,J)	=	inverse matrix (I,J=1,N) (output)
N	=	number of rows and columns of SA and AI (input)
NR	=	maximum number of rows of input matrix (input)
M	=	output indicator for singularity of input matrix

$$M \begin{cases} = 0 & \text{SA is singular} \\ \neq 0 & \text{SA is non-singular} \end{cases} \text{(output)}$$

Appendix VIII

Multi-mission Simulator Control

Code Definitions

MULTIMISSION SIMULATOR CONTROL CODE LISTING

(Identifies centroid unless specified otherwise)

Flight Controls (FC)

(1)	FCFWS	Flap/Wing sweep control (primary)
(2)	FCEPS	Emergency flap switch
(3)	FCRT	Electrical control (rudder trim)
(4)	FCEPT	Electrical control (emergency pitch trim)
(5)	FCCSC	Center stick control
(6)	FCSPS	Speed brake switch
(7)	FCT	Throttles

Flight Instruments (FI)

(1)	FIRPML	Engine display (left % RPM)
(2)	FIRPMR	Engine display (right % RPM)
(3)	FIPT	Pitch/trim
(4)	FIAT	Aileron trim
(5)	FIWSFP	Wing sweep/flap position
(6)	FIFT	Total fuel
(7)	FIFI	Digital fuel (panel centroid)
(8)	FITSI	Turn and slip indicator
(9)	FIAAT	Angle of attack (tolerance indicator)
(10)	FIAAA	Angle of attack (analog)
(11)	FIHUC	Head-Up Display Control (center)
(12)	FIHUD	Head-Up Display (screen center)
(13)	FIAI	Attitude Indicator
(14)	FIAS	Airspeed
(15)	FIMM	Machmeter
(16)	FIVSD	VSD (Vertical Situation Display) centroid
(17)	FIVSDSS	VSD Symbol Selector (upper left corner)
(18)	FIVSDH	VSD horizon selector (upper right)
(19)	FIVSDI	VSD intensity/contrast (lower right)
(20)	FIVSDC	VSD switch (lower left)
(21)	FIC	Clock
(22)	FIBA	Barometric altimeter
(23)	FIRA	Radar Altimeter
(24)	FIMC	Magnetic compass
(25)	FIRC	Rate of climb
(26)	FIGM	G-meter
(27)	FIRMI	RMI (Radio magnetic indicator)
(28)	FIHSD(1)	HSD (horizontal situation display)
(29)	FIHSD(2)	HSD (extreme bottom point on screen)
(30)	FIHSDGC	HSD (gain/contrast selector)
(31)	FIHSDCI	HSD (chart intensity selector)

Automatic Flight Control (AFC)

(1)	AFCSCP	AFC System control panel
(2)	AFCSSP	Steering select panel
(3)	AFCSAP	Stability augmentation panel
(4)	AFCADC	Air data command control panel

Mechanical Systems (MS)

(1)	MSESP	Engine start panel
(2)	MSEPS	Elec Control panel - external power switch
(3)	MSLCI	Lights control - internal
(4)	MSLCE	Lights control - external
(5)	MSLGC	Landing gear control
(6)	MSFMP	Fuel management panel
(7)	MSECP	Environment control panel
(8)	MSCPI	Cabin pressure altitude gage
(9)	MSLØQ	Liquid Oxygen quantity gage
(10)	MSHP	Hydraulic panel (utility power control)
(11)	MSAHC	Arresting hook control
(12)	MSRAT	Ram air turbine (RAT) handle
(13)	MSEBI	Emergency brake indicator
(14)	MSMLS	Master light switch
(15)	MSAS	Anti-skid

Defense/Countermeasure (DCM)

(1)	DCMSCP	Defense System Control Panel
(2)	DCMRHAWI	RHAW (Radar Homing and Warning) operation indicator
(3)	DCMLWI	Launch Warning Indicator
(4)	DCMEDSP	Electronic Defensive Status (EDS) panel

Advisory and Caution Annunciator (ACA)

(1)	ACAMC	Master caution (reset)
(2)	ACAFFL	Engine (1) fire
(3)	ACAFFR	Engine (2) fire
(4)	ACALP	Light panel (ACA panel)
(5)	ACALFW	Low fuel
(6)	ACAWW	Wheels
(7)	ACALAW	Low altitude warning

Weapon System (WS)

(1)	WSAIS	Weapon monitor and select - aim interlock switch
(2)	WSMP	Weapon management panel (centroid)
(3)	WSASP	Attack select panel
(4)	WSMAS	Master arm switch
(5)	WSGS	Gun switch
(6)	WSWS	Weapon status
(7)	WSSPS	Special weapon control switch
(8)	WSSPAI	Special weapon arm indicator
(9)	WSEJ	Emergency jettison
(10)	WSLLLTV	LLLTV - Laser control panel

Communication, Navigation and Identification (CNI)

(1)	CNIMPS	Integrated CNI Control Panel (master power switch)
(2)	CNIMSS	Integrated CNI Control Panel (MIC selector switch)
(3)	CNITS	Integrated CNI Control Panel (transceiver selector)
(4)	CNIICSP	ICS control panel
(5)	CNIRBC	Radar beacon control panel
(6)	CNIDLR	Data link readout
(7)	CNIDLC	Data link command
(8)	CNINMS	Navigation control panel _NAV mode selector
(9)	CNILØS	Navigation Control Panel - LORAN Operate Switch
(10)	CNIDSS	Navigation Control Panel - Destination select switch
(11)	CNINDP	Navigation display panel
(12)	CNIHTTD	Heading and TTD (time to destination) switch
(13)	CNIMDS	Map and data storage locker

Appendix IX

FORTRAN IV SUBROUTINE MINUM
(MA-077)

SUBJECT   FORTRAN II, IV Subroutines MINUM

PURPOSE:

The subroutine MINUM determines the minimum value of a function of several para-
meters. This optimization technique is applicable to a more general class of
problems than many of the existing optimization subroutines. The function
(1) may have discontinuities, (2) may have several relative minima and (3) need
not be differentiable.

METHOD:

A complete description of the algorithm for MINUM is given in Ref. 1. Only a
brief outline of the computational technique is given here.

The minimum value of a function is determined by evaluating the function for many
values of the parameters (parameters must be in 0-1 interval). The following four
methods are used to change the parameter values.

(1)   a gradient step (GR) or a boundary gradient step (BG)

(2)   a random direction step (RD). If this step is unsuccessful, a step
       in the opposite direction will be taken (ND)

(3)   an average direction step (AV). The direction for this step is the
       average of the directions for the last five successful tries.

(4)   a jump step (JS). The values of the parameter are random numbers.

The step size for steps 1 to 3 is dependent on the number of successes and fail-ures in previous steps. The relative frequencies for each of these 4 steps is specified by the user.

DISCUSSION:

Several applications for this subroutine and the construction of the merit function are discussed in Ref. 1. Some simple examples are presented here.

Example 1 - Step Function

The merit function for this example is the step function given by

$$F(x_1,x_2) = \begin{cases} 0 & \text{if} \quad .9 \le x_1 \quad \text{and} \\ & \quad .5 \le x_2 \le .7 \\ 1 & \text{otherwise} \end{cases}$$

This function is illustrated in Fig. 1.
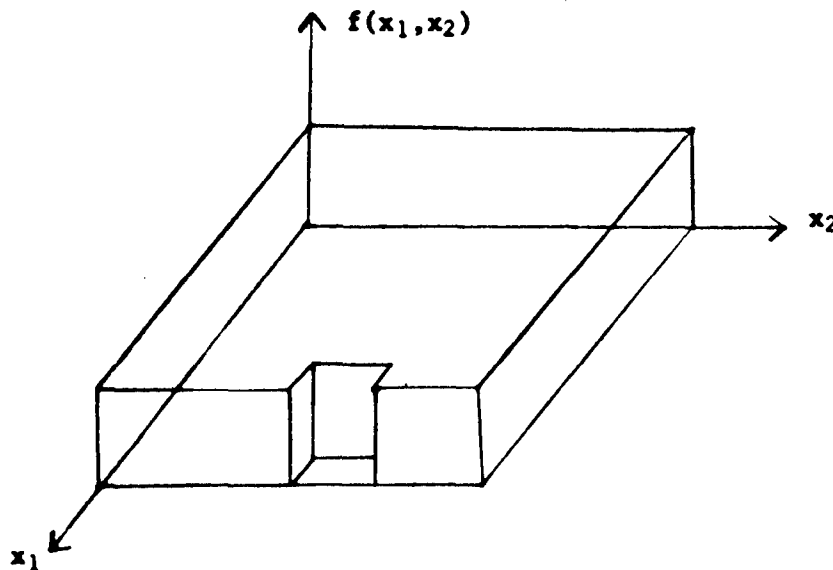
FIGURE 1

Merit Function for Ex. 1

For this example only jump steps were requested. The other 3 types of steps all require longer computation times and could not be expected to decrease the number of steps required. The gradient step is the least desirable step for this type of function. When a gradient step is requested where the function is constant in the neighborhood of the current position, the function is evaluated n (no. of parameters) times, a test is made, the gradient step is abandoned, and another type of step is tried.

The total number of steps requested for this problem was 200 and the minimum was found for the first time on step 108. The 200 points, (x_1, x_2) , where the function was evaluated by MINUM are shown in Fig. 2.
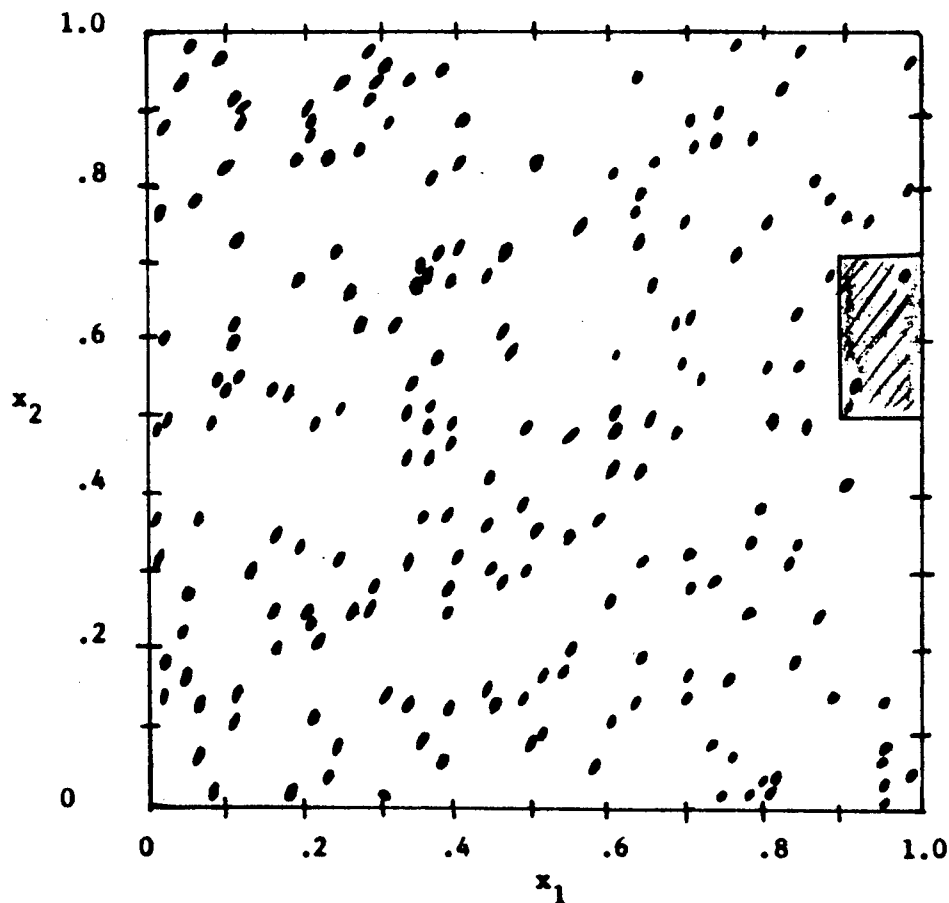
FIGURE 2

Function Evaluation - Ex. 1

The probability of finding the minimum for this particular function on a single step is $Pr = (.1)(.2) = .02$. Thus, in 200 steps the average number of times the point (x_1, x_2) will be in the minimum region is 4 .

The probability of finding the minimum for this type of function will decrease as the number of parameters increases. For a function of 10 parameters, where the sides of the minimum region have length .5 , the probability of finding the minimum on a single step is $(.5)^{10} = .001$. Thus, on the average, only one step in 1000 will fall in the minimum region for this function. This demonstrates the importance of defining the parameters so only the region of interest is searched.

Example 2 - Exponential-Cosine Function

The merit function for this example is

$$f(r) = e^{-2r} \cos (6\pi r)$$

where the radius r is $\sqrt{x_1^2 + x_2^2}$. This function is illustrated in Fig. 3. A summary of the printout from MINUM is given in Table 1 for the first 90 steps. The radii at which the function was evaluated in these 90 steps are indicated in Fig. 3.

Some of the search characteristics exhibited in Table 1 are typical for MINUM.

(a) The relative frequency of random direction steps will always be greater than requested (except for the case when the requested frequency is zero). The requested relative frequencies for this example were 12.5%, 37.5%, 25% and 25% for RD, GR, AV and JS steps

Figure 3

Merit Function – Ex. 2

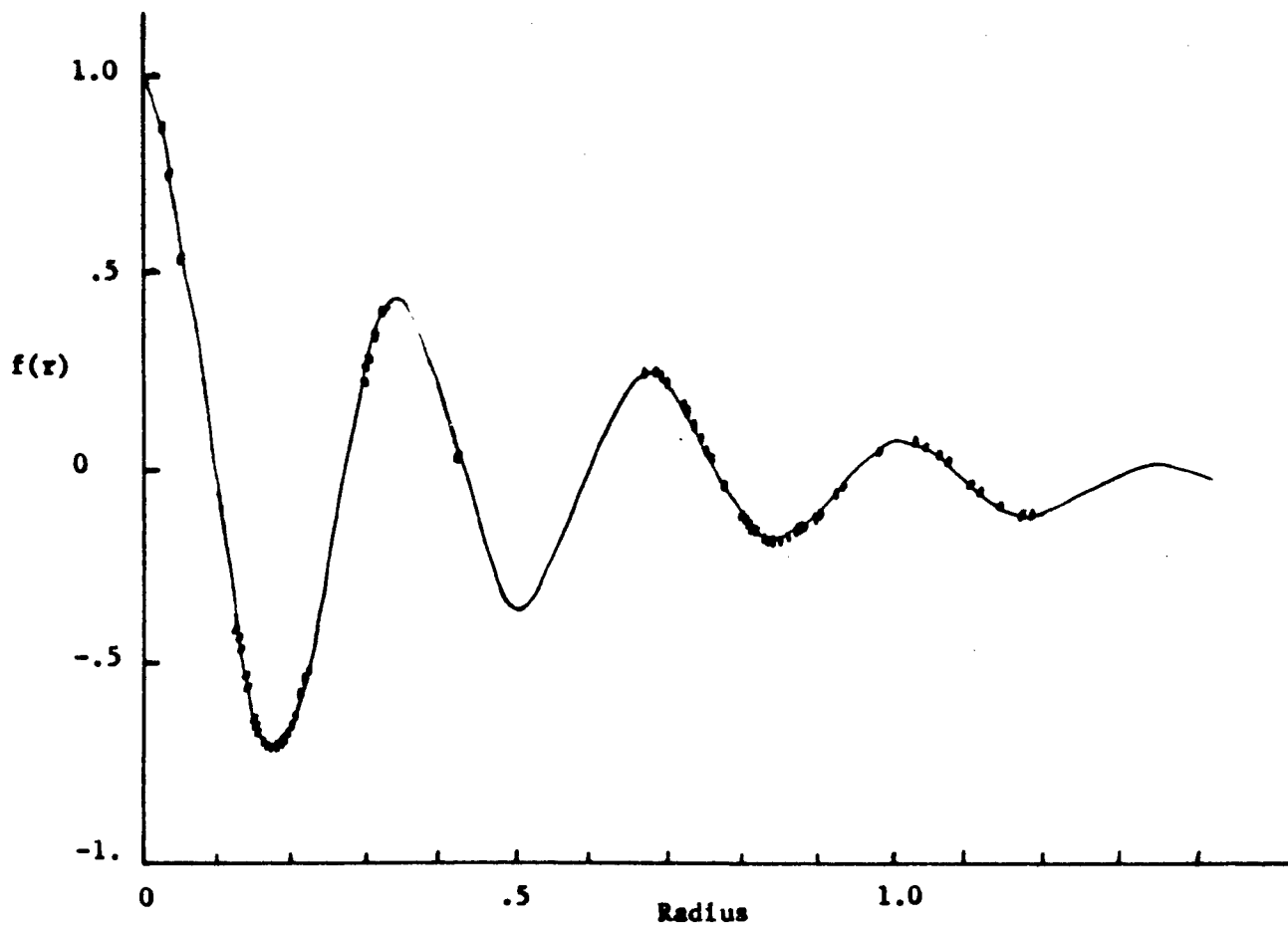| Step No.	Step Type	Current F	Radius	Best F	Step No.	Step Type	Current F	Radius	Best F
1	RD	-.014	1.090	-.014	46	ND	-.458	.116	
	RD	-.063	1.116	-.063		RD	-.410	.112	
	RD	-.097	1.160	-.097		ND	.285	.278	
	RD	-.032	1.099			RD	.865	.023	
5	AV	-.032	1.099		50	ND	.435	.298	
	RD	.097	1.156			RD	-.679	.143	
	ND	.115	1.025			ND	-.661	.183	
	JS	-.162	.857	-.162		AV	-.562	.198	
	AV	-.177	.808	-.177		GR	-.392	.110	
10	AV	-.119	.782		55	RD	-.701	.174	
	GR	.254	.677			ND	-.715	.155	
	AV	-.119	.782			JS	.107	.725	
	RD	-.072	.893			RD	-.578	.196	
	ND	.038	.741			ND	-.556	.126	
15	RD	.066	.940		60	JS	.264	.656	
	ND	.249	.680			JS	.098	1.036	
	RD	.083	.730			JS	-.126	.874	
	ND	-.046	.902			RD	-.527	.202	
	RD	.260	.671			ND	-.451	.115	
20	ND	-.167	.854		65	AV	-.709	.170	
	RD	-.166	.855			JS	.048	1.061	
	ND	-.047	.761			GR	-.673	.142	
	RD	-.190	.830	-.190		RD	-.714	.168	
	RD	-.166	.855			ND	-.697	.148	
25	AV	-.186	.818		70	JS	-.097	1.158	
	GR	-.124	.784			JS	.061	1.302	
	RD	-.181	.812			RD	-.719	.157	
	ND	-.172	.851			ND	-.720	.159	-.7200
	RD	-.152	.775			ND	-.720	.162	-.7204
30	ND	-.141	.868		75	ND	-.716	.167	
	RD	-.130	.872			RD	-.692	.146	
	ND	-.140	.790			ND	-.685	.178	
	RD	-.121	.876			AV	-.700	.174	
	ND	-.129	.786			JS	.049	.739	
35	JS	-.719	.157	-.719	80	GR	-.693	.146	
	RD	.416	.295			RD	-.721	.161	-.7206
	ND	.533	.050			RD	-.720	.161	
	AV	.346	.285			AV	-.701	.173	
	JS	-.179	.846			GR	-.688	.145	
40	JS	.026	.414		85	RD	-.720	.160	
	GR	.892	.020			ND	-.720	.164	
	JS	-.097	1.155			RD	-.692	.146	
	RD	.233	.272			ND	-.693	.176	
	ND	-.551	.125			JS	.131	.719	
45	RD	.270	.276		90	RD	-.700	.174	

TABLE 1

MINUM Results for Exponential Cosine Function

respectively. The observed frequencies for these 4 types of steps were 69%, 6.5%, 9% and 15.5% after 200 steps.

(b) Unless the region of the minimum is large, the random direction steps will be more successful than the gradient steps at the beginning of the search. The step length is always the maximum length at the start of the search and the gradient steps tend to overshoot the relative minima. When there are many successive unsuccessful steps, the step length will be decreased and the gradient step will be more successful.

(c) During the initial steps the average direction steps are usually as successful as the random direction steps. The directions for the last five successful steps are used to determine the direction for this step and since the successful steps initially occur in different regions, the average direction is usually not an optimum direction.

(d) After the initial steps a random direction step (RD) is nearly always followed by a negative direction step (ND). This in part accounts for the higher observed relative frequencies for the random direction step.

For most applications, little will be known about the merit function and thus it will usually be difficult to determine how to reduce the number of steps required for finding a minimum. This is not a problem for the cases where the merit function is easily evaluated. However, in the cases where the number of parameters is large and where the evaluation of the merit function requires

a long computer time, it is desirable to use MINUM as efficiently as possible. For these cases only the region of interest for each parameter should be transformed to the 0-1 interval. It will usually be difficult to determine the relative frequencies for the 4 types of steps which will provide the most efficient search. However, a careful examination of the step-by-step behavior of MINUM will in any times provide the information needed to improve the choice of the relative frequencies.

USAGE: DIMENSION   PARM(N),A(7*N)

CALL MINUM (N,PARM,A,FUN,IU,IX,ISTP,IPRINT,JR,JG,JA,JJ)

INPUTS: N - no. of parameters

PARM(I) - array of starting values for parameters. Must be between 0. and 1. .

A - array for working storage. Must have size at least 7*N.

FUN - name of function to be minimized (requires EXTERNAL card in calling program.)

IU - start multiplier for random no. generator RAND. IU must be odd.

IX - fixed multiplier for random no. generator. IX must be = $8n \pm 3$ where n is an arbitrary integer and IX should be near $2**X$ where X = 17.5 for the 7094 and X = 24 for the 6600.

ISTP - total no. of steps to be tried

IPRINT - modulus of steps when printing desired. If IPRINT is negative the history information is suppressed.

JR - unnormalized frequency of use of random steps.

JG - unnormalized frequency of use of gradient steps

JA - unnormalized frequency of use of average steps

JJ - unnormalized frequency of use of jump steps

OUTPUTS:

PARM(I) - the values of the parameters which gave the best minimum

IU - final value of random no. multiplier

PRINTED OUTPUT:

The subroutine MINUM prints intermediate values of the parameters and the corresponding function value. The initial values are always printed. Each IPRINT steps MINUM will print the following information.

(1) No. of steps taken

(2) Current value of best minimum

(3) Values of parameters for best minimum

(4) Current step size.

If the history information has not been suppressed (IPRINT > 0), the following information is also printed each IPRINT steps.

(1) the current parameter values

(2) the current function value

(3) the current type step

(4) the no. of successful and the no. of unsuccessful trials for each of the 4 types of step

(5) the total successful distance for each the average, random direction, and gradient steps.

(6) the direction cosines for current step (if not jumpstep)

After the final step the value of the current best minimum and the
corresponding parameter values are printed.

SUBPROGRAMS:

Subroutine RAND - PS419,PS-419A   Function FUN   which evaluates
function to be minimized.

ERROR RETURN:

None

RESTRICTIONS:

MINUM may be used for several purposes in one program only if calls
to MINUM are made serially. Nested calls to MINUM (i.e., MINUM
calls FUN which in turn calls MINUM to evaluate function) are not
allowed.

SAMPLE PROBLEM:

The source program and the printout for the circuit design problem
described in Ref. 1 are given here. Two runs were made, each having
different initial values. The 6600 central processor time was 3.1
sec. for each run.

```
FORTRAN IV PROGRAM MZZY(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
       C
       C   MAIN PROGRAM FOR MINUM TEST PROBLEM 2
       C
000043        DIMENSION X(4),TEMP(28)
000043        COMMON HPI,PI,TWOPI,THETA(10)
000043   F    FSS
000043        PI=3.1415926536
000045        HPI= PI/2,
000047        TWOPI= 2.*PI
000050        N=4
000051        IU=3746531
000052        IX=16777213
000054     5 FORMAT(6I10)
000054    10 FORMAT(6F10.0)
000054        READ(5,5)   NCASE
000061        DO 999  IJK=1,NCASE
000063   100 READ(5,5)   IPRINT,ISTEPS
000072        READ(5,5)   JR,JG,JA,JJ
000105        READ(5,10)  A1,B1,A2,B2
000120        X(1) = 1./(1.+A1)
000123        X(2) = 1./(1.+B1)
000126        X(3) = 1./(1.+A2)
000130        X(4) = 1./(1.+B2)
000133        DM=FSS(X)
000135        WRITE(6,903)  (THETA(I),I=1,10),A1,B1,A2,B2
000156        CALL MINUM(N,X,TEMP,FSS,IU,IX,ISTEPS,IPRINT,JR,JG,JA,JJ)
000170        DM=FSS(X)
000173        A1= 1./X(1) -1.
000175        B1= 1./X(2) -1.
000177        A2= 1./X(3) -1.
000201        B2= 1./X(4) -1.
000203        WRITE(6,901)   DM
000210   901 FORMAT(1H1,11HBEST MIN = ,E20.10)
000210   900 FORMAT(1H ,10X,6HTHETAS/1H ,10F12.4/1H-,10X,2HA1,18X,2HB1,18X,2HA2
              Z,18X,2HB2/4E20.7)
000210   903 FORMAT(1H1,10X,6HTHETAS/1H ,10F12.4/1H-,10X,2HA1,18X,2HB1,18X,2HA2
              Z,18X,2HB2/4E20.7)
000210   999 WRITE(6,900) (THETA(I),I=1,10),A1,B1,A2,B2
000234        STOP
000235        END
```

```
                        FORTRAN IV FUNCTION FSS(X)
000417          DIMENSION X(4),FNM(2),DEN(2),ALF(2)
000417          COMMON  HPI,PI,TWOPI,THETA(10)
000417          FSS =0.
000420          DO 100  N=1,10
000421          FN=N
000422          FNM(1)= FN*(1.-1./X(1))
000425          DEN(1)= 1./X(2) - 1. -FN*FN
000431          FNM(2)= FN*(1.-1./X(3))
000434          DEN(2)= 1./X(4) -1. -FN*FN
        C
        C  COMPUTE    ALF(I)  SO THAT BETWEEN  0   AND TWOPI
        C
000437          DO 50 I=1,2
000441          IF(DEN(I)) 10,2,1
000443        1 IF(FNM(I)) 20,21,21
000446        2 IF(FNM(I))  30,31,32
000451       10 ALF(I)=PI +ATAN(FNM(I)/DEN(I))
000461          GO TO 50
000461       20 ALF(I)=TWOPI + ATAN(FNM(I)/DEN(I))
000471          GO TO 50
000471       21 ALF(I)= ATAN(FNM(I)/DEN(I))
000500          GO TO 50
000500       30 ALF(I) = TWOPI-HPI
000503          GO TO 50
000504       31 ALF(I)=0.
000506          GO TO 50
000506       32 ALF(I)=HPI
000510       50 CONTINUE
000512          THETA(N)= (ALF(1)+ALF(2))*2.
000516       60 IF(THETA(N)*FN.LE.PI)  GO TO 100
000522          THETA(N)=THETA(N)-TWOPI
000525          GO TO 60
000525      100 FSS=FSS+(11.-FN)*ABS(THETA(N)*FN)
000537          RETURN
000540          END
```

THETAS
-1.0134 -2.0000 -2.9702 -3.9679 -4.0219 -4.9996 -6.6739 -7.7360 -8.3033 -8.7454

A1   B1   A2   B2
1.3860000E+01   3.8510000E+01   5.8100000E+00   3.8600000E+01

STARTING MINIMUM= 45.411894E-01 AT
67.385465E-03 25.310048E-03 14.684280E-02 25.252525E-03

FREQUENCIES= 20 RANDOM 60 GRADIENT 20 AVERAGE 20 JUMP

BEST MINIMUM = 45.411894E-01 AFTER 20 STEPS (CURRENT STEP SIZE 66.666667E-03)
BEST POSITION .06738544 .02531005 .16684288 .02525253
            GD   AD   JS   RATIO   MINIMUM   RD
RD 0/ 13 -.2870   0/ 5   .3333 66.464855E+00 0.   0.   0.   AD
DIR COS -.2870   -.6356   2   0/ -.4019   -.5213
CURRENT POSITION
.06825467   .01706156   .11405213   .00950192

BEST MINIMUM = 45.411894E-01 AFTER 40 STEPS (CURRENT STEP SIZE 22.222222E-03)
BEST POSITION .06738544 .02531005 .16684288 .02525253
            GD   AD   JS   RATIO   MINIMUM   RD
RD 0/ 28   .0615   0/ 9   .3333 63.017430E+00 0.   0.   0.   AD
NO 0/   -.0406   3   0/ .7652   -.6412
DIR COS .0615
CURRENT POSITION
.06930844   .02440712   .16384642   .01100333

BEST MINIMUM = 45.411894E-01 AFTER 60 STEPS (CURRENT STEP SIZE 24.691356E-04)
BEST POSITION .06738544 .02531005 .16684288 .02525253
            GD   AD   JS   RATIO   MINIMUM   RD
RD 0/ 42 -.0866   0/ 13   .3333 12.053036E+00 0.   0.   0.   AD
GR 0/   -.5785   5   0/ -.0314   -.8107
DIR COS -.0866
CURRENT POSITION
.06717663   .02388168   .16676531   .02325083

BEST MINIMUM = 45.411894E-01 AFTER 80 STEPS (CURRENT STEP SIZE 82.304527E-05)
BEST POSITION .06738544 .02531005 .16684288 .02525253
            GD   AD   JS   RATIO   MINIMUM   RD
RD 0/ 59 -.6016   0/ 15   .3333 46.634411E-01 0.   0.   0.   AD
DIR COS -.6016 .0029   6   0/ -.7591   -.2491
CURRENT POSITION
.06689047   .02531240   .16621808   .02504752

D162-10127-1
227

BEST MINIMUM = 45.411896E-01 AFTER   100 STEPS (CURRENT STEP SIZE 91.449474E-06)
BEST POSITION .06738544 .02531005 .16864288 .02525253

    RD   GD   JS   RATIO   MINIMUM   RD   GD   AD
    0/   75   0/   0   18   .3333 45.670527E-01   0.   0.   0.
DIR COS .0552   .0463 .8422 -.5343
CURRENT POSITION .06739049   .02531428   .14691990   .02520367

BEST MINIMUM = 45.381896E-01 AFTER   120 STEPS (CURRENT STEP SIZE 30.483158E-06)
BEST POSITION .06741421 .02532639 .16869364 .02524581

    RD   GD   JS   RATIO   MINIMUM   RD   GD   AD
    2/   90   0/   0   21   0.   45.381896E-01 60.966E-06 0.   0.
DIR COS .4719   .2681 .8327 -.1101
CURRENT POSITION .06742295

BEST MINIMUM = 45.374176E-01 AFTER   160 STEPS (CURRENT STEP SIZE 10.161053E-06)
BEST POSITION .06742453 .02534368 .16691746 .02524143

    GR   GD   JS   RATIO   MINIMUM   RD   GD   AD
    3/ 104   1/   11   23   0.   45.374176E-01 91.449E-06 10.161E-06 0.
DIR COS -.4007   .8975 -.1542 -.1009
CURRENT POSITION

BEST MINIMUM = 45.208649E-01 AFTER   160 STEPS (CURRENT STEP SIZE 27.434842E-05)
BEST POSITION .06817660 .02599096 .16835271 .02502800

    AV   GD   JS   RATIO   MINIMUM   RD   GD   AD
    3/ 104   1/   19   21   0.   45.208649E-01 91.449E-06 10.161E-06 17.579E-04
DIR COS .4278   .3682 .8165 .1214
CURRENT POSITION

BEST MINIMUM = 44.398371E-01 AFTER   180 STEPS (CURRENT STEP SIZE 82.304527E-05)
BEST POSITION .07005461 .02760728 .15193668 .02449503

    RD   GD   JS   RATIO   MINIMUM   RD   GD   AD
    3/ 113   1/   25   24   .3333 48.019000E-01 91.449E-06 10.161E-06 61.474E-04
DIR COS -.3696   -.3006 .1761 .8614
CURRENT POSITION .06975042   .02735986   .15208159   .02520401

BEST MINIMUM = 44.342618E-01 AFTER   200 STEPS (CURRENT STEP SIZE 82.304527E-05)
BEST POSITION .06994940 .02729060 .15118440 .02450610

    RD   GD   JS   RATIO   MINIMUM   RD   GD   AD
    4/ 127   1/   25   27   .3333 48.477990E-01 91.449E-05 10.161E-06 61.474E-04
DIR COS .0156   -.0462 -.0313 -.3983
CURRENT POSITION .07070295   .02775258   .15115866   .02417829

BEST MINIMUM = 43.777940E-01 AFTER   220 STEPS (CURRENT STEP SIZE 27.434842E-05)
BEST POSITION .07017109 .02715750 .15119294 .02462818

    RD   GD   JS   RATIO   MINIMUM   RD   GD   AD
    6/ 141   1/   25   28   .3333 44.962856E-01 91.449E-05 10.161E-06 61.474E-04
DIR COS .1048   .3715 -.7557 .5291
CURRENT POSITION .07010905   .07725941   .15098562   .02477334

BEST MINIMUM = 43.777960E-01 AFTER   240 STEPS (CURRENT STEP SIZE 30.483158E-06)
BEST POSITION .07017109 .07715750 .15119294 .02462818

ND   6/   156   1/   25/   JS   AD   RATIO   MINIMUM   RD   GD   AD
DIR COS .2928   .0155 18 .0581   0/   31   .3333 43.898477E-01 91.449E-05 10.161E-06 61.474E-04
CURRENT POSITION   .9544   .02445728
   .07014002   .02715797   .15119465

BEST MINIMUM = 43.774461E-01 AFTER   260 STEPS (CURRENT STEP SIZE 10.161053E-06)
BEST POSITION .07016556 .02715481 .15118816 .02463470

ND   7/   169   1/   70/ 25/   JS   AD   RATIO   MINIMUM   RD   GD   AD
DIR COS -.5447   -.2650 -.4704 0/   34   .3333 43.811910E-01 91.449E-05 10.161E-06 61.474E-04
CURRENT POSITION   .641R   .02464122
   .07014002   .02715212   .15118338

BEST MINIMUM = 43.741790E-01 AFTER   280 STEPS (CURRENT STEP SIZE 30.483158E-06)
BEST POSITION .07022206 .02719214 .15127620 .02462220

ND   7/   177   1/   21/ 32/   JS   AD   RATIO   MINIMUM   RD   GD   AD
DIR COS .1191   .8748 .2324 -.4081 0/   36   .3333 43.769395E-01 91.449E-05 10.161E-06 62.592E-04
CURRENT POSITION   .02460976
   -0.22569   .02721881   .15128328

BEST MINIMUM = 43.741790E-01 AFTER   300 STEPS (CURRENT STEP SIZE 33.870176E-07)
BEST POSITION .07022206 .02719214 .15127620 .02462220

ND   7/   189   1/   23/ 32/   JS   AD   RATIO   MINIMUM   RD   GD   AD
DIR COS -.7432   .2793 .4587 -.3990 0/   40   .3333 43.747583E-01 91.449E-05 10.161E-06 62.592E-04
CURRENT POSITION   .93287805   .02462085
   .07021954   .02719309   .15127775

BEST MINIMUM = 43.740192E-01 AFTER   320 STEPS (CURRENT STEP SIZE 11.290059E-07)
BEST POSITION .07021624 .02718933 .15127455 .02462358

JS   9/   201   1/   25/ 32/   JS   AD   RATIO   MINIMUM   RD   GD   AD
DIR COS .6690   -.6516 .3265 .1460 0/   44   .3333 10.484930E-01 91.449E-05 10.161E-06 62.592E-04
CURRENT POSITION   .57365240   .51241781
   .07021805   .67174087   .15127414

BEST MINIMUM = 43.739475E-01 AFTER   340 STEPS (CURRENT STEP SIZE 37.633528E-08)
BEST POSITION .07021664 .02718944 .15127401 .02462351

ND   11/   217   1/   26/ 32/   JS   AD   RATIO   MINIMUM   RD   GD   AD
DIR COS .6690   -.6516 .3265 .1460 0/   44   .3333 43.741330E-01 91.449E-05 10.161E-06 62.592E-04
CURRENT POSITION   .57365240   .02462356
   .07021693   .02718920   .15127414

BEST MINIMUM = 43.739875E-01 AFTER   360 STEPS (CURRENT STEP SIZE 12.544509E-08)
BEST POSITION .07021668 .02718944 .15127401 .02462351

```
              RD              GD            AD
NO    11/   233     1/    27     32/    54     RATIO     MINIMUM         RD            GD            AD
DIR COS  -.4476     .6765    -.3670    -.4584    .3333  43.739907E-01  91.449E-05  10.161E-06  62.592E-04
CURRENT POSITION
       .07021662        .02718953        .15127397        .02462345
```

BEST MINIMUM = 43.739422E-01 AFTER   380 STEPS (CURRENT STEP SIZE 37.633528E-08)
BEST POSITION .07021678 .02718959 .15127439 .02462337

```
              RD              GD            AD
RD    13/   244     1/    28     37/    61     RATIO     MINIMUM         RD            GD            AD
DIR COS  -.2868    -.8617    -.1482    -.3915    .3333  43.743477E-01  91.449E-05  10.161E-06  62.598E-04
CURRENT POSITION
       .07021667        .02718927        .15127434        .02462322
```

BEST MINIMUM = 43.739399E-01 AFTER   400 STEPS (CURRENT STEP SIZE 12.544509E-08)
BEST POSITION .07021678 .02718964 .15127428 .02462333

```
              RD              GD            AD
RD    14/   259     1/    29     37/    63     RATIO     MINIMUM         RD            GD            AD
DIR COS  -.0838    -.0639     .1304     .985A     .3333  43.740018E-01  91.449E-05  10.161E-06  62.598E-04
CURRENT POSITION
       .07021677        .02718963        .15127430        .02462346
IN  40ISTEPS MIN=.  43.739399E-01 AT
70.216777E-03  27.189639E-03  15.127428E-02  24.623335E-03
```

BEST MIN = 4.3739901200E+00

```
    THETAS
-1.0162       -2.0000       -2.9608       -3.9468       -5.0000       -6.0750       -7.0350       -7.7994       -8.3817       -8.8306
    A1            A2                                        A2
1.3241611E+01  3.5778715E+01  3.5778715E+01  5.6105089E+00  5.6105089E+00  3.9611003E+01
```

```
THETAS
-.9018  -1.8586  -2.9074  -4.0373  -5.1706  -6.2063  -7.0801  -7.7913  -8.3666  -8.8297
A1                A2                R2
1.0000000E+01  4.7000000E+01  7.0000000E+00  3.0000000E+01


STARTING MINIMUM= 86.7909910E-01 AT
90.9090910F-03  20.8333333E-03  12.5000000E-02  32.2580645E-03

FREQUENCIES= 20 RANDOM  60 GRADIENT  60 AVERAGE  20 JUMP


BEST MINIMUM = 86.7909910E-01 AFTER  20 STEPS (CURRENT STEP SIZE  66.6666667E-03)
BEST POSITION .09090909 .02083333 .12500000 .03225806
        GD          N/       JS     RATIO     MINIMUM         RD
RD  N/  13    N/  2    N/  0    .3333  81.8571315E+00  0.     5    0.    86    0.    AD
DIR COS  -.2870  -.6356  -.4919  -.5217
CURRENT POSITION
.07177832  .07153828  .09220926  .00269638


BEST MINIMUM = 86.7909910E-01 AFTER  60 STEPS (CURRENT STEP SIZE  22.2222222E-03)
BEST POSITION .09090909 .02083333 .12500000 .03225806
        GD          N/       JS     RATIO     MINIMUM         RD
NO  N/  28    N/  3    N/  0    .3333  46.813636E+00  0.     9    0.    86    0.    AD
DIR COS  .0415  -.0606  .7652  -.6412
CURRENT POSITION
.09183209  .01993040  .14200354  .01800887


BEST MINIMUM = 72.6310886E-01 AFTER  80 STEPS (CURRENT STEP SIZE  74.0740746E-04)
BEST POSITION .08678166 .02371421 .11056567 .02558837
        GD          N/       JS     RATIO     MINIMUM         RD
JS  4/  41    N/  4    N/  2    .3333  10.89151F+01 29.63nF-03  0.    13    0.    86    0.    AD
DIR COS  .8906  -.2187  -.3615
CURRENT POSITION
.24277721  .88801397  .68492136  .08634622


BEST MINIMUM = 66.2093957E-01 AFTER  80 STEPS (CURRENT STEP SIZE  24.6913586E-04)
BEST POSITION .08793066 .02964816 .10868127 .02081255
        GD          N/       JS     RATIO     MINIMUM         RD
NO  6/  56    N/  6    N/  4    .3333  70.666739E-01 72.0000E-01  0.    14    0.    86    0.    AD
DIR COS  -.1684  .8906  -.2187  -.3615
CURRENT POSITION
.08751491  .03164709  .10814118  .01991091
```

REST MINIMUM = 59.416693E-01 AFTER 100 STEPS (CURRENT STEP SIZE 24.69135RF-04)
REST POSITION .08732317 .02R45553 .11043729 .0222RR01

RD	A/ 69	0/ R	0/	JS 16	RATIO	MINIMUM	RD	GD	AD
DIR COS .513R	.3490	-.2174	-.7530		.3333	R6.489069F-01	32.099F-03	0.	0.

CURRENT POSITION
.0RA59179 .02931716 .11010062 .02042875

BEST MINIMUM = 58.188753E-01 AFTER 120 STEPS (CURRENT STEP SIZF 82.704527F-05)
REST POSITION .0R773R43R .02955514 .11013684 .02137757

JS	A/ 80	0/ 9	2/ 12	0/ 19	RATIO	MINIMUM	RD	GD	AD
					.3333 R9.26589RE+00	32.099F-03	0.	16.461F-04

CURRENT POSITION
.1109R565 .47560707 .1474R855 .R3460505

REST MINIMUM = 57.71164RE-01 AFTER 140 STEPS (CURRENT STEP SIZF 27.436842E-05)
BEST POSITION .0R786530 .02958851 .11033400 .02151225

JS	A/ 92	1/ 13	2/ 14	0/ 21	RATIO	MINIMUM	RD	GD	AD
					.3333 11.67259IF-01 32.099E-03	27.435F-05	16.461F-04

CURRENT POSITION
.73041277 .57615165 .96937519 .09838602

REST MINIMUM = 53.903031F-01 AFTER 160 STEPS (CURRENT STEP SIZF 24.691358F-04)
BEST POSITION .09355669 .03279595 .11R35062 .02050000

RD	16/ 102	1/ 14	R/ 22	0/ 27	RATIO	MINIMUM	RD	GD	AD
DIR COS .4711	-.3063	.7174	.411R		.3333 53.984375E-01 35.96RE-03	27.435F-05	R2.705F-04

CURRENT POSITION
.0947197R .03203963 .12012201 .02151696

REST MINIMUM = 47.234119F-01 AFTER 1R0 STEPS (CURRENT STEP SIZF 24.69135RF-04)
REST POSITION .09RR6302 .03655011 .11655221 .01906056

ND	1R/ 116	1/ 15	10/ 26	0/ 23	RATIO	MINIMUM	RD	GD	AD
DIR COS .1947	.4161	.1R36	-.R660		.3333 11.059665E+00 1R.400F-03	27.435F-05	13.169F-03

CURRENT POSITION
.09934371 .03757762 .11702962 .01690003

REST MINIMUM = 42.734077F-01 AFTER 200 STEPS (CURRENT STEP SIZF 24.69135RF-04)
REST POSITION .10687579 .04130790 .11929313 .01795R84

ND	19/ 124	1/ 16	17/ 36	0/ 24	RATIO	MINIMUM	RD	GD	AD
DIR COS .2130	.5015	-.6229	-.5613		.3333 67.919540E-01 19.232F-03	27.635F-05	22.222F-03

CURRENT POSITION
.10740175 .06254622 .11775511 .01657RR1

BEST MINIMUM = 37.565054E-01 AFTER   220 STEPS (CURRENT STEP SIZE 26.074074E-04)
BEST POSITION .13163118 .05135669 .13459099 .01652724

RD   21/   133   1/   17   24/   45   JS/   25   RATIO   MINIMUM   RD   AD
DIR COS .4955   .4010 -.5109 -.5023   -.3333 14.259578E+00 19.232E-03 27.635E-05 49.383E-03
CURRENT POSITION .13510189   .05489745   .13081522   .01280635

BEST MINIMUM = 37.565054E-01 AFTER   240 STEPS (CURRENT STEP SIZE 26.691358E-04)
BEST POSITION .13163118 .05135669 .13459099 .01652724

RD   21/   149   1/   18   24/   46   JS/   27   RATIO   MINIMUM   RD   AD
DIR COS .2101   .0437 -.5958 -.7726   .3333 10.261473E+00 19.232E-03 27.635E-05 49.383E-03
CURRENT POSITION .13194906   .05151600   .13312898   .01461966

BEST MINIMUM = 30.083638E-01 AFTER   260 STEPS (CURRENT STEP SIZE 26.691358E-04)
BEST POSITION .12851173 .05352609 .14008595 .01669654

RD   27/   165   1/   19   24/   48   JS/   28   RATIO   MINIMUM   RD   AD
DIR COS -.1983   -.4419 .8602 -.1595   .3333 40.383156E-01 39.232E-03 27.635E-05 49.383E-03
CURRENT POSITION .12802212   .05243605   .14220993   .01630280

BEST MINIMUM = 27.79059AE-01 AFTER   280 STEPS (CURRENT STEP SIZE 82.304527E-05)
BEST POSITION .12979089 .05372577 .14184569 .01660363

AV   179   1/   20   27/   53   JS/   28   RATIO   MINIMUM   RD   AD
DIR COS .6661   .3599 .6562 -.0392   0.   .79059AE-01 40.055E-03 27.635E-05 51.452E-03

BEST MINIMUM = 27.304547E-01 AFTER   300 STEPS (CURRENT STEP SIZE 82.304527E-05)
BEST POSITION .13252378 .05520674 .14455772 .01644224

ND   29/   191   1/   22   30/   50   JS/   28   RATIO   MINIMUM   RD   AD
DIR COS .1764   -.1316 -.8317 -.5098   .3333 26.359853E-01 40.055E-03 27.635E-05 55.967E-03
CURRENT POSITION .13266900   .05509839   .14387321   .01602269

BEST MINIMUM = 25.36347AE-01 AFTER   320 STEPS (CURRENT STEP SIZE 26.691358E-04)
BEST POSITION .13247563 .05631146 .14916796 .01639057

RD   30/   201   1/   23   25/   66   JS/   30   RATIO   MINIMUM   RD   AD
DIR COS .0251   -.2007 -.2086 -.9568   .3333 11.04063AE+00 40.055E-03 27.635E-05 60.082E-03
CURRENT POSITION .13253770   .05581582   .14865277   .01602801

BEST MINIMUM = 23.048861E-01 AFTER 340 STEPS (CURRENT STEP SIZE 82.704527E-05)
BEST POSITION .13145867 .05638127 .15181650 .01676372

```
                     GN              JS        RATIO          MINIMUM
AV  32/  212   1/   25   39/   73   .30   0.    23.048861E-01  40.055E-03 27.435E-05 63.374E-03
DIR COS   -.1633    .1580    .9760    .0432
```

BEST MINIMUM = 20.068641E-01 AFTER 360 STEPS (CURRENT STEP SIZE 82.704527E-05)
BEST POSITION .12956565 .05618643 .15674356 .01686978

```
                     GN              JS        RATIO          MINIMUM
RD  34/  220   1/   27   44/   R2   .31   0.    20.068641E-01  40.878E-03 27.435E-05 67.600E-03
DIR COS   -.5215    -.7025    .6843    .0041
```

BEST MINIMUM = 18.073537E-01 AFTER 380 STEPS (CURRENT STEP SIZE 82.704527E-05)
BEST POSITION .12787464 .05522772 .15922343 .01703764

```
                     GN              JS        RATIO          MINIMUM
RD  35/  231   1/   28   47/   R7   .3333   18.643076E-01  41.701E-03 27.435E-05 69.959E-03
DIR COS   -.2736    .101R    .9231    .2505
CURRENT POSITION  .12764933   .05531096   .15998316   .01724372
```

BEST MINIMUM = 17.735213E-01 AFTER 400 STEPS (CURRENT STEP SIZE 27.434862E-05)
BEST POSITION .12794650R .05527102 .15931452 .01706620

```
                     GN              JS        RATIO          MINIMUM
RD  36/  246   1/   29   4R/   91   .3333   21.690260E-01  41.701E-03 27.435E-05 70.233E-03
DIR COS    .7138    -.4245    -.3036    -.4670
CURRENT POSITION  .12814092   .05515457   .15923123   .01693807
```

IN 40 STEPS MIN = 17.735213E-01 AT
12.79450RE-02 55.271022E-03 15.931452E-02 17.064201E-03

BEST   1.7735212569E+00

```
        THETAS    1.7735212569E+00
-.0872    -2.0000   -7.0177   -6.0007   -6.0585   -5.9571   -7.0048   -9.8107   -9.4925
          R1        R2
6.8158534E+00   1.709246635E-01   5.2768919E+00   5.2768919E+00   5.7505347E+01
```

EQUIPMENT: 6600

LANGUAGE: FORTRAN IV

STORAGE:

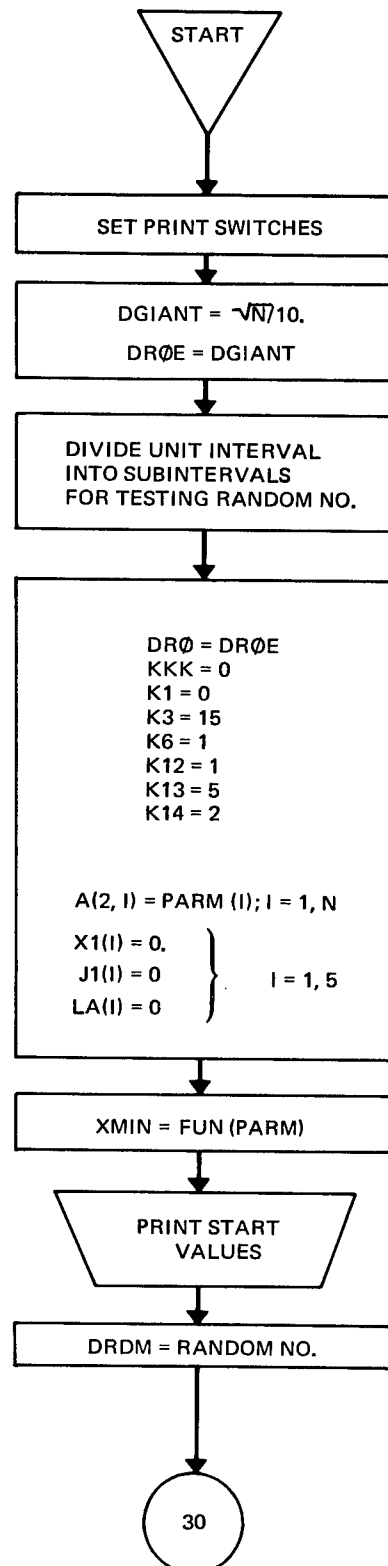|	Octal	Decimal
6600	1647	935

REFERENCES: (1) Pearson, C. E. and W. S. Willman, C Computer Program for

Design Optimization, Sperry Rand Research Center, Rpt. No.
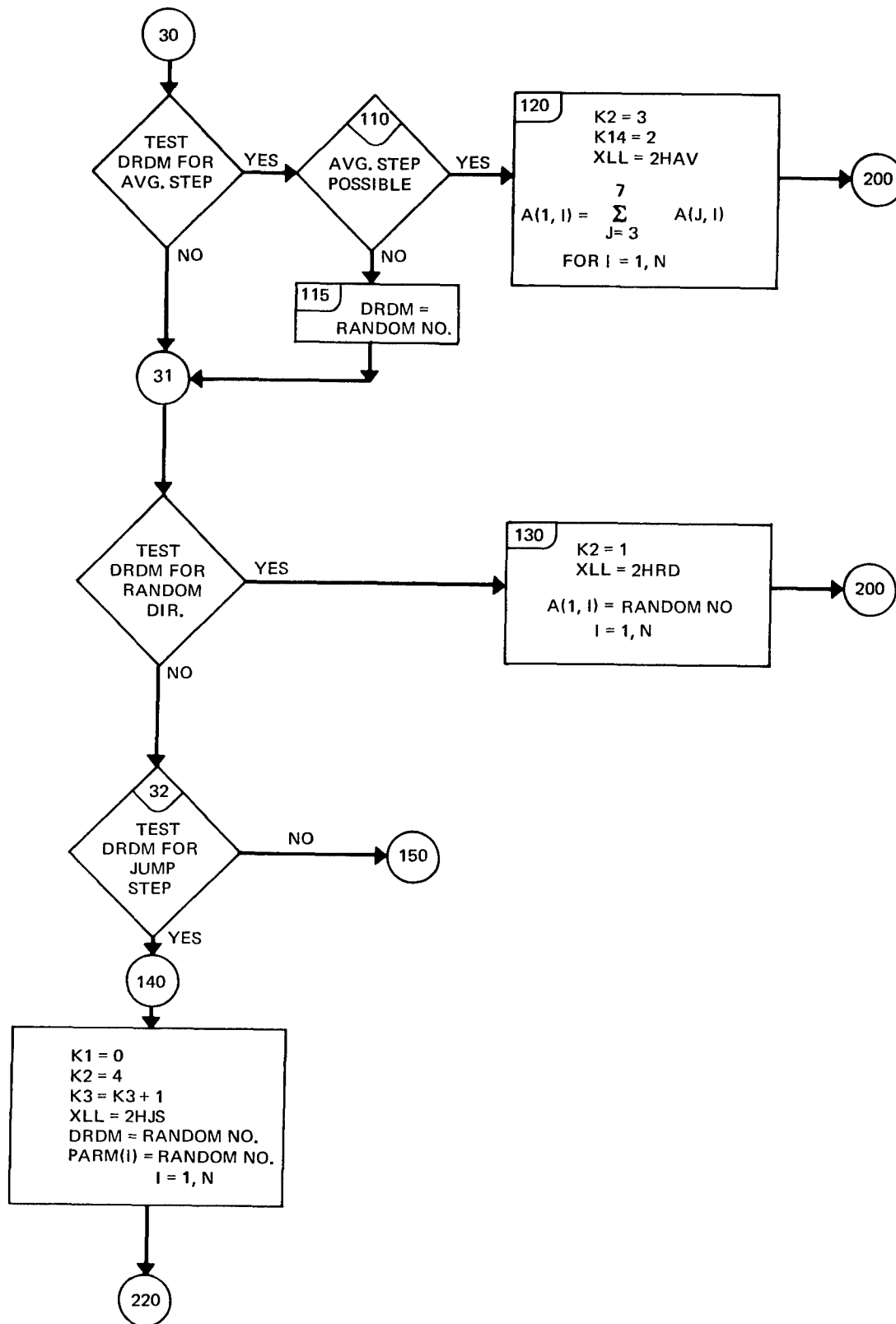
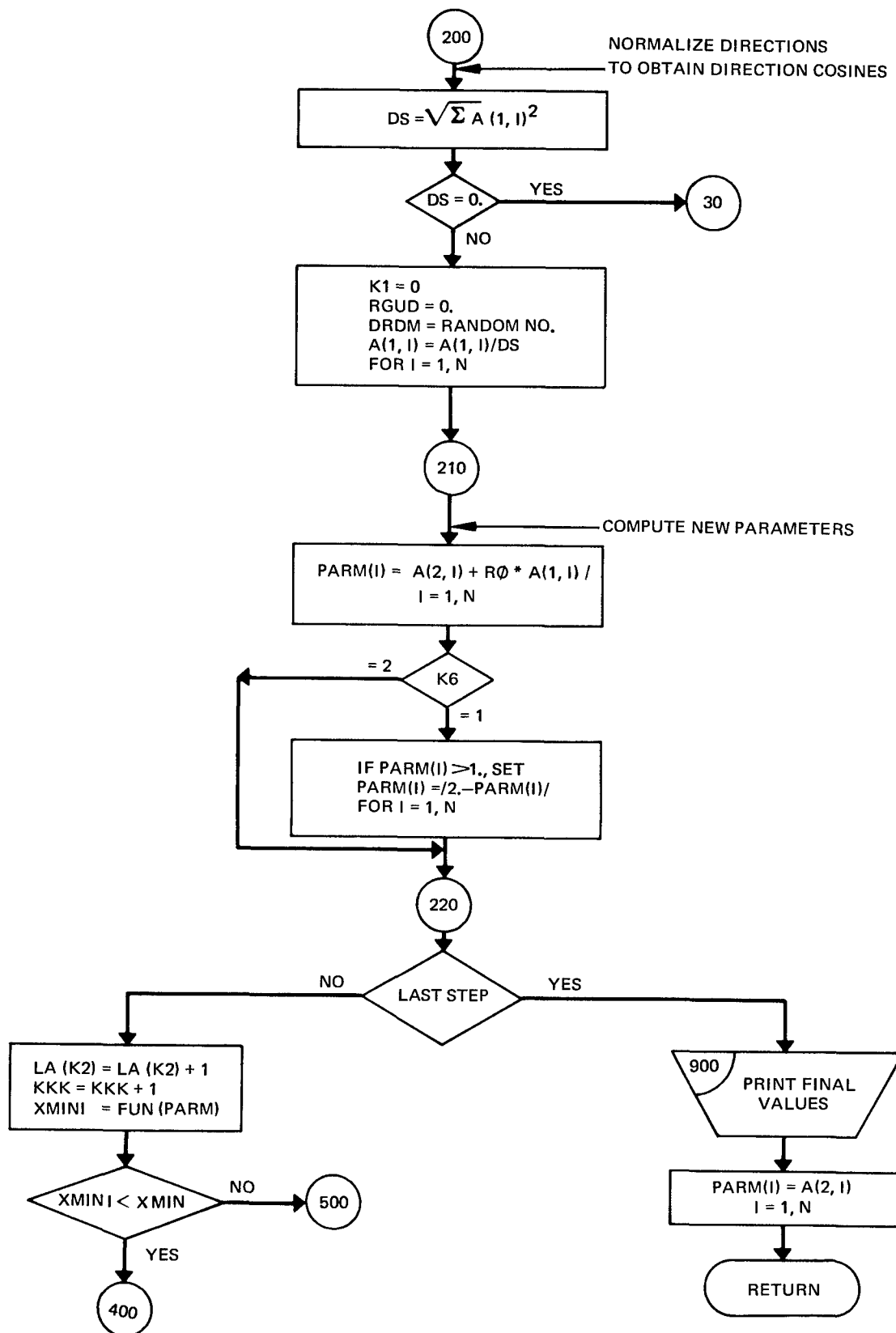SRRC-RR-65-48, June, 1965 (See author for copies)

TABLE OF BASIC SYMBOLS

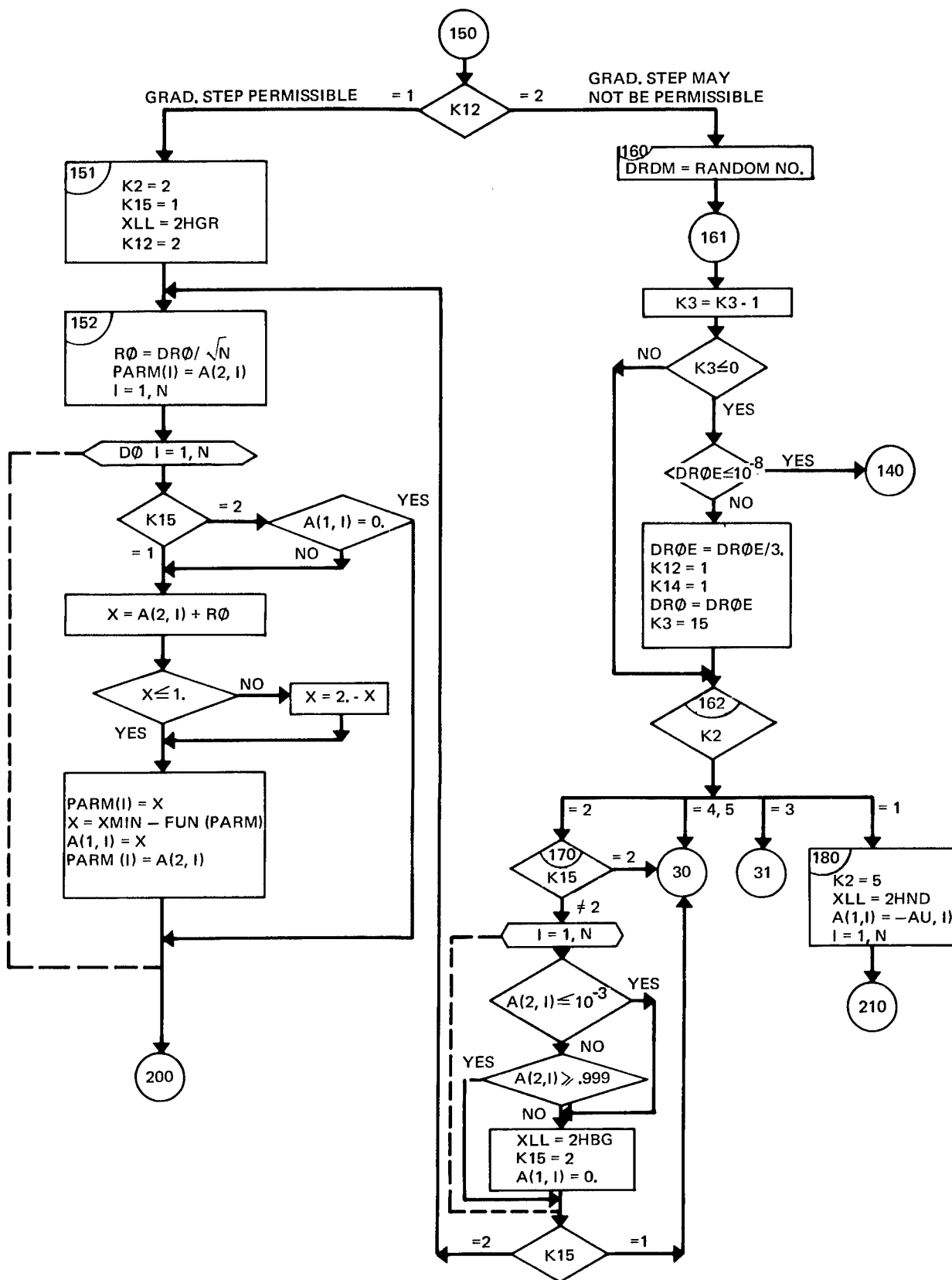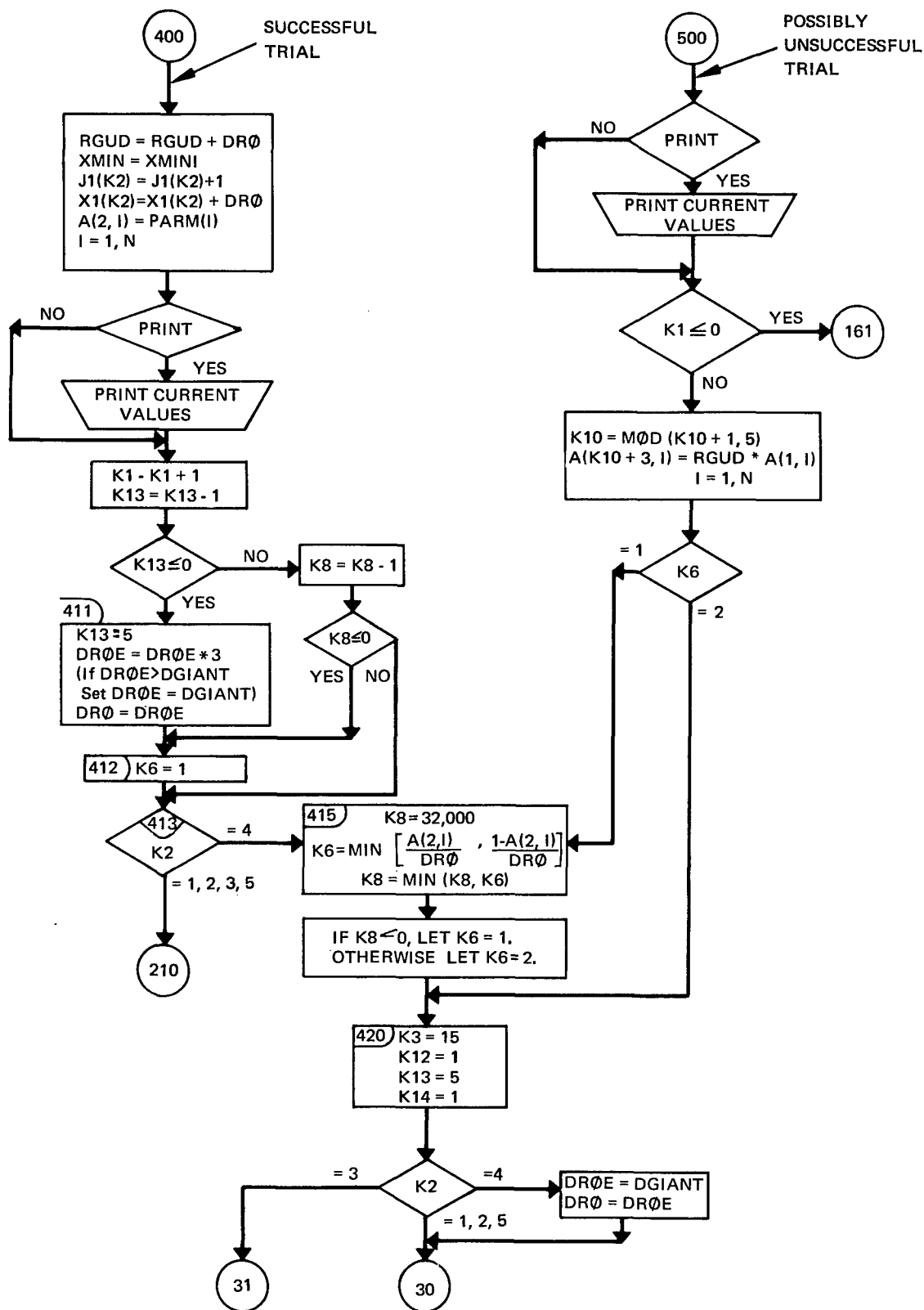|	
DGIANT	$= .1/N$
DROE	= STEP SIZE
XMIN	= BEST MINIMUM
DRO	= STEP SIZE
KKK	= NO. OF CURRENT STEPS
K1	= NO. OF SUCCESSFUL STEPS IN THIS DIRECTION
K3	= 15-NO. OF SUCCESSIVE UNSUCCESSFUL STEPS
K6	= SWITCH IF 1 SOME PARAMETER MAY BE AT THE BOUDARY
	IF 2 THERE ARE K8 STEPS TO THE BOUNDARY
K12	= SWITCH IF 1 GRADIENT IS PERMISSIBLE
	IF 2 " " NOT PERMISSIBLE
K13	= 5-NO. OF SUCCESSFUL STEPS IN THIS DIRECTION
K14	= SWITCH IF 1 AVERAGE DIRECTION IS PERMISSIBLE
	IF 2 " " " NOT PERMISSIBLE
K15	= SWITCH IF 1 NORMAL GRADIENT
	2 USE GRADIENT AT THE BOUNDARY
K2	= SWITCH IF 1 LAST TRY WAS A RANDOM TRY
	" 2 " " " " GRADIENT TRY
	" 3 " " " " AVERAGE "
	" 4 " " " " JUMP "
	" 5 " " " " NEGATIVE "
K8	= MINIMUM # OF STEPS TO BOUNDARY
DRDM	= RANDOM #
RGUD	= TOTAL GOOD DISTANCE IN THIS TRY
LA(I)	= NO. OF STEPS IN EACH OF THE FIVE TYPES OF TRIALS
J1(I)	= NO. OF SUCCESSFUL STEPS IN EACH OF THE FIVE TYPES OF TRIALS
X1(I)	= SUCCESSFUL DISTANCE IN EACH OF THE FIVE TYPES OF TRIALS
K10	= LOCATION OF LAST SUCCESSFUL PARAMETERS IN STORAGE
XLL	= HOLERITH TITLE TELLING WHAT TYPE OF TRIAL THE LAST STEP WAS
A(3,I)-A(7,I)	= STORAGE OF SUCCESSFUL DIRECTIONS. DISTANCE IN THAT DIRECTION
A(1,I)-A(2,I)	= WORKING AREAS FOR DIRECTIONS AND PARAMETERS

```
                    ▽
                  START


        ┌─────────────────────────────┐
        │      SET PRINT SWITCHES      │
        └─────────────────────────────┘


        ┌─────────────────────────────┐
        │   DGIANT = √N̄/10.            │
        │   DRØE = DGIANT              │
        └─────────────────────────────┘


        ┌─────────────────────────────┐
        │   DIVIDE UNIT INTERVAL       │
        │   INTO SUBINTERVALS          │
        │   FOR TESTING RANDOM NO.     │
        └─────────────────────────────┘


        ┌─────────────────────────────┐
        │                             │
        │         DRØ = DRØE           │
        │         KKK = 0              │
        │         K1 = 0              │
        │         K3 = 15             │
        │         K6 = 1             │
        │         K12 = 1            │
        │         K13 = 5            │
        │         K14 = 2            │
        │                             │
        │   A(2, I) = PARM (I); I = 1, N │
        │   X1(I) = 0.    ⎫           │
        │   J1(I) = 0     ⎬  I = 1, 5  │
        │   LA(I) = 0     ⎭           │
        └─────────────────────────────┘


        ┌─────────────────────────────┐
        │      XMIN = FUN (PARM)       │
        └─────────────────────────────┘


         ╱─────────────────────────╲
        ╱       PRINT START          ╲
        ╲         VALUES             ╱
         ╲───────────────────────────╱


        ┌─────────────────────────────┐
        │    DRDM = RANDOM NO.         │
        └─────────────────────────────┘


                  ( 30 )
```

150

GRAD. STEP PERMISSIBLE = 1   K12   = 2   GRAD. STEP MAY NOT BE PERMISSIBLE

151
K2 = 2
K15 = 1
XLL = 2HGR
K12 = 2

160
DRDM = RANDOM NO.

161

K3 = K3 - 1

K3 ≤ 0   NO / YES

$< DR\emptyset E ≤ 10^{-8}$ YES → 140

NO

DR$\emptyset$E = DR$\emptyset$E/3.
K12 = 1
K14 = 1
DR$\emptyset$ = DR$\emptyset$E
K3 = 15

152
R$\emptyset$ = DR$\emptyset$/ $\sqrt{N}$
PARM(I) = A(2, I)
I = 1, N

D$\emptyset$ I = 1, N

K15   = 2 → A(1, I) = 0.   YES

= 1   NO

X = A(2, I) + R$\emptyset$

X ≤ 1.   NO → X = 2. - X

YES

PARM(I) = X
X = XMIN − FUN (PARM)
A(1, I) = X
PARM (I) = A(2, I)

200

162
K2

= 2   = 4, 5   = 3   = 1

170
K15   = 2 → 30

≠ 2

I = 1, N

$A(2, I) ≤ 10^{-3}$ YES

NO

A(2,I) ≥ .999   YES

NO

XLL = 2HBG
K15 = 2
A(1, I) = 0.

K15   = 2 / = 1

30

31

180
K2 = 5
XLL = 2HND
A(1,I) = −AU, I)
I = 1, N

210

Appendix X

Subroutine Name Index

Appendix X: Subroutine name Index

*System Subroutine

+These routines are also listed under PROGRAM MAN1

*System Subroutine

+These routines are also listed under PROGRAM INTRAN

ACTIVE SHEET RECORD											
		ADDED SHEETS						ADDED SHEETS			
SHEET NUMBER	REV LTR	SHEET NUMBER	REV LTR	SHEET NUMBER	REV LTR	SHEET NUMBER	REV LTR	SHEET NUMBER	REV LTR	SHEET NUMBER	REV LTR
1-249											

DISTRIBUTION LIST

Office of Naval Research
Aeronautics, Code 461
Department of the Navy
Washington, D. C. 20360
ATTN: LCDR F. L. Cundari (2)

Commanding Officer
U. S. Naval Missile Center
Box 15
Point Mugu, California 93041
ATTN: Code 5342 (1)

Commanding Officer
U. S. Army Human Engineering Labs.
Aberdeen Proving Grounds
Aberdeen, Maryland 21005
ATTN: AMXHE-SYS (J. Barnes) (1)

U. S. Army Natick Laboratories
Anthropology Group
Pioneering Research Lab.
Natick, Massachusetts 01760
ATTN: R. M. White (5)

Defense Documentation Center
Cameron Station
Alexandria, Virginia 22314 (2)

Commander
U. S. Army Aeromedical Research Lab.
Fort Rucker, Alabama 36360
ATTN: LTCOL D. E. Littell (1)

Commander
Naval Air Systems Command
Washington, D. C. 20360
ATTN: NAVAIR 53371 (1)
      NAVAIR 5313 (2)

Commander
U. S. Naval Air Development Center
Johnsville, Warminster, Pa. 18974
ATTN: AMXI-V (1)
      ACEL   (5)

Commander
U. S. Naval Test Center
Aeromedical Branch
Service Test Division
Patuxent River, Maryland 20670 (1)

Director
U. S. Naval Research Laboratory
Washington, D. C. 20390
ATTN: Tech Info Office (1)

Commanding General
U. S. Army Electronics Command
Fort Monmouth, New Jersey 07703
ATTN: AMSEL-VL-I (3)

Commander
USAF 6570th Aero Medical Research
      Laboratory
Wright-Patterson AFB
Dayton, Ohio 45433
ATTN: AMRL (MRHA) (5)

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*	2a. REPORT SECURITY CLASSIFICATION
The Boeing Company P. O. Box 3955 Seattle, Washington 98124	Unclassified
	2b. GROUP N/A

3. REPORT TITLE

COCKPIT GEOMETRY EVALUATION, PHASE I FINAL REPORT
VOLUME III: COMPUTER PROGRAM

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

Final Report, January 1, 1968 - December 31, 1968

5. AUTHOR(S) *(First name, middle initial, last name)*

Katz, Robert
Healy, Michael J.
Meeker, Gary O.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
January 20, 1969	249	15

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
N00014-68-C-0289	D162-10127-1
b. PROJECT NO. NR 213-065	
c.	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
d.	JANAIR REPORT 690103

10. DISTRIBUTION STATEMENT

This document has been approved for public release and sale;
its distribution is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
N/A	Office of Naval Research Department of the Navy Aeronautics, Code 461 Washington D. C. 20360

13. ABSTRACT

A computer program for the evaluation of cockpit configurations using a
23 pin-joint articulated stick-man (BOEMAN-I) is presented. This program
utilized an updatable bank of anthropological and environmental data, and
simulates the motion of a real pilot performing tasks in a crewstation.
The program provides information concerning reach capability, locations
and orientations of joints, pilot-cockpit visual interferences, numerical
performance indicators on joint displacement and deflection, and mass
displacements. The program provides also a statistical validation when
comparing real pilot and BOEMAN-I paths of motion.   (U)

The distribution of this abstract is unlimited.

DD FORM 1473
1 NOV 65

D162-10127-1

14. KEY WORDS		LINK A		LINK B		LINK C	
		ROLE	WT.	ROLE	WT.	ROLE	WT.
Anthropometry	Non-Linear Optimization						
Cockpit	Motion						
Design	Performance						
Environment	Simulation						
Evaluation	System						
Geometry	Task						
Human Engineering	Visual						
Interference	Workload						
Line of Sight							

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through
_____ ."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through
_____ ."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through
_____ ."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS), (S), (C),* or *(U).*

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.