AAMRL-TR-90-015

CREW CHIEF CAD SYSTEM INTERFACE GUIDE (VERSION 2 - SI)

P.J. Krauskopf
J.W. Quinn
M.W. Jones
W.J. Stump

UNIVERSITY OF DAYTON RESEARCH INSTITUTE
DAYTON, OHIO 45469-0001

MARCH 1990

PERIOD OF PERFORMANCE: MAY 1988 TO NOVEMBER 1989

Approved for public release; distribution is unlimited

ARMSTRONG AEROSPACE MEDICAL RESEARCH LABORATORY
AIR FORCE HUMAN RESOURCES LABORATORY
HUMAN SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433

NOTICES

Please do not request copies of this report from the Armstrong Aerospace Medical Research Laboratory. Additional copies may be purchased from:

National Technical Information Service
5285 Port Royal Road
Springfield, Virginia 22161

Federal Government agencies and their contractors registered with the Defense Technical Information Center should direct requests for copies of this report to:

Defense Technical Information Center
Cameron Station
Alexandria, Virginia 22314

TECHNICAL REVIEW AND APPROVAL

AAMRL-TR-90-015

This report has been reviewed by the Office of Public Affairs (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

The voluntary informed consent of the subjects used in this research was obtained as required by Air Force Regulation 169-3.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

CHARLES BATES, JR.
Director, Human Engineering Division
Armstrong Aerospace Medical Research Laboratory

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
NOVEMBER 1989

3. REPORT TYPE AND DATES COVERED
Technical 5/88 - 11/89

4. TITLE AND SUBTITLE
CREW CHIEF CAD SYSTEM INTERFACE GUIDE
(VERSION 2 - SI)

5. FUNDING NUMBERS
F33615-84-C-0519
PE62202F
PR 7184
TA 08
WU 41

6. AUTHOR(S)
Krauskopf, P.J.; Quinn, J.W.; Jones, M.W.; and
Stump, W.J.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
University of Dayton Research Institute
300 College Park Avenue
Dayton, Ohio 45469-0001

8. PERFORMING ORGANIZATION REPORT NUMBER
UDR-TR-90-15

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Armstrong Aerospace Medical Research Laboratory
(AAMRL/HEG)
Air Force Human Resources Laboratory (AFHRL/LRA)
Wright-Patterson Air Force Base, Ohio 45669

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
AAMRL-TR-90-015

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This report describes the procedures to interface the CREW CHIEF programs with a user's CAD system through the Common Users Interface (CUI). The CUI was developed as the vehicle to pass information between the CREW CHIEF core programs and data bases and the user's CAD system. A set of interface subroutines pass input, output, and diagnostic parameters, in that order. Each passed variable is named according to FORTRAN variable type defaults, with identical parameter entities always represented by identical parameter names.

CADAM and Computervision CV4001 and CADDStation are commercial CAD systems, used during the development of CREW CHIEF and CREW CHIEF interfaces. Their use, and reference to them, does not constitute an endorsement by the United States Air Force or the University of Dayton.

14. SUBJECT TERMS

CADAM	CREW CHIEF
Computervision	Common Users Interface (CUI)
FORTRAN	

15. NUMBER OF PAGES
189

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	

SUMMARY

CREW CHIEF has been developed to allow assessment of a system's maintainability through the interaction of a computer graphics simulation of the physical characteristics and capabilities of a maintenance technician and the design elements existing in a Computer Aided Design (CAD) system. The purpose of these interactive analyses is early identification of design-induced maintainability problems. The earlier in the development phase that such problems are identified, the easier they are to correct. The CREW CHIEF system of programs is currently interfaced with CADAM (VersionS 20 and 21) and Computervision CDS 4001 and CADDStation. Since there are many commercial and proprietary CAD systems in use, direct interface of CREW CHIEF to all CAD systems would shortly become an impossible task. The CREW CHIEF program structure has been developed to allow the users to develop their own interface.

The Common User Interface (CUI) module drives the user written CAD dependent routines. Menu selection, point definition, geometry selection, floating point and alpha/numeric key-in, icon and help table selections are allowed by the module.

Information is passed between the CREW CHIEF core programs and data bases and the user's CAD system by the CUI. The CUI uses a set of interface subroutines, each of which performs a specific user function. Each subroutine contains a parameter listwhich follows established conventions. The subroutines pass input, output, and diagnostic parameters, in that order. Each passed variable is named according to FORTRAN variable type defaults. Identical parameter entities are always represented by identical parameter names.

PREFACE

This version of the CREW CHIEF system of programs was developed by University of Dayton Research Institute, 300 College Park Avenue, Dayton, Ohio 45469-0001, under United States Air Force Contract F33615-84-C-0519, entitled "Techniques for Workplace and Maintenance Evaluation." Dr. J. W. McDaniel, of the Armstrong Aerospace Medical Research Laboratory's Workload Ergonomics Branch (AAMRL/HEG), is the contract monitor. The contract was jointly funded and managed by the Armstrong Aerospace Research Laboratory and the Air Force Human Resources Laboratory. This technical report is forwarded to fulfill the requirements of CDRL Attch 1, Sequence No. 24.

The assistance of Ms. Trudy Grube and Mr. Leroy Gibbons in editing this document is greatly appreciated by the authors. The authors also acknowledge the contributions of Mr. James Rothey and Mr. Robert Berlin to this document for their co-authorship of previous corollary documents.

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

TABLE OF CONTENTS (continued)

Section Title Page

TABLE OF CONTENTS (continued)

APPENDICES

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION 1
INTRODUCTION

The CREW CHIEF system of programs provides designers a tool for early identification of design-related maintainability problems by analyzing the interaction of maintenance technicians' physical capabilities and the design elements related to specific maintenance tasks. Historically, many of such maintainability problems have been found when the system design prevents timely and cost-efficient corrective action. Such problems are often passed to the logisticians to correct, or to endure, after the system has been delivered. Since maintenance accounts for approximately 35 percent of the total cost of a system during its years of use, early identification and correction of design-induced maintenance problems can result in significant savings.

CREW CHIEF is not intended to provide solutions for all problems. For example, current military standards provide guidelines for locating components for ease of accessibility based upon such factors as frequency of maintenance actions required and the criticality of the sub-system concerned. Due to space constraints, the possible locations of sub-system components may be limited. The CREW CHIEF program will not create design, but will allow the designer to evaluate the maintainability of a candidate design. The program will also allow the user to analyze the interaction of a maintenance technician with a system design, and will enable the user to evaluate limitations and capabilities in three main areas: physical accessibility, strength, and visibility.

SECTION 2
CREW CHIEF PROGRAM STRUCTURE

A major problem encountered in developing a wholly integrated CREW CHIEF/CAD system of programs is the large number of commercial and proprietary CAD systems in use by the major manufacturers. Integrating CREW CHIEF with all combinations and options of current CAD systems, and maintaining the viability as the capabilities of both CREW CHIEF and CAD systems expand and improve, is an enormous task. Adding to the complexity of this task is the understandable reluctance of proprietors to release source data of the systems they develop in-house.

The problems inherent with supporting such a large number of CAD systems are overcome by the CREW CHIEF program structure. CREW CHIEF is composed of several program modules, each of which contains multiple layers of functional modules, as seen in Figure 2.1.

At the heart of the CREW CHIEF programs lie the Subfunction Modules. These modules perform various complex, discrete tasks, such as interference calculations or enfleshment assembly, which are required for the successful execution of many CREW CHIEF functions. The CAD system installer need not be concerned with these subfunctions, as they are already contained in the appropriate function module.

The next layer of modules contains the CREW CHIEF core functions, themselves. Each function contains its own module which must be invoked to execute that function. Each core function module is named xxxUSR, where xxx is a three-letter prefix identifying the function which that module executes (Table 2.1). Core function modules are the lowest level at which an installer may interface the CREW CHIEF programs.

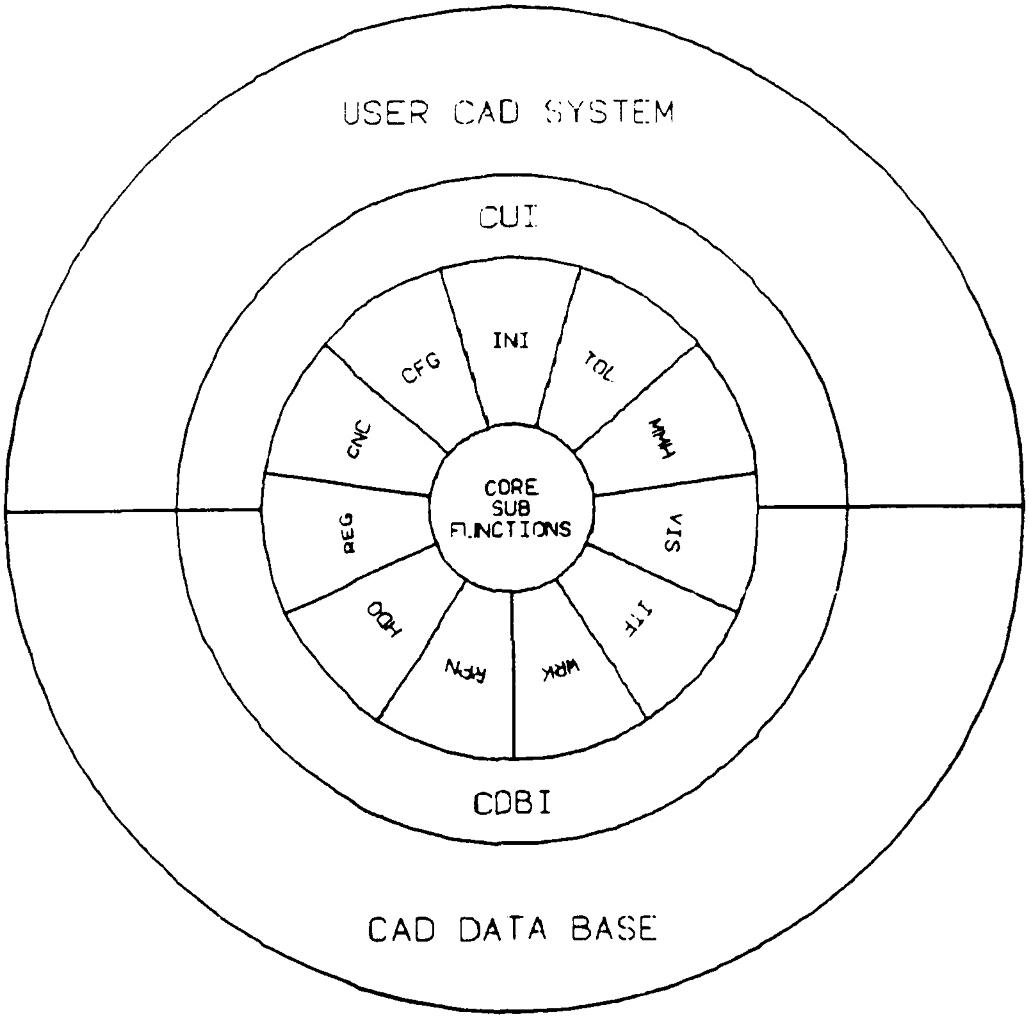


Figure 2.1. CREW CHIEF Program Modules.

To interface CREW CHIEF at this level, the installer must develop a user interface for each of the 16 CREW CHIEF functions. The installer must program the interface logic, obtain the user variables (through menu selections, prompts, etc), place these variables in the Function Control FORTRAN Common Block, and then invoke the function by calling the appropriate entry-point subroutine listed in Table 2.1. Section 4 contains instructions for developing the user interface, setting the Function Control Common, and calling the appropriate CREW CHIEF entry-point subroutine.

When interfacing CREW CHIEF at this level, the installer is responsible for all input verification, on-line HELP, and function control. Because of this, interfacing at the core function level may be an extremely complicated task. For most CAD systems, interfacing at a higher level is more desirable.

The types of user input required for the CREW CHIEF programs can be found in most CAD systems. Alpha-numeric key-in, on-screen menu selection, and on-screen geometry selection, are usually available to an applications program interfaced under a particular CAD system. Under these circumstances, the installer has another choice for the level of interface, the Common User Interface (CUI).

CUI is the outer shell of the CREW CHIEF host-independent code. This shell contains the user interface logic, and retrieves any needed information from the user by calling a small set of CAD system-dependent subroutines, each of which performs a specific user interface operation. These operations include specific types of user inputs (such as menu selection, or geometry selection), as well as CAD system bookkeeping routines. To interface CREW CHIEF to a particular CAD system, the installer need merely write routines which perform these simple operations. Section 3 describes those routines which must be modified to interface CREW CHIEF at this level.

TABLE 2.1: CREW CHIEF FUNCTION ENTRY POINTS

CREW CHIEF Functions	CREW CHIEF Entry Points
Initialization	INIUSR
Regeneration	REGUSR
Head Orientation	HDOUSR
Tool Analysis	TANUSR
Materials Handling:	
CARRY	CRYUSR
LIFT	LFTUSR
HOLD	HLDUSR
PUSH	PSHUSR
PULL	PLLUSR
REACH	RECUSR
Connector Analysis	CTRUSR
Interference Analysis	ITFUSR
Visibility Analysis	VISUSR
Configuration Analysis	CFGUSR
Work Envelope Analysis	WRKUSR
Manual Reposition Analysis	RPNUSR

To accurately analyze the maintainability of a proposed design, the CREW CHIEF system of programs must access the data base of the resident CAD system. Access may be required for data base input, data base output, or data base bookkeeping. Data base access is attained through generic subroutine calls to CAD system-dependent routines.

The CREW CHIEF system of programs outputs geometry for display of the man-model, as well as for the graphic presentation of analysis results. The installer will have to modify the 8 supplied subroutines to allow CREW CHIEF to display these elements. Section 5 contains instructions for modifying the geometry output routines.

Certain CAD utilities, such as those generating color, erasing elements, or changing line type, are accessed by CREW CHIEF through these generic subroutine calls. Section 6 contains instructions for modifying the CAD utility routines.

One of the most important aspects of CREW CHIEF is its ability to interact with the user's design. This mandates transferring information about the user's drawing from the CAD data base to the CREW CHIEF core. This is done through the CAD Data Base Input routines (CDBI), described in Section 7.

The User's Guide for CREW CHIEF: A Computer Graphics Simulation of an Aircraft Maintenance Technician (Version 2 - CD21) UDR-TR-89-103, November 1989, is the document describing the operation of the CREW CHIEF system of programs when interfaced with the CADAM computer aided design system. It has been forwarded with this guide for your information and referral for CREW CHIEF concepts of operation. Appendices A and B contain descriptions of tools and materials handling tasks, respectively.

SECTION 3
COMMON USER INTERFACE

One of the most difficult tasks in interfacing CREW CHIEF, Version I, to a new CAD system was that of programming the user interface logic. Proper interface flow, interactive user input verification and correction, and appropriate parameter setting placed a burden on the installer, and occupied the majority of time required to re-host CREW CHIEF. The CREW CHIEF Common User Interface (CUI) was developed to alleviate the burden of rehosting CREW CHIEF to a new system, by incorporating the user interface logic and input verification into a CAD independent module. All CAD system dependent processing is incorporated into a small set of subroutines, each of which performs a simple, discrete user interface function. Each CAD dependent subroutine begins with the three-letter prefix "CCX."

3.1 CUI STANDARDIZED ARGUMENT LISTS

Each CCX subroutine contains an argument list which very closely follows established conventions. These conventions cover everything from variable naming to argument order.

CCX subroutines always pass input arguments first, then output arguments, and finally, diagnostic arguments. Each passed variable is named according to FORTRAN variable type defaults, and identical argument entities are always represented by identical names.

The first variable in any argument list is the Function Identifier, **IFUNCT**. This variable is passed into the CAD-dependent subroutine, and may be used to determine the CREW CHIEF function currently being executed. Thus, a particular subroutine may key off the function being executed, to more closely tailor the interface to the application. A complete list of Function Identifiers can be found in Table 3.1.

TABLE 3.1: FUNCTION IDENTIFIERS

FUNCTION NUMBER	FUNCTION IDENTIFIER	FUNCTION NAME
1	CUIINI	CUI INITIALIZATION
2	CUIREG	MAN-MODEL RE-GENERATION
3	CUIRPN	MANUAL REPOSITION
4	CUIHDO	HEAD ORIENTATION
5	CUITOL	TOOL ANALYSIS
6	CUICRY	CARRY ANALYSIS
7	CUIHLD	HOLD ANALYSIS
8	CUILFT	LIFT ANALYSIS
9	CUIPSH	PUSH ANALYSIS
10	CUIPLL	PULL ANALYSIS
11	CUIREC	REACH ANALYSIS
12	CUICTR	CONNECTOR ANALYSIS
13	CUIVIS	VISIBILITY ANALYSIS
14	CUIITF	INTERFERENCE ANALYSIS
15	CUIWRK	WORK ENVELOPE ANALYSIS
16	CUICFG	CONFIGURATION

The second variable in each list is the Option Selector, **IOPT**. This variable is used to request the various execution modes available for a particular subroutine. The meaning of IOPT varies from subroutine to subroutine, and is described for each later in this guide.

The last variable in each argument list is the Subroutine Status Indicator, **ISTAT**. This variable is used to communicate to the programmer any special processing notes, such as HELP selection. The value of the status indicator can range from 0 to 99, and the meaning of each value remains constant across all CCX subroutines.

ISTAT= 0, Subroutine completed execution with no errors

This value for ISTAT is returned to the calling CUI subroutine when the CCX subroutine processed normally, with no errors, and with no special processing notes.

ISTAT= 1, CCX access error

This value is set when the subroutine called encounters a problem caused by the calling order of a CUI function. This status code may result from trying to add a menu before opening the CAD system (CCXOPN), or it may result from trying to close the CAD system after it was already closed. This status value is further defined for each subroutine.

ISTAT= 2, End sequenced input

Many selections are open-ended, that is, the user may have the option of making an indeterminate number of selections or inputs. For instance, the user may be given the option of selecting an indeterminate number of geometric entities to be included in the analysis. In this case, the CCX subroutine always gives the user the option of selecting "END" from the screen, to indicate that he is finished selecting. ISTAT=2 indicates to CUI that the user made this selection. Those CCX subroutines which allow sequenced input contain an argument, ISEQ, to turn on or off this type of input.

ISTAT= 3, HELP Option selected

At all times during CREW CHIEF execution, the user should have the option of selecting on-line HELP. This will usually be in the form of an on-screen selection presented to the user at all times. When the user selects the HELP option from the screen, ISTAT is set to three, thereby indicating to the calling CUI function that it should switch to the HELP mode. **Note: CUI interactive HELP is not currently available to the general CREW CHIEF installer, so the installer need not be concerned with this capability. Future releases of CUI may incorporate this feature.**

ISTAT= 4, No valid input available

When the user is prompted to select or otherwise input information from the design drawing, it may not have any valid elements to be input. For instance, the user may be prompted to select geometry from an empty drawing. When this occurs, ISTAT is set to 4.

ISTAT= 5, Return to main menu

Each time the user is prompted for input, he is allowed the option of returning to the main CREW CHIEF menu. If this option is selected, ISTAT is set to 5.

ISTAT= 6, Return to previous prompt

The user is also always presented with the option of returning to the previous prompt. When this option is selected, ISTAT is set to 6.

ISTAT= 7, Page forward (HELP only)

The on-screen HELP capabilities of CREW CHIEF allow the user to "browse" through HELP pages in either direction. The next page of HELP is called up by setting ISTAT to 7 in CCXPIC, when the HELP option is specified. This status is not available under the current release of CUI.

ISTAT= 8, Page backward (HELP only)

This is the counterpart to ISTAT= 7. This return value allows the user to see the previous HELP page. This status isnot available under the current release of CUI.

ISTAT= 10, Subroutine not installed

The set of CCX subroutines is CAD system-dependent; therefore, the installer must replace the dummy CCX subroutines included in the core, with the appropriate CAD counterpart. If the subroutine has not been replaced, the calling CUI subroutine will receive ISTAT set to 10.

ISTAT= 16, Subroutine execution error

If a subroutine finishes with any type of execution error, the status indicator is set to 16. Depending on the subroutine and the context of the call, the calling CUI subroutine may set defaults and continue on, or it may terminate the function.

ISTAT= 99, Exit Function requested

Whenever the user is prompted for input through one of the CCX routines, he should have the option of terminating the function, and returning to the CREW CHIEF main menu. This may be done through a permanent menu option, or through an icon permanently displayed on-screen. When the user selects this option, the CCX subroutine sets ISTAT to 99. This ISTAT takes precedence over all others!

* ISTAT Priority *

Most of the values for ISTAT are mutually exclusive; however, at times the interfacer may have to choose among several values to which ISTAT should be set. Exit Function (ISTAT= 99) takes precedence over all other values, followed by Subroutine execution error (ISTAT= 16). Next comes any CUI access errors (ISTAT= 1), and then the HELP-selected option. Next come the special processing notes ISTAT= 2, 4, 5, 6, 7, and 8. Finally, if no error conditions exist, and if there are no special processing notes, ISTAT should be set to 0. When ISTAT= 10, the subroutine has not been installed. The CREW CHIEF installer need not be concerned with setting this value.

3.2 INVOKING CREW CHIEF THROUGH CUI

Under CUI, the CREW CHIEF system of programs may be invoked using one of two methods, depending on the needs and capabilities of the host CAD system. Each type has its own link-edit and access methods.

Those CAD systems for which main memory is at a premium may access each CREW CHIEF function individually, thereby saving the overhead memory of loading all CREW CHIEF code at once. This type of access requires, however, that the host CAD system be able to load specific load modules into memory during execution. Under this method, the CREW CHIEF installer is responsible for developing a user interface for the CREW CHIEF main menu system.

On the other hand, some CAD systems do not have the ability to load run-time modules, and under these systems CREW CHIEF will have to be link-edited with the CAD load module, itself. In this case, the CREW CHIEF system of programs is invoked through a single subroutine call, after which the entire user interface is under the control of CREW CHIEF. No user interface need be developed by the installer using this method of interface.

3.2.1 Invoking Individual CREW CHIEF Functions

When invoking a single CREW CHIEF function, the CREW CHIEF installer must first develop a set of 5 menus to allow the user to select the desired function. The installer can use the user input from these menus to determine the function selected. Figure 3.1 shows the user interface flow for these five top-level menus.

The first menu the installer must create is the CREW CHIEF Main Menu. This is the menu first presented when the user enters the CREW CHIEF system of programs, and displays a list of available functions and function classes (Figure 3.2). The item selected from this menu may determine the function to be executed (such as Visibility or Current Configuration), or it may determine the next menu to be displayed to the user.

The Generation Menu (Figure 3.3) is displayed when the user selects "Generation Functions" from the CREW CHIEF Main Menu. The user selection from this menu will determine the CREW CHIEF function to be executed.

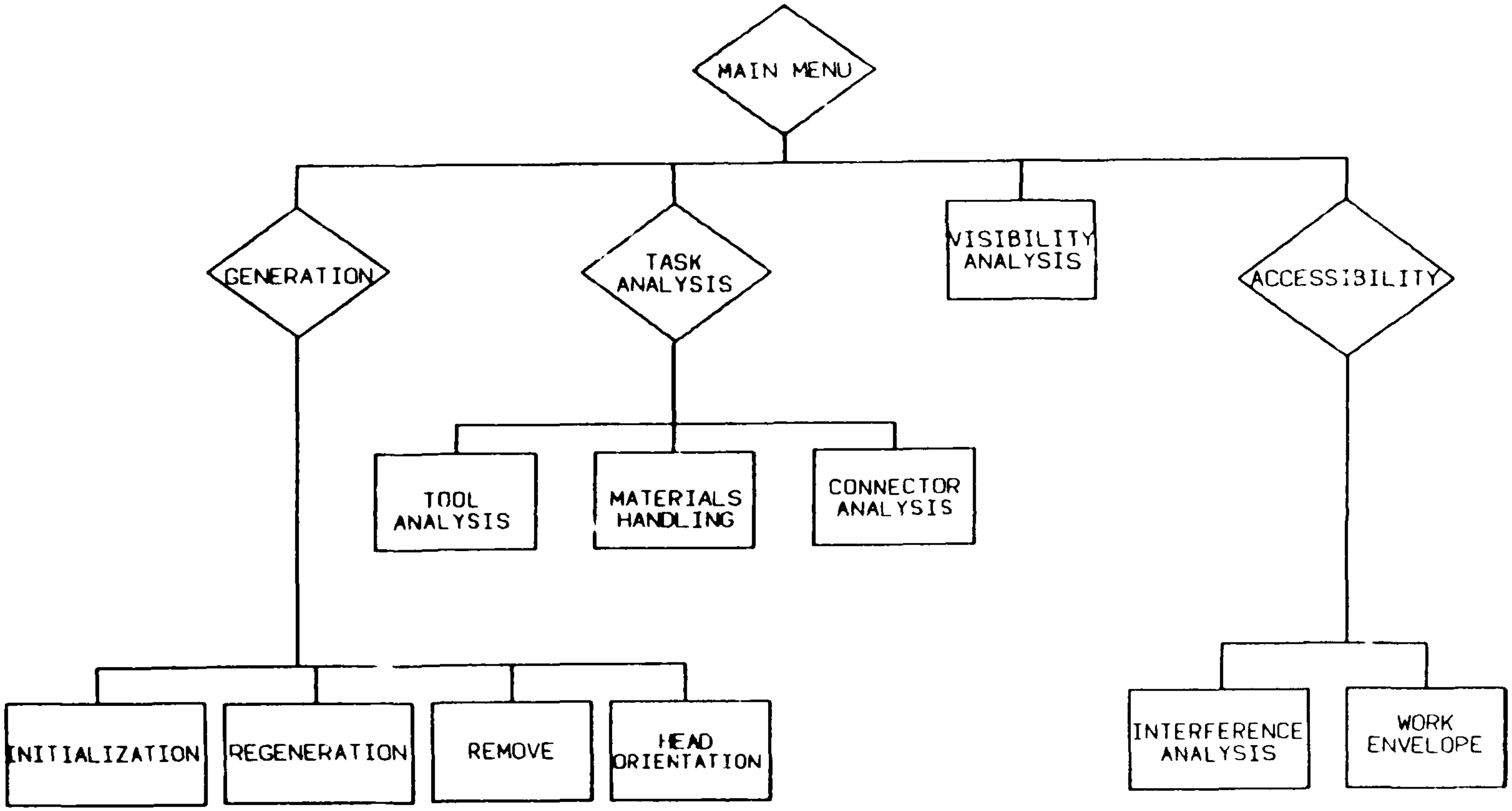
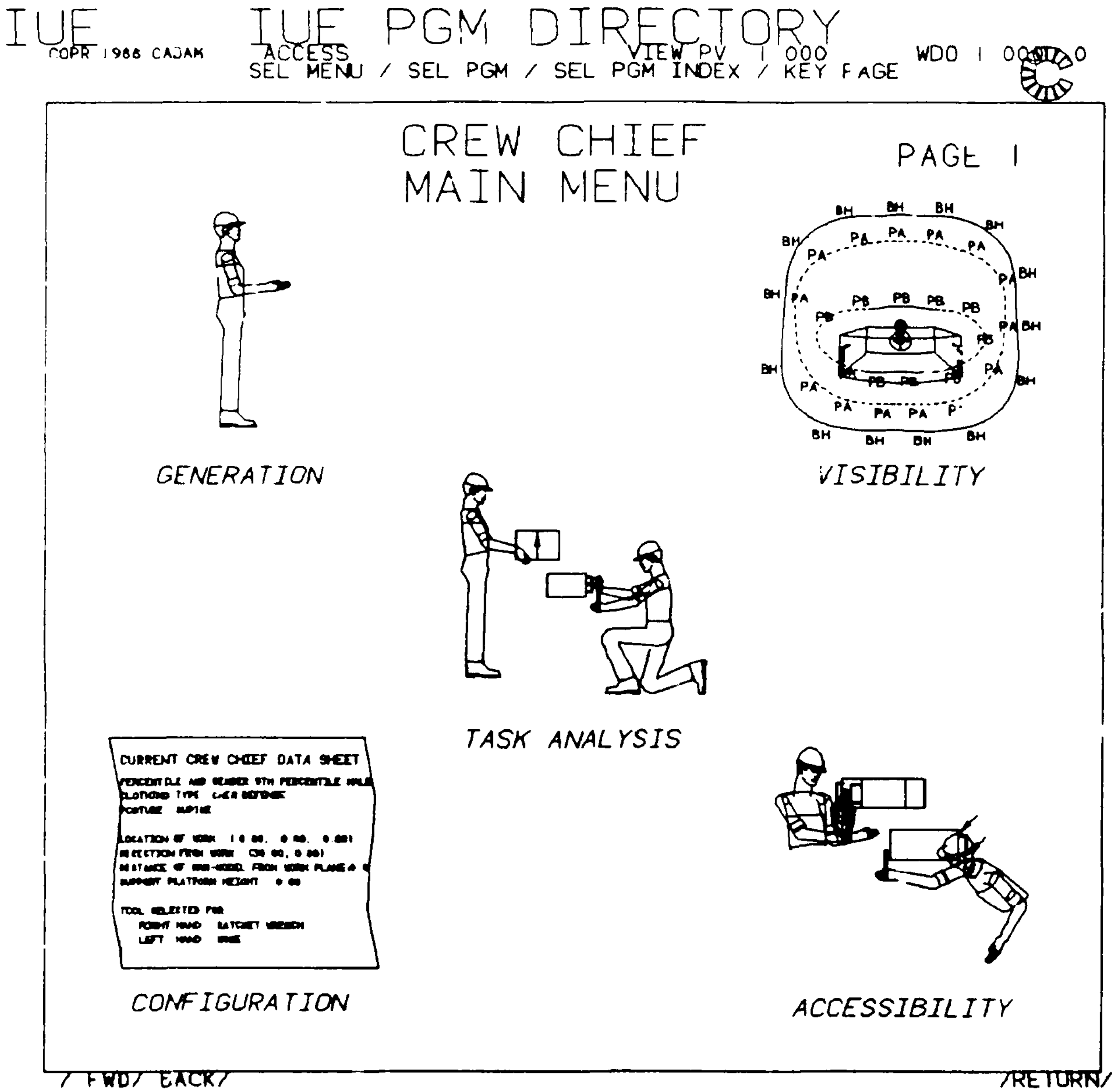


Figure 3.1. User Interface Flow



IUE IUE PGM DIRECTORY
COPR 1988 CAJAK ACCESS VIEW PV 1 000 WDO 1 000 0
SEL MENU / SEL PGM / SEL PGM INDEX / KEY PAGE

CREW CHIEF
MAIN MENU

PAGE 1

CURRENT CREW CHIEF DATA SHEET
PERCENTILE AND GENDER 5TH PERCENTILE MALE
CLOTHING TYPE CHEM DEFENSE
POSTURE SUPINE

LOCATION OF WORK (0.00, 0.00, 0.00)
DIRECTION FROM WORK (30.00, 0.00)
DISTANCE OF MAN-MODEL FROM WORK PLANE [illegible]
SUPPORT PLATFORM HEIGHT 0.00

TOOL SELECTED FOR
RIGHT HAND RATCHET WRENCH
LEFT HAND NONE

/ FWD/ BACK/ /RETURN/

Figure 3.2. CREW CHIEF Main Menu

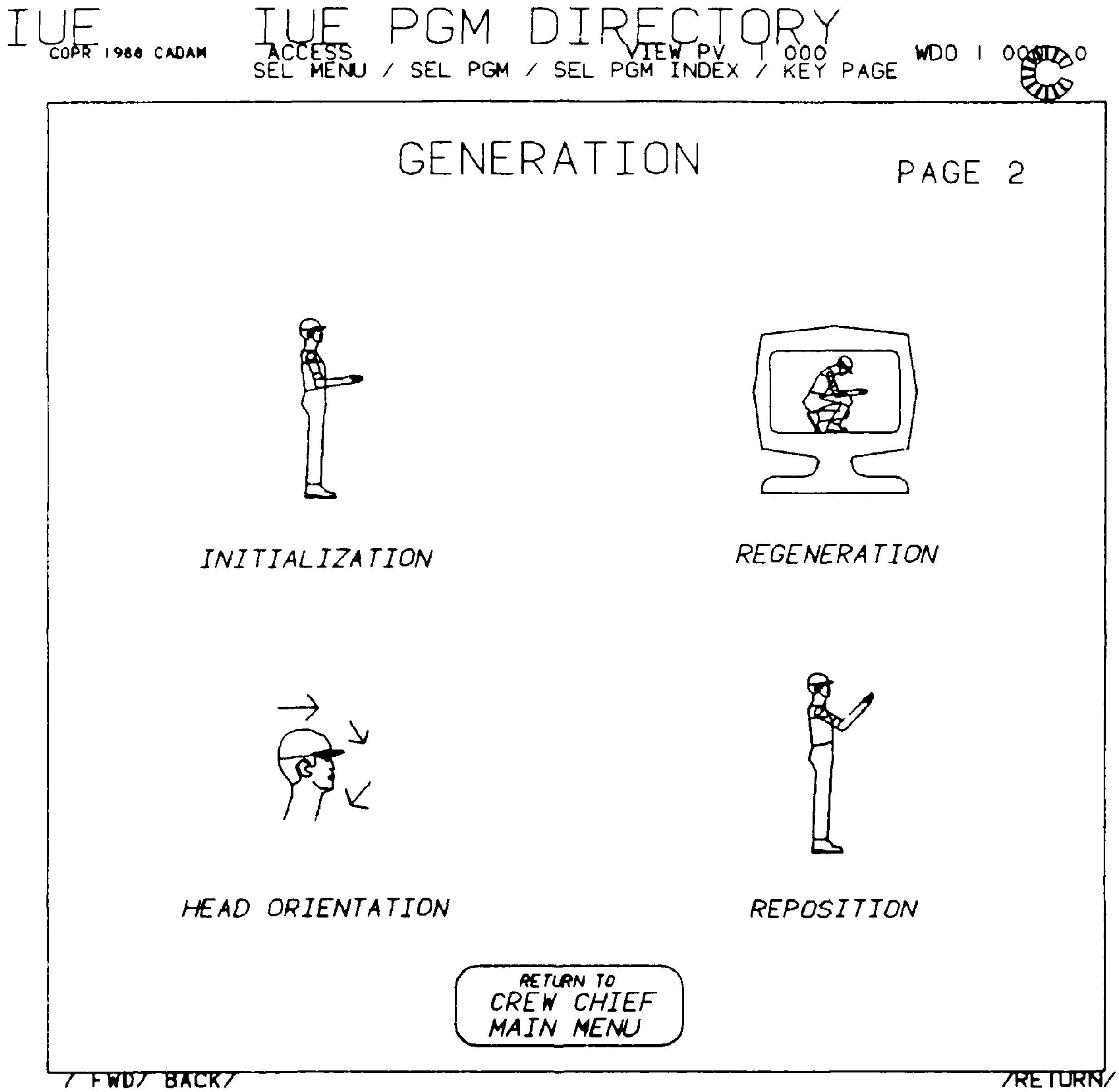


Figure 3.3. Generation Menu

The Task Analysis Menu (Figure 3.4) is displayed when the user selects "Task Analysis Functions" from the CREW CHIEF Main Menu. The user selection from the menu may determine the function to be executed (such as Tool Analysis or Connector Analysis) or, if the user selects "Manual Materials Handling Functions," the next menu displayed, the Manual Materials Handling Menu.

The Manual Materials Handling Menu (Figure 3.5) displays the list of object-handling analyses available for selection. The user selection from this menu will determine the function to be executed.

Finally, the Accessibility Analysis Menu (Figure 3.6) is displayed when the user selects "Accessibility Analyses" from the CREW CHIEF Main Menu. The user selection from this menu will also determine the function to be executed.

Once the function has been selected, the installer must invoke the CREW CHIEF system of programs through the appropriate subroutine call, as determined by Table 2.1. Each CREW CHIEF entry-point subroutine follows the nomenclature "CUIxxx," where "xxx" is the three-letter function identifier shown in the table.

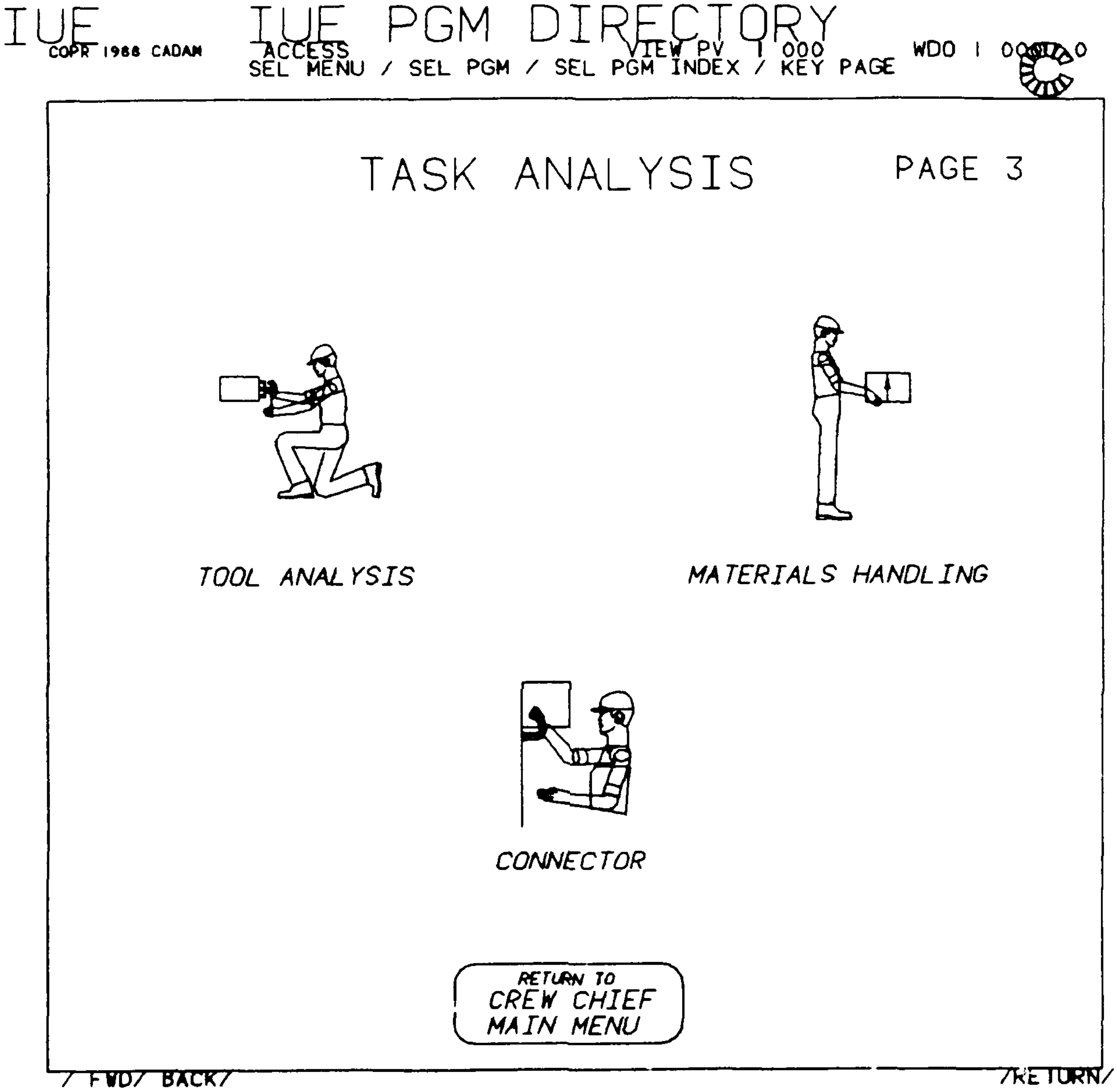


Figure 3.4. Task Analysis Menu

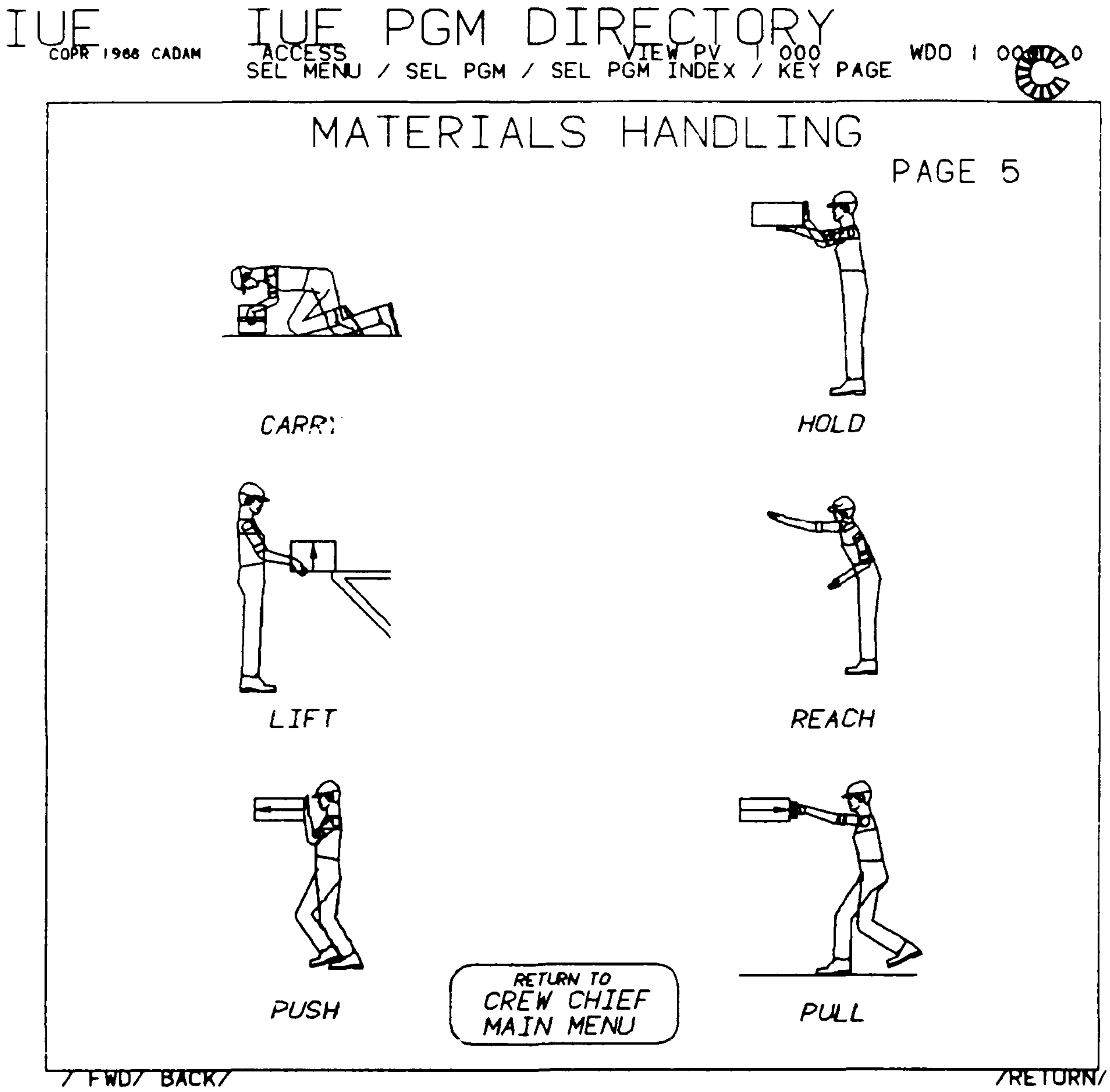


Figure 3.5. Manual Materials Handling Menu

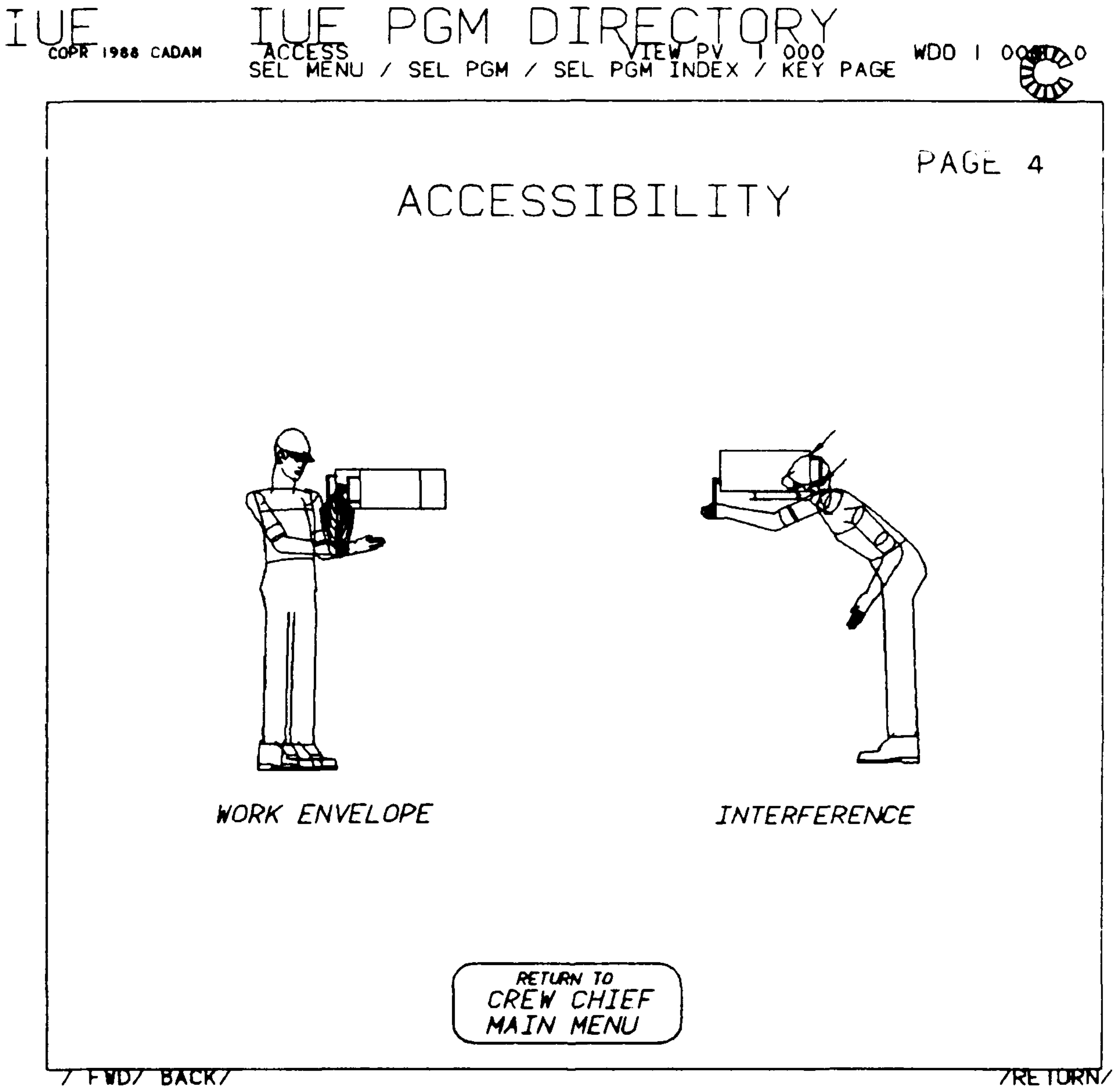


Figure 3.6. Accessibility Analysis Menu

The argument lists of all entry-point subroutines are the same:

Input

IFUNCT - Function requested.
IBUGLV - Level of debug requested.
= 0, perform no debug processing.
= 1, Perform function-level debug.
= 2, Perform sub-function debug.
= 3, perform CUI-level debug.
ICODSP - Icon display flag
= 1, Display only menu text; no icons or on-line HELP available (this is the only one currently available).
= 2, Icons, on-line HELP displayed (not currently available).
UNITS - Current drawing units-per-inch (2.54= cm.)
VUMTRX - 3 x 3 screen view matrix.

Subroutine Status Indicator

ISTAT= 0, CUI completed execution with no errors.
= 1, CCX access error, CAD system not opened.
= 5, Return to main menu requested.
= 10, one or more subroutines not installed.
= 16, CUI execution error-- invalid parameters
= 99, Exit function requested.

3.2.2 Linking CREW CHIEF to a CAD System

If the host CAD system does not have the capability to load and execute external modules, and if the host computer has enough real or virtual memory to load the CAD system and the CREW CHIEF system of programs, the installer may link CREW CHIEF directly to the host CAD system. The CREW CHIEF programs require approximately 3000K of memory in addition to the requirements of the host CAD system.

Under this type of interface, the CREW CHIEF system of programs is entered vιa a single subroutine call to the entry-point subroutine CUIFAC. This level of interface performs all user interface logic, including the top-level menuing described earlier. The CREW CHIEF installer is responsible for developing only the simple interface routines described in paragraph 3.3.

The CUI program operation is controlled via the arguments passed to CUIFAC:

TITLE: Enter Common User Interface

CUIFAC(IFUNCT,IOPT,IOPT2,VUMTRX,UNITS,IBUGLV,ISTAT)

Input:

```
IFUNCT - Function requested (IOPT= 2, only)
IOPT   - Execution level option
         = 1, Execute all functions
         = 2, Execute only the function specified
              by IFUNCT
IOPT2  - Display level (must be set to 1 for this
                                  release)
         = 1, Display only menu text; no icon or
              on-line HELP available.
         = 2, Icons, on-line HELP displayed.
VUMTRX - 3 x 3 screen view matrix
UNITS  - Current drawing units-per-inch (2.54 = cm.)
IBUGLV - Level of debug requested
         = 0, Perform no debug processing
         = 1, Process function-level debugging
         = 2, Process sub-function level debugging
                 = 3, Perform CUI level debugging
```

Subroutine Status indicator

ISTAT= 0, CUI completed execution with no errors.
= 1, CCX access error, CAD system not opened.
= 10, One or more subroutines not installed
= 16, CUI execution error--invalid parameters.

3.3 CCX SUBROUTINE DESCRIPTIONS

TITLE: Alpha-Numeric Key-In Processor

CCXAKY(IFUNCT,IOPT,ISEQ,PROMPT,NPROMP,VALUE,NOWORD,ISTAT)

Subroutine CCXAKY is used when CUI needs alpha-numeric input from the user. It can be used in single mode, as well as in sequenced mode.

Input

IFUNCT - CREW CHIEF Function Identifier

IOPT - Not currently used

ISEQ - Sequenced input indicator
= 0, single selection only
= 1, Sequenced input on

PROMPT(15) - Text of prompt to be displayed to the user while waiting for user input

NPROMP - Number of words of text in prompt message.

Output

NUMVAL - Numbers of values keyed-in. Note that if NUMVAL is passed into CCXAKY and is greater than zero, then the NUMVALth entry in VALUE is used as the default key-in.

VALUE(30,30) - Text keyed in by user

NOWORD(30) - Number of 4-byte words used to store text.

Subroutine Status Indicator

ISTAT= 0, Subroutine completed execution with no errors.
= 1, CCX access error, CAD system not opened.
= 2, End sequenced input--Last key-in
= 3, HELP Option selected
= 5, Return to main menu
= 6, Return to previous prompt
= 10, Subroutine not installed
= 16, Subroutine execution error
= 99, Exit Function requested

TITLE: Close CUI calls

CCXCLO(IFUNCT,IOPT,ISTAT)

Subroutine CCXCLO is used to perform any close-out processing required before returning to the resident CAD system.

Input

IFUNCT - CREW CHIEF Function Identifier
IOPT - Not currently used.

Subroutine Status Indicator

ISTAT= 0, Subroutine completed execution with no errors.
= 1, CCX access error, CAD system not opened.
= 10, Subroutine not installed
= 16, Subroutine execution error

TITLE: Display Geometry

CCXDSP(IFUNCT,IOPT,ISTAT)

Subroutine CCXDSP is used to display geometry on the graphics terminal. All geometry, including the work place, is displayed.

Input

IFUNCT - CREW CHIEF Function Identifier
IOPT - Not currently used

Subroutine Status Indicator

ISTAT= 0, Subroutine completed execution with no errors.
= 1, CCX access error, CAD system not opened.
= 16, Subroutine execution error

TITLE: Floating-point Key-in processor

CCXFKY(IFUNCT,IOPT,ISEQ,PROMPT,NPROMP,VALUE,ISTAT)

Subroutine CCXFKY is used when CUI needs real number input from the user. It can be used in single and sequenced mode.

Input

IFUNCT - CREW CHIEF Function Identifier
IOPT - Not currently used
ISEQ - Sequenced input indicator
= 0, single selection only
= 1, Sequenced input on
PROMPT(15) - Text of prompt to be displayed to the user while waiting for user input
NPROMP - Number of words of text in prompt message.

Output

NUMVAL - Number of values keyed-in. Note that if NUMVAL is passed into CCXFKY and is greater than zero, then the NUMVALth entry in VALUE is used as the default key-in.

VALUE - Array containing keyed-in/default values.

Subroutine Status Indicator

ISTAT= 0, Subroutine completed execution with no errors.
= 1, CCX access error, CAD system not opened.
= 2, End sequenced input--Last key-in
= 3, HELP Option selected

= 5, Return to main menu
= 6, Return to previous prompt
= 10, Subroutine not installed
= 16, Subroutine execution error
= 99, Exit Function requested

TITLE: Geometry Selector

CCXGEO(IFUNCT,IOPT,ISEQ,PROMPT,NPROMP,ITYPE,NUMTYP, INTITY,INTTYP,ISTAT)

Subroutine CCXGEO prompts the CAD user to select a single piece of geometry from his design. The types of geometry selected can be specified, and the programmer has the option of allowing multiple types to be selected.

Input

IFUNCT - CREW CHIEF Function Identifier

IOPT - Geometry selection option
= 1, select workplace geometry
= 2, select man-model body segment
= 3, select man-model rotation axis

ISEQ - Sequenced input indicator
= 0, single selection only
= 1, Sequenced input on

PROMPT(15) - Text of prompt to be displayed to the user while waiting for user input

NPROMP - Number of words of text in prompt message

ITYPE(20) - Selectable geometry types
- If IOPT = 1, this array contains selectable workplace geometry, (see Table 3.2).

TABLE 3.2: SELECTABLE GEOMETRY

ITYPE(20) - Types of geometry allowed selection

= 0, All 3-D geometry

= 1, Point

= 2, Line segment

= 3, Arc

= 4, Parabola

= 5, Cubic spline

= 6, B-spline

= 7, Ruled surface

= 8, Surface of revolution

= 9, B-surface

= 10, Bicubic surface

= 11, Planar polygonal (mesh) surface

= 12, Bezier patches

- If IOPT = 2, this array contains the set of selectable body segments, (see Table 3.3).
- If IOPT = 3, this array contains the set of body segments whose axes are selectable, (see Table 3.3).

NUMTYP - The total number of selectable geometry types.

Output

INTITY - Identifier for the selected individual entity

NBRSEL - Number of elements selected. Note: If CCXGLO is called with NBRSEL > 0, then the first through NBRSELth elements in INTITY will be defaulted.

INTTYP - Entity type identifier array.

Subroutine Status Indicator

ISTAT= 0, Subroutine completed execution with no errors.
= 1, CCX access error, CAD system not opened.
= 2, End sequenced input--end of geometry selection
= 3, HELP Option selected
= 4, No valid input available--No selectable entities in model
= 5, Return to main menu
= 6, Return to previous prompt
= 10, Subroutine not installed
= 16, Subroutine execution error
= 99, Exit Function requested

TABLE 3.3: SELECTABLE BODY SEGMENTS

Segment #	Description
1	Hips
2	Trunk
3	Right Upper Arm
4	Right Lower Arm
5	Right Hand
6	Left Upper Arm
7	Left Lower Arm
8	Left Hand
9	Right Upper Leg
10	Right Lower Leg
11	Left Upper Leg
12	Left Lower Leg
13	Head
14	Right Boot
15	Left Boot
16	Right Main Tool
17	Right Extension
18	Left Main Tool
19	Left Extension
20	Right Socket
21	Left Socket
22	Misc. Geometry
23	Entire Man-model

TITLE: User Menu Processor

**CCXMNU(IFUNCT,IOPT,ISEQ,PROMPT,NPROMP,
MENU,NBRSEL,MENWRD,XY,ITEM,ISTAT)**

Subroutine CCXMNU is called when the user needs to select from a list of possible choices. It can be combined with CCXPIC to produce icon selections. It can also be used in sequenced mode to allow the user the ability to select a subset of the available choices (such as which visibility contours to display).

Input

IFUNCT - CREW CHIEF Function Identifier

IOPT - Placement option
= 0, Do not place. Selections defined as purely icon.
= 1, Use default placement procedures
= 2, Place according to the values in Xy

ISEQ - Sequenced input indicator
= 0, single selection only
= 1, Sequenced input on

PROMPT(15) - Text of prompt to be displayed to the user while waiting for user input

NPROMP - Number of words of text in prompt message

MENU(15,30) - List of menu items (max= 30). The description of menu item #I can be found in MENU(1-MENWRD(I),I).

NBRSEL - Number of items in menu

MENWRD(30) - Number of words needed to store description of each menu item in MENU

XY(2,30) - The leftmost, center of the first character in MENU(1,I) begins at XY(1-2,I) (for IOPT= 2, only)

Output

ITEM(30) - Array containing items selected

NBRITM - Number of items selected. Note that if NBRITM is passed into CCXMNU and is greater than zero, than ITEM(1) - ITEM(NBRITM) contains those items to be pro-selected.

Subroutine Status Indicator

ISTAT= 0, Subroutine completed execution with no errors
= 1, CCX access error, CAD system not opened.
= 2, End sequenced input--Last item to be selected from this menu
= 3, HELP Option selected
= 5, Return to main menu
= 6, Return to previous prompt
= 10, Subroutine not installed
= 16, Subroutine execution error
= 99, Exit Function requested

TITLE: Open CUI Calls

CCXOPN(IFUNCT,IOPT,IDEBUG,ISTAT)

Subroutine CCXOPN is called to initialize the CAD system in preparation for the user interface. It is used to prepare any pointers, text, or anything else that may need to be prepared prior to the execution of a function.

<u>Input</u>

IFUNCT - CREW CHIEF Function Identifier
IOPT - Not currently used
IDEBUG - CUI debug indicator

<u>Subroutine Status Indicator</u>

```
ISTAT=  0, Subroutine completed execution with no
           errors.
     =  1, CCX access error, CAD system not opened.
     = 10, Subroutine not installed
     = 16, Subroutine execution error
```

TITLE: Picture Processor

CCXPIC(IFUNCT,IOPT,IASSOC,NUMPIC,MAXPIC,XYPAT)

Subroutine CCXPIC is used to overlay pictures on the user's screen. It can be used for displaying HELP pages, as well as defining icon selections for use with menus. **(Note: not currently available.)**

Input

IFUNCT - CREW CHIEF Function Identifier.

IOPT - Display option
- = 1, Clear all previous pictures from screen
- = 2, Add requested picture to screen

IASSOC - Menu item to which to associate this picture
- = 0, This picture is not selectable
- = N, Selecting this picture corresponds to selecting menu item # N from the any preceding menus. Note: All associative pictures are cleared each time CCXPIC is called with IOPT= 1.

NUMPIC - Identifier for picture or icon to be displayed
- = 0, No picture desired (used with IOPT= 1 to clear all pictures and associations)
- = N, Display the Nth icon for this function.
- = -N, Display the Nth HELP page for function.

MAXPIC - Identifier of last available help page

XYPAT(2) - Screen coordinates of picture attach point

Subroutine Status Indicator

ISTAT= 0, Subroutine completed execution with no errors.

= 1, CCX access error, CAD system not opened.

= 4, No valid input available--Picture not available.
= 7, Page forward (for NUMPIC < 0, only).
= 8, Page backward (for NUMPIC < 0, only).
= 10, Subroutine not installed.
= 16, Subroutine execution error.
= 99, Exit HELP function requested.

TITLE: Point Processor

CCXPNT(IFUNCT,IOPT,ISEQ,PROMPT,NPROMP,NUMPNT,POINT,ISTAT)

Subroutine CCXPNT is used to allow the user to define a point for input into the program. This subroutine is very similar to CCXGEO, with ITYPE= 1. However, the main difference here is that, in addition to selecting an existing point, the user may be given the opportunity to define a point through key-in, or some other method. Precisely how the point is defined will be determined by the CAD system, as well as how this particular subroutine is written.

Input

IFUNCT - CREW CHIEF Function Identifier
IOPT - Type of point to be input
= 2, 2-D point input (Screen Coordinates)
= 3, 3-D point input (Global coordinates)
ISEQ - Sequenced input indicator
= 0, Single selection only
= 1, Sequenced input on
PROMPT(15) - Text of prompt to be displayed to the user while waiting for user input
NPROMP - Number of words of text in prompt message

Output

NUMPNT - If sequenced, number of points already selected. Note that if NUMPNT is passed into CCXPNT and is greater than zero, then the first NUMPNTth values of POINT contain default data.

POINT(IOPT,30) - Coordinates of selected point

Subroutine Status Indicator

ISTAT= 0, Subroutine completed execution with no errors
= 1, CCX access error, CAD system not opened.
= 2, End sequenced input--End point definition sequence
= 3, HELP Option selected
= 4, No valid input available--points not available
= 5, Return to main menu
= 6, Return to previous prompt
= 10, Subroutine not installed
= 16, Subroutine execution error
= 99, Exit Function requested

SECTION 4
INTERFACING TO THE CREW CHIEF CORE

If the proposed CREW CHIEF host does not allow interfacing using the CUI modules, then the installer must access the CREW CHIEF functions by direct invocation of the appropriate core modules. Each module is FORTRAN-callable, and function control parameters are passed to the module through a specific FORTRAN COMMON block. In this type of interface, the installer is directly responsible for interactive input verification, as well as user interface logic and design, including the top-level menuing described in paragraph 3.3.1. Interfacing to CREW CHIEF at this level requires significantly more work than is required when interfacing through CUI.

This section contains instructions for interfacing each function of the CREW CHIEF core to the host system. The general method is to prompt the user for the specified information, use this information to set the appropriate COMMON block variables, and then invoke the particular function by calling the appropriate entry-point subroutine. Note that the installer will still need to develop the CAD output and input routines.

4.1 GENERATION FUNCTIONS

The three CREW CHIEF Generation functions are Initialization, Regeneration, and Head Orientation. These functions are used to define the body size, clothing type, initial posture, position and orientation, display type, and head orientation of the man-model in the user's drawing. Descriptions of man-model generation, body size, clothing type, and initial posture selections can be found in the CREW CHIEF User's Guide.

4.1.1 CREW CHIEF Initialization Function

The CREW CHIEF Initialization function generates the elements of the man-model and displays them in the user's drawing. The following input parameters must be defined and passed to the CREW CHIEF Initialization function through the common block INICTL.

NUMMOD - Defines body size as a function of population percentile and gender (INTEGER).

NUMCLO - Defines clothing type (INTEGER).

NUMPOS - Defines the initial posture of the man-model (INTEGER).

NEWPOS - A flag for placing the man-model in the same position and orientation as in the last execution of the Initialization function (INTEGER).

WORKLC(3) - Defines the Location of Work (REAL).

WORKDR(2) - Defines the Work Direction (REAL).

DISWRK - Defines the Distance from Work (REAL).

PLATHT - Defines Platform Height (REAL).

NUMDSP - Defines the display type (INTEGER).

UNITS - Defines the drawing units per inch (REAL).

VUMAT(3,3) - A 3x3 array containing the view matrix (REAL).

IBGINI - Debug flag for the Initialization function (INTEGER).

The input parameters are grouped in three general areas: Body Size, Clothing Type, and Initial Posture Definition; Man-model Position and Orientation Definition; and, Screen Display Definition and Debugging Flag Status.

4.1.1.1 Body Size, Clothing Type, and Initial Posture Definition

NUMMOD - Defines body size as a function of population percentile and gender. Set NUMMOD to the index for the percentile and gender desired.

1 - 1st percentile male
2 - 5th percentile male
3 - 50th percentile male
4 - 95th percentile male
5 - 99th percentile male
6 - 1st percentile female
7 - 5th percentile female
8 - 50th percentile female
9 - 95th percentile female
10 - 99th percentile female

NUMCLO - Defines the clothing type. Set NUMCLO to the index corresponding to the clothing type to be used.

1 - FATIGUES
2 - FATIGUES WITH JACKET
3 - ARCTIC
4 - CHEMICAL DEFENSE

NUMPOS - Defines the initial posture of the man-model. Set NUMPOS to the index for the required initial posture.

1 - STAND
2 - SIT
3 - BEND
4 - SUPINE
5 - PRONE
6 - SIDE
7 - KNEEL 1 (ON ONE KNEE)
8 - KNEEL 2 (ON TWO KNEES)
9 - SQUAT
10 - WALK
11 - CRAWL
12 - CLIMB

4.1.1.2 Man-model Position and Orientation Definition

The next five parameters are used to define the position and orientation of the man-model in the drawing. The CREW CHIEF programs make the following assumptions:

- the man-model faces the Location of Work, and
- vertical parallels the drawing Z axis, which is positive in the upward direction.

NEWPOS - A flag for placing the man-model in a new position and/or orientation, or in the same position and orientation used during the last execution of the Initialization function during the program run. Set NEWPOS to the index corresponding to the placement desired.

0 - New position and orientation
1 - Same position and orientation

NOTE: NEWPOS = 0 is required for the first execution of the Initialization function during a program run. Parameters to be defined for the remainder of the Position and Orientation definition are dependent on the NEWPOS index selection.

- **When NEWPOS = 0**, the following parameters must be defined.

WORKLC - Defines the 3-D point, e.g., a bolt, connector, etc., around which the man-model is to work. Set the array WORKLC, in drawing coordinates, as follows:

WORKLC (1)	- X coordinate of point
WORKLC (2)	- Y coordinate of point
WORKLC (3)	- Z coordinate of point

WORKDR - Defines the X and Y coordinates of a 3-D point from which the man-model faces the Location of Work. Set the array WORKDR, in drawing coordinates, as follows:

WORKDR (1)	- X coordinate of point
WORKDR (2)	- Y coordinate of point

DISWRK - Defines the distance from work, which is the horizontal distance between the Location of Work and the man-model. Set DISWRK to the desired horizontal distance.

PLATHT - Defines the Platform Height as the elevation of the horizontal support plane (negative or positive Z values from the drawing origin) on which the man-model is located, e.g., the ground, a maintenance platform, etc. Set PLATHT to the elevation required.

- **When NEWPOS = 1**, the program uses the values established for WORKLC, WORKDR, DISWRK, and PLATHT during the last execution of the Initialization function.

4.1.1.3 Screen Display Definition and Debugging Flag Status

NUMDSP - Defines the display type used when drawing the man-model. Set NUMDSP to the index corresponding to the desired display type.

1 - WIRE FRAME
2 - SURFACED
3 - PROFILE

UNITS - Set UNITS to the number of units per inch. For example, if the drawing unit is in centimeters, set UNITS to 2.54; for millimeters, set UNITS to 25.4.

VUMAT(3,3) - A 3x3 array containing the view matrix which describes the orientation of the user's drawing with respect to the display screen. Available in most CAD systems, this array allows presentation of an uncluttered profile image of the man-model (If the view matrix is not available in your system, set Display Type [NUMDSP=1] to wire frame and ignore the setting VUMAT.) CREW CHIEF assumes that the screen system positive axis points in the following directions: X points to the viewer's right, Y points up, and Z points toward the viewer. Set the array VIEWMAT as follows:

VUMAT(1-3,1) - The screen coordinates of the direction vector defining the positive X axis of the drawing coordinate system.

VUMAT(1-3,2) - The screen coordinates of the direction vector defining the positive Y axis of the drawing coordinate system.

VUMAT(1-3,3) - The screen coordinates of the direction vector defining the positive Z axis of the drawing coordinate system.

IBGINI - A flag that defines whether or not control variables are to be printed. Set IBGINI to the index desired.

0 - Do not print control variables
1 - Print control variables

Once the required input parameters have been defined, the CREW CHIEF subroutine INIUSR is called to generate and display the man-model in the user's drawing.

4.1.2 CREW CHIEF Regeneration Function

The initial posture of the man-model is defined with the Initialization function. During Task Analysis functions, the initial posture may be modified. The modified posture data is retained by the program until a change is made, either with another execution of the Initialization function, or during another Task Analysis function. The Regeneration function is provided to allow the user to recall the man-model to the screen in the configuration currently retained by the program. When the Regeneration function is called, the following screen display input parameters must be defined and passed through the common block REGCTL.

NUMDSP - Defines display type (INTEGER).

UNITS - Defines drawing units per inch (REAL).

VUMAT(3,3) - A 3x3 array containing the view matrix (REAL).

IBGREG - Debug flag for the Regeneration function (INTEGER).

4.1.2.1 Screen Display Definition and Debugging Flag Status

NUMDSP - Defines the display type used when drawing the man-model. Set NUMDSP to the index corresponding to the desired display type.

1 - WIRE FRAME
2 - SURFACED
3 - PROFILE

UNITS - Set UNITS to the number of units per inch. For example, if the drawing unit is in centimeters, set UNITS to 2.54; for millimeters, set UNITS to 25.4.

VUMAT(3,3) - A 3x3 array containing the view matrix which describes the orientation of the user's drawing with respect to the display screen. Available in most CAD systems, this array allows presentation of an uncluttered profile image of the man-model. (If the view matrix is not available in your system, set Display Type [NUMDSP=1] to wire frame and ignore the setting VUMAT.) CREW CHIEF assumes that the screen system's positive axis points in the following directions: X points to the viewer's right, Y points up, and Z points toward the viewer. Set the array VUMAT as follows:

VUMAT(1-3,1) - The screen coordinates of the direction vector defining the positive X axis of the drawing coordinate system.

VUMAT(1-3,2) - The screen coordinates of the direction vector defining the positive Y axis of the drawing coordinate system.

VUMAT(1-3,3) - The screen coordinates of the direction vector defining the positive Z axis of the drawing coordinate system.

IBGREG - A flag that defines whether or not control variables are to be printed. Set IBGREG to the index desired.

0 - Do not print control variables
1 - Print control variables

Once the input parameters have been defined and passed, the CREW CHIEF subroutine REGUSR is called to display the man-model in the drawing.

4.1.3 CREW CHIEF Head Orientation Function

During task analysis functions, the man-model automatic-ally looks at the Location of Work. The CREW CHIEF Head Orienta-tion function is used to allow the man-model look at a different point. The following parameters must be defined and passed through the common block HDOCTL:

TARGET(3) - Defines the 3-D point toward which the man-model is to look (REAL).

NUMDSP - Defines the display type (INTEGER).

UNITS - Defines the drawing units per inch (REAL).

VUMAT(3,3) - A 3x3 array containing the view matrix (REAL).

IBGHDO - Debug flag for the Head Orientation function (INTEGER).

4.1.3.1 Target Point Definition

TARGET - Defines the 3-D point toward which the man-model is to look. Set the array TARGET, in drawing coordinates, as follows:

TARGET (1) - X coordinate of point
TARGET (2) - Y coordinate of point
TARGET (3) - Z coordinate of point

4.1.3.2 Screen Display Definition and Debugging Flag Status

NUMDSP - Defines the display type used when drawing the man-model. Set NUMDSP to the index corresponding to the desired display type.

1 - WIRE FRAME
2 - SURFACED
3 - PROFILE

UNITS - Set UNITS to the number of units per inch. For example, if the drawing unit is in centimeters, set UNITS to 2.54; for millimeters, set UNITS to 25.4.

VUMAT(3,3) - A 3x3 array containing the view matrix which describes the orientation of the user's drawing with respect to the display screen. Available in most CAD systems, this array allows presentation of an uncluttered profile image of the man-model. (If the view matrix is not available in your system, set Display Type [NUMDSP=1] to wire frame and ignore the setting VUMAT.) CREW CHIEF assumes that the screen system positive axis points in the following directions: X points to the viewer's right, Y points up, and Z points toward the viewer. Set the array VUMAT as follows:

VUMAT(1-3,1) - The screen coordinates of the direction vector defining the positive X axis of the drawing coordinate system.

VUMAT(1-3,2) - The screen coordinates of the direction vector defining the positive Y axis of the drawing coordinate system.

VUMAT(1-3,3) - The screen coordinates of the direction vector defining the positive Z axis of the drawing coordinate system.

IBGHDO - A flag that defines whether or not control variables are to be printed. Set IBGHDO to the index desired.

0 - Do not print control variables
1 - Print control variables

Once the required parameters are defined and passed, the CREW CHIEF subroutine HDOUSR is called to position the man-model's head to look at the specified point.

4.2 TASK ANALYSIS FUNCTIONS

The Task Analysis functions are of three distinct types: Connector Analysis, Tool Analysis, and Manual Materials Handling. As stated in the Introduction, a unique adaptation is required for each CREW CHIEF function. This section details the adaptations for the Task Analysis functions. When a function is called, the initial position and orientation of the man-model is that of the last successful positioning operation of the computer run.

4.2.1 CREW CHIEF Connector Analysis Function

The CREW CHIEF Connector Analysis function permits the user to evaluate the positioning and accessibility of electrical line connectors within the drawing. By defining the size, location, and direction of a connector within the drawing, the user may evaluate the man-model's ability to reach it with either hand. Within the range of the data bases, the amount of torque a technician may apply to the connector with the hand is displayed.

The following input parameters must be defined and passed to the CREW CHIEF Connector Analysis function through the common block CTRCTL:

IWHAND - Defines hand(s) to perform reach to the connector (INTEGER).

MOBILE - Defines the amount of motion allowed to the man-model (INTEGER).

IVOIDO - Obstacle Avoidance Flag (INTEGER).

ISIZE - Defines size of connector to be used (INTEGER).

NUMGRP - Defines grip type used to hold connector (INTEGER).

HEADAT (3) - Defines head point of attach vector for defining connector direction (REAL).

TAILAT (3) - Defines tail point of attach vector for defining connector direction (REAL).

CNTTBL (2) - Defines center of the region used to display the strength table (REAL).

NUMDSP - Defines display type (INTEGER).

UNITS - Defines drawing units per inch (REAL).

VUMAT(3,3) - A 3x3 array containing the view matrix (REAL).

IBGCTR - Debug flag for the Connector Analysis function (INTEGER).

The parameters are grouped into three general areas. The first five variables define hand, grip type, connector size, amount of man-model mobility, and obstacle avoidance status. The next two define the attach vectors, and the last five define the screen display and debugging flag status.

4.2.1.1 Hand, Grip Type, Connector Size, Mobility, and Obstacle Avoidance

IWHAND - Defines hand used to grasp connector. Only one hand, right or left, may be used. Set IWHAND to the index for the hand performing the reach to the connector.

1 - RIGHT

2 - LEFT

MOBILE - Defines the amount of movement allowed to the man-model. Set MOBILE to the index corresponding to the desired mobility.

1 - Full body movement allowed. (Currently not available for Connector function.)

2 - Upper body movement (including arms and shoulders).

3 - Arm and shoulder movement only.

IVOIDO - A flag that is set to either ignore or consider interference with obstacles in the work place during the reach portion of Task Analysis functions. Set IVOIDO to the index of the desired setting.

0 - Perform reach ignoring obstacles in work place.

1 - Reach around obstacles in work place.

ISIZE - Defines connector size (diameter in inches) to be used in the analysis. Set ISIZE to the index corresponding to the desired connector size.

1 - 0.9 inches

2 - 1.5 inches

3 - 2.0 inches

NUMGRP - Defines grip type to be used in the Connector Analysis. Set NUMGRP to the index of the desired grip.

1 - GRIP CENTER

2 - FUNCTIONAL GRIP

4.2.1.2 Attach Vector Definition

For the man-model to grasp the connector, the location and direction of the connector center axis must be known. The connector location is defined by the array HEADAT, and the direction by the array TAILAT.

HEADAT - The Attach Vector Head Point is the 3-D point where the connector center axis is placed in the drawing. Set the array coordinates as follows:

HEADAT (1) - X coordinate of point.

HEADAT (2) - Y coordinate of point.

HEADAT (3) - Z coordinate of point.

TAILAT - The Attach Vector Tail Point is the 3-D point which defines the direction of the connector with respect to the Attach Vector.

TAILAT (1) - X coordinate of point.

TAILAT (2) - Y coordinate of point.

TAILAT (3) - Z coordinate of point.

4.2.1.3 Screen Display Definition and Debugging Flag Status

CNTTBL - Defines the point where the center of the strength table will be plotted on the screen. Set the array CNTTBL to the screen coordinates of the desired plot center as follows:

CNTTBL (1) - X coordinate of the plot center point.

CNTTBL (2) - Y coordinate of the plot center point.

NUMDSP - Defines the display type used when drawing the man-model. Set NUMDSP to the index corresponding to the desired display type.

1 - WIRE FRAME
2 - SURFACED
3 - PROFILE

UNITS - Set UNITS to the number of units per inch. For example, if the drawing unit is in centimeters, set UNITS to 2.54; for millimeters, set UNITS to 25.4.

VUMAT(3,3) - A 3x3 array containing the view matrix which describes the orientation of the user's drawing with respect to the display screen. Available in most CAD systems, this array allows presentation of an uncluttered profile image of the man-model. [If the view matrix is not available in your system, set Display Type (NUMDSP=1) to wire frame and ignore the setting system positive axis points in the following directions: ignore the setting VUMAT.] CREW CHIEF assumes that the screen system positive axis points in the following directions: X points to the viewer's right, Y points up, and Z points toward the viewer. Set the array VUMAT as follows:

VUMAT(1-3,1) - The screen coordinates of the direction vector defining the positive X axis of the drawing coordinate system.

VUMAT(1-3,2) - The screen coordinates of the direction vector defining the positive Y axis of the drawing coordinate system.

VUMAT(1-3,3) - The screen coordinates of the direction vector defining the positive Z axis of the drawing coordinate system.

IBGCTR - A flag that defines whether or not control variables are to be printed. Set IBGCTR to the index desired.

0 - Do not print control variables
1 - Print control variables

Once the required input parameters have been defined, the CREW CHIEF subroutine CTRUSR is called to display the man-model in the final working position with the connector in its hand. If a successful connector analysis is completed, a strength table (expressed in torque) of Air Force maintenance technicians is displayed.

4.2.2 CREW CHIEF Tool Analysis Function

The Tool Analysis function is used to determine if the man-model can reach and place a tool to a specified point in a drawing. A wide variety of tool selections are provided. For certain types of wrenches, tables of strength capabilities, expressed in torque, are provided.

The following input parameters must be defined and passed to the CREW CHIEF Tool Analysis function through the common block TANCTL. The parameters are:

IWHAND - Defines hand(s) to hold the tool (INTEGER).

IDRTOL - Defines direction of tool in the hand (INTEGER).

MOBILE - Defines the amount of motion allowed to the man-model (INTEGER).

IVOIDO - Obstacle Avoidance Flag (INTEGER).

ATTCHD (3) - Defines attach point of tool in drawing (REAL).

ATTCTL (3) - Defines direction of tool attachment (REAL).

TOLDIR (3) - Defines direction of tool handle (REAL).

CNTTBL (2) - Defines center of strength table (REAL).

NUMDSP - Defines display type (INTEGER).

UNITS - Defines drawing units per inch (REAL).

VUMAT(3,3) - A 3x3 array containing the view matrix (REAL).

IBGTAN - Debug flag for the Tool Analysis function (INTEGER).

MODID - Defines tool model identification number (INTEGER).

MSZID - Defines the tool size identification number (INTEGER).

MSSZID - Defines the tool sub-size identification number (INTEGER).

MACSZ(10) - Defines the accessory size identification numbers (INTEGER).

MACMID(10) - Defines the accessory model identification numbers (INTEGER).

MACSSZ(10) - Defines the accessory sub-size identification numbers (INTEGER).

KLASID - Defines the tool class identification number (INTEGER).

MACCLS(10) - Defines the accessory class identification numbers (INTEGER).

NOACC - Defines the number of accessories (INTEGER).

The first nine variables define the tool model, tool size, and accessory model and size. The next four variables are used to define the hand(s) used, how the tool is held, the mobility allowed to the man-model, and the obstacle avoidance status. The next three variables define the tool location, and the last five define the screen display and debugging status.

4.2.2.1 Tool and Accessory Definition

A tool and any accessories to be used with that tool is defined by identifying its class, model, size, and sub-size.

KLASID - Defines the tool class identification number. Set this variable to the index of the desired tool class.

1 - WRENCHES
2 - SCREWDRIVERS
3 - PLIERS
4 - MISCELLANEOUS TOOLS
8 - USER DEFINED TOOLS

MODID - Defines the tool model identification number. Each tool class contains multiple models. For example, the tool class WRENCHES contains 12 different wrench models, such as ratchet wrench and open end wrench. Appendix A contains a list of the tool models associated with each tool class. Set this variable to the index of the desired tool model.

MSZID - Defines the tool model size identification number. Each tool model may have multiple sizes. Appendix A shows the valid sizes for each tool model. Set this variable to the index of the desired tool model size.

MSSZID - Defines the tool model sub-size identification number. Each tool model size may have multiple sub-sizes. Appendix A shows the valid sub-sizes for each tool model size. Set this variable to the index of the desired tool model sub-size.

NOACC - Defines the number of accessories that are to be attached to the tool model. Some tool models have accessories associated with them. For some tool models the accessory definition may be required before the tool can be used. For example, the tool model ratchet wrench requires that a socket be selected. Other accessories may be optionally defined. For example the tool model hammer can be used with or without the selection of the optional accessory chisel. Appendix A lists the accessories available for each tool model. Set this variable to the number of accessories defined for the tool model (maximum of 10).

MACCLS - Defines the accessory class identification numbers. Set this array to the index of the desired accessory class as follows:

MACCLS(1) - Class identification number for the first accessory.

MACCLS(2) - Class identification number for the second accessory.

MACCLS(3) - Class identification number for the third accessory.

.

.

.

MACCLS(10) - Class identification number for the tenth accessory.

Accessory classes are:

5 - SOCKET
6 - EXTENSION
7 - CHISEL

MACMID - Defines the accessory model identification numbers. Each accessory class may contain numerous models. For example, the accessory class SOCKET contains four models, such as regular and hex-head sockets. Appendix A contains a list of the models that are associated with each accessory class. Set this array to the index of the desired accessory model as follows:

MACMID(1) - Model identification number for the first accessory.

MACMID(2) - Model identification number for the second accessory.

MACMID(3) - Model identification number for the third accessory.

.

.

.

MACMID(10) - Model identification number for the tenth accessory.

MACSZ - Defines the accessory size identification numbers. Each accessory class contains multiple sizes. Appendix A concains a list of model sizes that are associated with each accessory. Set this array to the index of the desired accessory model size as follows:

MACSZ(1) - Size index for the first accessory.

MACSZ(2) - Size index for the second accessory.

MACSZ(3) - Size index for the third accessory.

.

.

.

MACSZ(10) - Size index for the tenth accessory.

MACSSZ - Defines the accessory sub-size identification numbers. Each accessory model size may have multiple sub-sizes. Appendix A shows the valid sub-sizes for each accessory model size. Set this array to the index of the desired accessory sub-size as follows:

MACSSZ(1) - Sub-size index number for the first accessory.

MACSSZ(2) - Sub-size index number for the second accessory.

MACSSZ(3) - Sub-size index number for the third accessory.

.

.

.

MACSSZ(10) - Sub-size index number for the tenth accessory.

4.2.2.2 Hand(s), Tool, Tool Direction, Mobility, and Obstacle Avoidance Definition

IWHAND - Defines hand(s) in which tool is to be held. Set IWHAND to the index for the hand(s) to be used.

1 - RIGHT
2 - LEFT
3 - BOTH

NOTE: For all wrenches, the right hand grasps the shaft of the tool and the left hand holds the head of the tool when both hands are used. For example, if a torque wrench is used with both hands, the right hand will hold the handle of the wrench and the left will hold the head of the wrench in place. Some tools must be held in only the right or left hand, and not every tool uses all three grip types.

IDRTOL - Defines the direction of the tool in the hand. Most tools may be held so that the point of application of the tool is above the thumb (regular grip) or below the little finger (reverse grip). Screwdrivers and nutdrivers may be held so that the point of application is parallel to the hand (alternate grip). Set IDRTOL to the index corresponding to the desired tool direction in the hand.

1 - REGULAR
2 - REVERSE
3 - ALTERNATE

MOBILE - Defines the amount of motion allowed to the man-model. Set MOBILE to the index for the desired man-model mobility.

1 - Full body movement allowed. (Currently not available for Tool function.)

2 - Upper body movement (including arms and shoulders).

3 - Arm and shoulder movement only.

IVOIDO - A flag that is set to either ignore or consider interference with obstacles in the work place during the reach portion of Task Analysis functions. Set IVOIDO to the index for the desired setting.

0 - Perform reach ignoring obstacles in work place
1 - Reach around obstacles in work place

4.2.2.3 Tool Location Definition

The next three parameters define the position and orientation of the tool in the drawing. Tool position and orientation are based upon the assumption that the tool will be applied at the Tool Attach Point (ATTCHD) along the Tool Attach Vector defined from the Tool Attach Point to the Tool Direction Point (ATTCTL). The Tool Attach Vector lies parallel to the bolt. Tool Handle Direction is defined by the vector from the Tool Direction Point to the Tool Handle Direction Point (TOLDIR).

ATTCHD - The 3-D point in the user's drawing where the tool will attach (i.e., a nut or a screw). Set the array ATTCHD to the desired 3-D point in drawing coordinates as follows:

ATTCHD (1) - X coordinate of point.
ATTCHD (2) - Y coordinate of point.
ATTCHD (3) - Z coordinate of point.

ATTCTL - The Tool Direction Point is the 3-D point defining the direction of the tool attachment with respect to the Tool Attach Point. Set the array ATTCTL to the desired 3-D point in drawing coordinates as follows:

ATTCTL (1) - X coordinate of point.
ATTCTL (2) - Y coordinate of point.
ATTCTL (3) - Z coordinate of point.

TOLDIR - Tool Handle Direction is the 3-D point that defines the direction of the tool handle with respect to the Tool Direction Point. Set the array TOLDIR to the 3-D point in drawing coordinates as follows:

TOLDIR (1) - X coordinate of point.
TOLDIR (2) - Y coordinate of point.
TOLDIR (3) - Z coordinate of point.

<u>NOTE</u>: For file, screwdrivers and nutdrivers, the Tool Attach Vector is the same as the Tool Handle Direction. The variable TOLDIR is not defined for file, screwdrivers, and nutdrivers.

4.2.2.4 Screen Display Definition and Debugging Flag Status

CNTTBL - Defines the point where the center of the strength table will be plotted on the screen. Appendix A lists the tool models associated strength data. Set the array CNTTBL to the screen coordinates of the desired plot center as follows:

CNTTBL (1) - X coordinate of the plot center point.

CNTTBL (2) - Y coordinate of the plot center point.

NUMDSP - Defines the display type used when drawing the man-model. Set NUMDSP to the index of the desired display type.

1 - WIRE FRAME
2 - SURFACED
3 - PROFILE

UNITS - Set UNITS to the number of units per inch. For example, if the drawing unit is in centimeters, set UNITS to 2.54; for millimeters, set UNITS to 25.4.

VUMAT(3,3) - A 3x3 array containing the view matrix which describes the orientation of the user's drawing with respect to the display screen. Available in most CAD systems, this array allows presentation of an uncluttered profile image of the man-model. (If the view matrix is not available in your system, set Display Type [NUMDSP=1] to wire frame and ignore the setting VUMAT.) CREW CHIEF assumes that the screen system positive axis points in the following directions: X points to the viewer's right, Y points up, and Z points toward the viewer. Set the array VUMAT as follows:

VUMAT(1-3,1) - The screen coordinates of the direction vector defining the positive X axis of the drawing coordinate system.

VUMAT(1-3,2) - The screen coordinates of the direction vector defining the positive Y axis of the drawing coordinate system.

VUMAT(1-3,3) - The screen coordinates of the direction vector defining the positive Z axis of the drawing coordinate system.

IBGTAN - A flag that defines whether or not control variables are to be printed. Set IBGTAN to the desired index.

0 - Do not print control variables

1 - Print control variables

Once all of the necessary parameters for the Tool Analysis function have been defined through the common block TANCTL, the CREW CHIEF subroutine TANUSR is called. The CREW CHIEF Tool Analysis function generates a display of the man-model, in the final reach position, with the desired tool in the hand(s). For the appropriate wrenches a table of strength capabilities (expressed in foot pounds of torque) is displayed for a successful placement of the tool in the drawing.

4.2.3 Manual Materials Handling Functions

The six separate Manual Materials Handling functions are CARRY, HOLD AND POSITION, LIFT, PUSH, PULL, and REACH. The input parameters for all Manual Materials Handling functions are passed through the common blocks CRYCTL, HLDCTL, LFTCTL, PSHCTL, PLLCTL, and RECCTL, respectively. The first input variable under XXXCTL is ITASK, which defines the task type to be performed.

ITASK	= 7	CARRY TASK
	= 8	HOLD AND POSITION TASK
	= 9	LIFT TASK
	= 10	PUSH TASK
	= 11	PULL TASK
	= 14	REACH TASK

Once the task type is defined by selecting the appropriate index, the input variables for that task type must be defined. Since some of the input variables are common to multiple task types, and some are not, each task type requires a unique adaptation (see Appendix B). The details of these adaptations are covered in the following paragraphs.

4.2.3.1 CREW CHIEF Carry Function

The CREW CHIEF Carry function provides the user the MIL-STD 1472 maximum weight limits, based on object depth, which correlate to the distance of the center of mass of the object from the body. For an object of the defined dimensions, a maximum weight which can be carried for a distance of ten feet is predicted. The predicted maximum weight is expressed in pounds for the 1st, 5th, 50th, 95th and 99th percentiles for the gender population selected during the Initialization function. Ceiling height, expressed as the distance above the support platform, is a variable in the function that may create a barrier. The following input variables must be defined (see Appendix B):
ITASK - Determines which task is to be performed (INTEGER).

NUMHDL - Defines whether the object does, or does not, have handles (INTEGER).

IWHAND - Defines the hand(s) used to perform the task (INTEGER).

MOBILE - Defines mobility of man-model (REAL).

ISELCE - Selects the mode for defining the ceiling height (INTEGER).

CEILHT - Defines the ceiling height (REAL).

DMKOBJ(3) - Defines the dimensions of the object (height, weight, and depth) (REAL).

CNTTBL(2) - Defines the center of the screen area to display the CREW CHIEF Strength Table (REAL).

IVOIDO - Obstacle Avoidance Flag (INTEGER).

NUMDSP - Defines display type (INTEGER).

UNITS - Defines drawing units per inch (REAL).

VUMAT(3,3) - A 3 x 3 array containing the view matrix (REAL).

IBGCRY - Debug flag for the CARRY function (INTEGER).

The first variable defines the task. The next three variables define the object's handles, and hand(s) used for the task, and mobility. The next three variables define the ceiling height selection mode, the ceiling height, and the object's dimensions. The last six variables define the screen display, obstacle avoidance, and debugging flag status.

4.2.3.1.1 Task Identifier

ITASK - To perform a CARRY Task, ITASK must be set to 7.

4.2.3.1.2 Handle, Hand(s), Mobility, and Ceiling Height Selection Mode Definitions

NUMHDL - Defines whether the object does, or does not, have a handle(s). Set NUMHDL to the index for the number of handles on the object.

0 - NO HANDLE
1 - ONE HANDLE
2 - TWO HANDLES

IWHAND - Defines which hand(s) will be used to perform the task. Set IWHAND to the index for the required hand selection.

1 - RIGHT
2 - LEFT
3 - BOTH

- **When NUMHDL = 0 or 2 (No handle or two handles)**, IWHAND must be set to 3.

- **When NUMHDL = 1 (One handle)**, IWHAND must be set to 1 or 2 (RIGHT or LEFT).

MOBILE - Defines the amount of movement allowed the man-model. For CARRY, MOBILE must be set to 1 to indicate full body movement.

4.2.3.1.3 Ceiling Height and Object Dimension Definition

ISELCE - Defines the mode for selecting ceiling height. Set ISELCE to the index for the desired mode.

1 - SELECT FROM SCREEN
2 - KEY IN

CEILHT - The procedure for defining ceiling height depends on the selection mode definition made in ISELCE.

- **When ISELCE = 1 (Select from screen)**, CEILHT is defined by selecting a line in the drawing. The element identifier is translated into height in the routine CCUEXF.

- **When ISELCE = 2 (Key in)**, CEILHT is defined by keying in the height above the support platform in drawing units.

NOTE: The ceiling height must be greater than the object height.

DMKOBJ - Defines the height, width, and depth of the object to be carried. Set the array DMKOBJ to the dimensions of the object, in drawing units, as follows:

DMKOBJ(1) - Height of the object.
DMKOBJ(2) - Width of the object.
DMKOBJ(3) - Depth of the object.

NOTE: There are restrictions to the minimum and maximum dimensions of the object. (Minimum dimensions must be at least 1 inch; maximum height and width is 30 inches.)

4.2.3.1.4 Screen Display Definition, Obstacle Avoidance Definition, and Debugging Flag Status

CNTTBL - Defines the point where the strength table center will be plotted on screen. Set the array CNTTBL to screen coordinates of desired plot center as follows:

CNTTBL (1) - X coordinate of the plot center point.

CNTTBL (2) - Y coordinate of the plot center point.

IVOIDO - A flag that is set to either ignore or consider interference with obstacles in the work place during the reach portion of task analysis functions. Set IVOIDO to the index of the desired setting.

0 - Perform reach ignoring obstacles in work place.

1 - Reach around obstacles in work place.

NUMDSP - Defines the display type used when drawing the man-model. Set NUMDSP to the index corresponding to the desired display type.

1 - WIRE FRAME
2 - SURFACED
3 - PROFILE

UNITS - Set UNITS to the number of units per inch. For example, if the drawing unit is in centimeters, set UNITS to 2.54; for millimeters, set UNITS to 25.4.

VUMAT(3,3) - A 3x3 array which describes the orientation of the user's drawing with respect to the display screen. Available in most CAD systems, this array allows presentation of an uncluttered profile image of the man-model. (If the view matrix is not available in your system, set Display Type [NUMDSP=1] to wire frame and ignore the setting VUMAT.) CREW CHIEF assumes that the screen system positive axis points in the following directions: X points to the viewer's right, Y points up, and Z points toward the viewer. Set the array VUMAT as follows:

VUMAT(1-3,1) - The screen coordinates of the direction vector defining the positive X axis of the drawing coordinate system.

VUMAT(1-3,2) - The screen coordinates of the direction vector defining the positive Y axis of the drawing coordinate system.

VUMAT(1-3,3) - The screen coordinates of the direction vector defining the positive Z axis of the drawing coordinate system.

IBGCRY - A flag that defines whether or not control variables are to be printed. Set IBGCRY to the index desired.

0 - Do not print control variables.
1 - Print control variables.

Once the variables have been defined, the CREW CHIEF subroutine CRYUSR is called. The CREW CHIEF program performs the Carry analysis, displays the man-model in the final carry position, and, if successful, will display the strength table. If too many obstacles prevent the man-model from performing the task, arrows are displayed indicating the points of interference.

4.2.3.2 CREW CHIEF Hold Function

The CREW CHIEF Hold function provides the maximum capabil-ity, expressed in pounds for the 1st, 5th, 50th, 95th, and 99th percentiles of the maintenance technician population to hold an object at a specified position with one hand while attaching supporting fasteners with the other hand. The following input parameters must be defined (see Appendix B):

ITASK - Determines which task is to be performed (INTEGER).

IHOLD - Defines the type of hold (INTEGER).

IWHAND - Defines the hand used to perform the task (INTEGER).

NUMHDL - Defines number of handles on object (INTEGER).

MOBILE - Defines mobility of man-model (INTEGER).

ISELBR - Selects the barrier type (INTEGER).

ISELCE - Selects the mode for defining ceiling height (INTEGER).

CEILHT - Defines the ceiling height (REAL).

ISELDI - Selects the mode for defining the object dimensions (INTEGER).

DMSOBJ(3) - Element identifier of height, width, and depth selected from screen (REAL).

SELRST(50) - Element identifiers for additional lines selected from the screen to define the object's surfaces (50 maximum) (REAL).

ISELNO - Defines number of additional lines selected for object surface definition (REAL).

DMKOBJ(3) - Defines dimensions of the object (height x width x depth) keyed in (REAL).

ALINE1 - Defines element number of first line selected to define the attach plane (REAL).

ALINE2 - Defines element number of second line selected to define the attach plane (REAL).

ATTPT(3) - Defines the attach point of the object to the attach plane (REAL).

CNTTBL(2) - Defines the center of the screen area to display the strength table (REAL).

IVOIDO - Obstacle avoidance flag (INTEGER).

NUMDSP - Defines display type (INTEGER).

UNITS - Defines drawing units per inch (REAL).

VUMAT(3,3) - A 3 x 3 array containing the view matrix (REAL).

IBGHLD - Debug flag for the HOLD function (INTEGER).

The first variable defines the task. The next seven variables define the hold type, barrier type, ceiling height,definition mode, ceiling height, hand used, number of handles, and the man-model mobility. The next eight define object dimensions, the attach plane, and the attach point. The last four define the center of plot, screen display, obstacle avoidance, and debugging flag status.

4.2.3.2.1 Task Identifier

ITASK - To perform a HOLD Task, ITASK must be set to 8.

4.2.3.2.2 Hold Type, Barrier Type, Ceiling Height Definition Mode, Ceiling, Hand Used, Number of Handles, and Mobilitv Definitions

IHOLD - Defines the type of hold. Set IHOLD to the required index.

1 - Object held against wall.
2 - Object held against ceiling.

- **When IHOLD = 1 (Object held against wall)**, the following variable must be defined.

ISELBR - Select the barrier type. Enter the index corresponding to the desired barrier type.

1 - NO BARRIER
2 - VERTICAL BARRIER
3 - CEILING BARRIER

- **When ISELBR = 1 or 2 (No, or, vertical barrier)**, no additional barrier definition is required.

- **When ISELBR = 3 (Ceiling barrier)** the following variables must be defined:

ISELCE - Selects the mode for defining ceiling height. Set ISELCE to the desired index.

1 - SELECT FROM SCREEN
2 - KEY IN

CEILHT - The procedure for defining ceiling height depends on the selection mode definition made in ISELCE.

- **When ISELCE = 1 (Select from screen)**, CEILHT is defined by selecting a line in the drawing. The element identifier is translated into height in the routine CCUEXF.

- **When ISELCE = 2 (Key in)**, CEILHT is defined by keying in the height above the support platform in drawing units.

<u>NOTE:</u> Ceiling height must be greater than the object.

NUMHDL - Defines number of handles used to perform the task. For hold, NUMHDL is always 0 (no handles).

IWHAND - Defines the hand used to perform the task. Set IWHAND to the required index.

1 - RIGHT
2 - LEFT

MOBILE - Defines the amount of movement allowed the man-model. Set MOBILE to the index for the mobility desired for the analysis.

1 - Full body movement allowed. (Currently not available for Hold function.)

2 - Upper body movement (including arms and shoulders).

3 - Arm and shoulder movement only.

4.2.3.2.3 Object Dimension, Attach Plane, and Attach Point Definition

ISELDI - Selects the mode for defining the object dimensions. Set ISELDI to the index for the desired mode.

1 - SELECT FROM SCREEN
2 - KEY IN

- **When ISELDI = 1 (Select from screen)**, the following variables must be defined:

DMSOBJ - Defines the height, width and depth of the object. Set the array DSMOBJ as follows:

DMSOBJ(1) - Select a line from the screen to define the height of the object.

DMSOBJ(2) - Select a line from the screen to the screen to define the width of the object.

DMSOBJ(3) - Select a line from the screen to define the depth of the object.

SELRST(50) - Select additional lines, if required, to define the object surface (50 maximum)

ISELNO - Enter the number of additional lines selected under SELRST.

- **When ISELDI = 2 (Key in)**, the following variables are defined.

DMKOBJ - Defines the height, width, and depth of the object. Set the array DMKOBJ to the dimensions of the object, in drawing units, as follows:

DMKOBJ(1) - Height of the object.
DMKOBJ(2) - Width of the object.
DMKOBJ(3) - Depth of the object.

<u>NOTE:</u> There are restrictions to the minimum dimensions of the objects. (Minimum for any dimension is 1 inch.)

ALINE1 - Element identifier of the first selected line to define the attach plane in the drawing.

ALINE2 - Element identifier of the second selected line to define the attach plane in the drawing.

ATTPT - Defines the point where the center of the object's attaching face will be positioned on the plane. The array ATTPT may be filled by selecting a 3-D point from the screen, or by entering the screen coordinates of the point as follows:

ATTPT(1) - X coordinate of the point.
ATTPT(2) - Y coordinate of the point.
ATTPT(3) - Z coordinate of the point.

4.2.3.2.4 Screen Display, Obstacle Avoidance Definition, and Debugging Flag Status

CNTTBL - Defines the point where the center of the strength table will be plotted on the screen. Set the array CNTTBL to the screen coordinates of the desired plot center as follows:

CNTTBL (1) - X coordinate of the plot center
CNTTBL (2) - Y coordinate of the plot center point.

IVOIDO - A flag that is set either to ignore or consider interference with obstacles in the work place during the reach portion of Task Analysis functions. Set IVOIDO to the index for the desired setting

0 - Perform reach ignoring obstacles in work place.

1 - Reach around obstacles in work place.

NUMDSP - Defines the display type used when drawing the man-model. Set NUMDSP to the index corresponding to the desired display type.

1 - WIRE FRAME
2 - SURFACED
3 - PROFILE

UNITS - Set UNITS to the number of units per inch. For example, if the drawing unit is in centimeters, set UNITS to 2.54; for millimeters, set UNITS to 25.4.

VUMAT(3,3) - A 3x3 array which describes the orientation of the user's drawing with respect to the display screen. Available in most CAD systems, this array allows presentation of an uncluttered profile image of the man-model. (If the view matrix is not available in your system, set Display Type [NUMDSP=1] to wire frame and ignore the setting VUMAT.) CREW CHIEF assumes that the screen system positive axis points in the following directions: X points to the viewer's right, Y points up, and Z points toward the viewer. Set the array VUMAT as follows:

VUMAT(1-3,1) - The screen coordinates of the direction vector defining the positive X axis of the drawing coordinate system.

VUMAT(1-3,2) - The screen coordinates of the direction vector defining the positive Y axis of the drawing coordinate system.

VUMAT(1-3,3) - The screen coordinates of the direction vector defining the positive Z axis of the drawing coordinate system.

IBGHLD - A flag that defines whether or not control variables are to be printed. Set IBGHLD to the index desired.

0 - Do not print control variables
1 - Print control variables

When all input variables have been defined, the CREW CHIEF subroutine HLDUSR is called. The man-model is displayed performing the HOLD task with a strength table displayed, or attempting the HOLD task if he cannot reach the object. If too many obstacles prevent the man-model from performing the task, arrows are displayed indicating the points of interference.

4.2.3.3 CREW CHIEF Lift Function

The CREW CHIEF Lift function is used to evaluate the ability of the man-model to lift objects. Lift capabilities for Air Force maintenance technicians, with the man-model in the final lift position, will be displayed when the function is completed. The following input variables must be defined (see Appendix B):

ITASK - Determines which task is to be performed (INTEGER).

NUMHDL - Defines whether the object does, or does not, have a handle (REAL).

IWHAND - Defines hand(s) to perform the lift (INTEGER).

HDL1BG (3) - Defines first end point of the object's handle (REAL).

HDL1EN (3) - Defines second end point of the object's handle (REAL).

MOBILE - Defines the amount of mobility the man-model is allowed (INTEGER).

ISELVD - Selects the mode for defining the lift distance (INTEGER).

HGTFLT - Defines height above the support plane to which the object is to be lifted (REAL).

DISOBJ - Defines horizontal distance from man-model position reference point to object (REAL).

ISELNO - Defines number of additional line segments selected for the object (REAL).

SELRST(50) - Defines element identifiers for additional lines selected from the screen to define the object's surfaces (50 maximum) (REAL).

ISELDI - Selects the mode for defining the object dimensions (INTEGER).

DMSOBJ(3) - Defines element identifiers of height, width, depth selected from the screen (REAL).

DMKOBJ(3) - Defines dimensions of the object (height x width x depth) keyed in (REAL).

CNTTBL(2) - Defines center point of the screen region to display the CREW CHIEF strength table (REAL).

IVOIDO - Obstacle Avoidance Flag (INTEGER).

NUMDSP - Defines display type (INTEGER).

UNITS - Defines drawing units per inch (REAL).

VUMAT(3,3) - A 3x3 array containing the view matrix.

IBGLFT - Debug flag for the LIFT function (INTEGER).

The first variable defines the task. The next four variables define the object's handles (if they exist), the hand(s) to be used for the task, and the amount of mobility allowed the man-model. The next eight variables define the height above the support plane to which the object is to be lifted and the dimensions of the object. The last six variables define the screen display and obstacle avoidance and debugging flag status.

4.2.3.3.1 Hand, Handle, and Mobility Definition

NUMHDL - Defines whether the object does or does not have a handle. Set NUMHDL to the index describing the object's handle.

0 - NO HANDLE
1 - ONE HANDLE

- **When NUMHDL = 1 (One handle), and ISELDI = 1 (Object dimension selected from screen)**, define the following variables:

HDL1BG - Defines one end point of a line extending through the center of the lift handle. The hand will be oriented with the thumb closest to this end point of the handle. Define this point in drawing coordinates as follows:

HDL1BG (1) - X coordinate of point.
HDL1BG (2) - Y coordinate of point.
HDL1BG (3) - Z coordinate of point.

HDL1EN - Defines the other end point of a line extending through the center of the handle. Define this point in drawing coordinates as follows:

HDL1EN (1) - X coordinate of point.
HDL1EN (2) - Y coordinate of point.
HDL1EN (3) - Z coordinate of point.

IWHAND - Defines which hand(s) will be used to perform the task. Selection depends on the NUMHDL selection:

1 - RIGHT
2 - LEFT
3 - BOTH

- **When NUMHDL = 0 (No handle)**, set IWHAND to 3.

- **When NUMHDL = 1 (One handle)**, set IWHAND to 1 or 2.

NOTE: Current strength data bases are not available for one-handed lifts of objects without handles. If 0 is set, a message informing you that one-handed lift without handles is not possible will be displayed.

Also, there is no strength data base for two-handed lifts of objects with a handle. Thus, if both hands and one handle are selected, a right-handed lift will occur.

MOBILE - Defines the amount of movement allowed the man-model. Set MOBILE to the index for the mobility desired in the analysis.

1 - Full body movement allowed. (Currently not available for Lift function.)

2 - Upper body movement (including arms and shoulders).

3 - Arm and shoulder movement only.

4 - No movement allowed. No movement is allowed when the man-model has previously performed a task and the current task analysis is to be performed without changing the man-model's position.

4.2.3.2.2 Lift Distance Selection Mode, Lift Distance, and Object Dimension Definition

ISELVD - Mode for selecting lift distance. Set ISELVD to the desired index.

1 - SELECT FROM SCREEN
2 - KEY IN

- **When ISELVD = 1 (Select from screen)**, define the following variable:

HGTPLT - Element identifier of a line segment selected from the screen to define the lift distance. The program computes and enters the value for HGTPLT.

- **When ISELVD = 2 (Key in),**
  define the following variable:

HGTPLT - Height above the support platform, in drawing units, to which the object is lifted.

DISOBJ - Horizontal distance, in drawing units, from man-model position reference point to object.

ISELDI - Selects the mode for defining the object dimensions. Set ISELDI to the index for the desired mode.

1 - SELECT FROM SCREEN
2 - KEY IN

- **When ISELDI = 1 (Select from screen),**
  the following variables must be defined:

DMSOBJ - Defines the height, width, and depth of the object. Set the array DSMOBJ as follows:

DMSOBJ(1) - Element identifier of a line selected from the screen to define the height of the object.

DMSOBJ(2) - Element identifier of a line selected from the screen to define the width of object.

DMSOBJ(3) - Element identifier of a line selected from the screen to define the depth of the object.

SELRST(50)- If required, additional element identifiers of lines defining the object's surfaces (50 maximum).

ISELNO - The number of additional lines selected under SELRST.

- **When ISELDI = 2 (Key in)**, the following variable is defined.

DMKOBJ - Defines the height, width, and depth of the object. Set the array DMKOBJ to the dimensions of the object, in drawing units, as follows:

DMKOBJ(1) - Height of the object.
DMKOBJ(2) - Width of the object.
DMKOBJ(3) - Depth of the object.

NOTE: There are restrictions to the minimum and maximum dimensions of the object. Maximum dimensions are 12, 18, and 30 inches in any combination of height, width, and depth. The minimum dimension must be at least 1 inch for any dimension. For example, 10 x 20 x 16 inches is acceptable. 14 x 18 x 30 inches is not.

4.2.3.3.3 Screen Display, Obstacle Avoidance Definition, and Debugging Flag Status

CNTTBL - Defines the point where the center of the strength table will be plotted on the screen. Set the array CNTTBL to the screen coordinates of the desired plot center as follows:

CNTTBL (1) - X coordinate of the plot center point.

CNTTBL (2) - Y coordinate of the plot center point.

IVOIDO - A flag that is set to either ignore, or consider, interference with obstacles in the work place during the reach portion of Task Analysis functions. Set IVOIDO to the index of the desired setting.

0 - Perform reach ignoring obstacles in work place.

1 - Reach around obstacles in work place.

NUMDSP - Defines the display type used when drawing the man-model. Set NUMDSP to the index of the desired display type.

1 - WIRE FRAME

2 - SURFACED

3 - PROFILE

UNITS - Set UNITS to the number of units per inch. centimeters, set UNITS to 2.54; for millimeters, set UNITS to 25.4.

VUMAT(3,3) - A 3x3 array which describes the orientation of the user's drawing with respect to the display screen. Available in most CAD systems, this array allows presentation of an uncluttered profile image of the man-model. (If the view matrix is not available in your system, set Display Type [NUMDSP=1] to wire frame and ignore the setting VUMAT.) CREW CHIEF assumes that the screen system positive axis points in the following directions: X points to the viewer's right, Y points up, and Z points toward the viewer. Set the array VUMAT as follows:

VUMAT(1-3,1) - The screen coordinates of the direction vector defining the positive X axis of the drawing coordinate system.

VUMAT(1-3,2) - The screen coordinates of the direction vector defining the positive Y axis of the drawing coordinate system.

VUMAT(1-3,3) - The screen coordinates of the direction vector defining the positive Z axis of the drawing coordinate system.

IBGLFT - A flag that defines whether or not control variables are to be printed. Set IBGLFT to the index desired.

0 - Do not print control variables
1 - Print control variables

Once the input parameters have been defined, the subroutine LFTUSR should be called. The CREW CHIEF program performs the lift analysis, displays the man-model in the final lift position, and if successful, displays the strength table. If too many obstacles prevent the man-model from performing the task, arrows are displayed indicating the points of interference.

4.2.3.4 CREW CHIEF Push Function

The CREW CHIEF Push function evaluates the man-model's ability to push an object at a specified height above the man-model support platform. The program displays the man-model in the final pushing position and also displays a strength table. The following input parameters must be defined (see Appendix B):

ITASK - Determines which task is to be performed (INTEGER).

IWHAND - Defines hand(s) to perform the push (INTEGER).

MOBILE - Defines the amount of motion the man-model is allowed (INTEGER).

IOBCLR - Flag to set the criticality of object clearance (INTEGER).

FRICTN - Coefficient of Friction between boots and the support surface (REAL).

NUMHDL - Defines the number of handles on the object (INTEGER).

HDL1BG (3) - Defines one end point of the first handle on the object to be pushed (REAL).

HDL1EN (3) - Defines other end of the first handle (REAL).

HDL2BG (3) - Defines one end point of the second hardle on the object to be pushed (REAL).

HDL2EN (3) - Defines other end of the second handle (REAL).

ISELVD - Selects the mode for defining the object distances (INTEGER).

DISOBJ - Defines horizontal distance from the man-model position reference point to the object (REAL).

HGTPLT - Defines vertical height from man-model

ISELDI - Selects the mode for defining the object dimensions (INTEGER).

DMSOBJ(3) - Element identifier of height, width, and depth selected from screen (REAL).

SELRST(50) - Element identifiers for additional lines selected from the screen to define the object's surfaces (50 maximum) (REAL).

ISELNO - Number of additional line segments selected to define the object dimensions (INTEGER).

DMKOBJ(3) - Defines dimensions of the object (height x width x depth) keyed in (REAL).

CNTTBL (2) - Defines center of region to display strength table (REAL).

IVOIDO - Obstacle Avoidance Flag (INTEGER).

NUMDSP - Defines display type (INTEGER).

UNITS - Defines drawing units per inch (REAL).

VUMAT(3,3) - A 3x3 array containing the view matrix (REAL).

IBGLFT - Debug flag for the PUSH function (INTEGER).

The first variable indicates the task to perform. The next four variables define the hand(s) to be used, the mobility to be allowed to the man-model, the obstacle clearance status, and the coefficient of friction between the boots and the surface. The next thirteen variables define the handles, horizontal and vertical distances, and the object dimensions. The last six variables define the screen display, and the obstacle avoidance and debugging flag status.

4.2.3.4.1 Task Identifier

ITASK - Identifies the task to perform. ITASK must be 10 for Push.

4.2.3.4.2 Hand(s), Mobility, Object Clearance, and Friction Definition

IWHAND - Defines which hand(s) will be used to perform the push. Set IWHAND to the index for the required hand selection.

3 - BOTH

NOTE: Both hands are always used when performing the push.

MOBILE - Defines the amount of movement allowed the man-model. Set MOBILE to the index for the mobility desired in the analysis.

1 - Full body movement allowed.

2 - Upper body movement (including arms and shoulders).

3 - Arm and shoulder movement only.

4 - No movement allowed. No movement is allowed when the man-model has previously performed a task and the current task analysis is to be performed without changing the man-model's position.

IOBCLR - Defines whether the object clearance is critical or non-critical. Object clearance is considered "critical" when it is desirable that 95% of the resultant pushing force is contained in the horizontal component. For example, pushing an electronic box into a rack. Set IOBCLR to the required index.

1 - CRITICAL

2 - NON-CRITICAL

FRICTN - Defines the coefficient of friction between the boot and the man-model support surface. Set FRICTN to an appropriate value between 0.1 and 1.0, for the surface on which the task is being performed.

4.2.3.4.3 Handle, Horizontal and Vertical Distances, and Object Dimension Definitions

NUMHDL - Defines the number of handles on the object. Set NUMHDL to the index for the number of handles.

0 - NONE
1 - ONE
2 - TWO

- **When NUMHDL = 1 (One handle), and ISELDI = 1 (Object dimensions selected from screen)**, the following arrays must be defined.

HDL1BG - One end of the line extending through the center of the handle of the object. The hand will be oriented with the thumb closest to this end of the handle. Define this point in drawing coordinates as follows:

HDL1BG(1) - X coordinate of the point.

HDL1BG(2) - Y coordinate of the point.

HDL1BG(3) - Z coordinate of the point.

HDL1EN - The other end of the line extending through the center of the object's handle. Define this point in drawing coordinates as follows:

HDL1EN(1) - X coordinate of the point.

HDL1EN(2) - Y coordinate of the point.

HDL1EN(3) - Z coordinate of the point.

- **When NUMHDL = 2 (Two handles)**, the following arrays must be defined:

HDL1BG - One end of the line extending through the center of the first handle of the object. The right hand will be oriented with the thumb closest to this end of the handle. Define this point in drawing coordinates as follows:

HDL1BG(1) - X coordinate of the point.

HDL1BG(2) - Y coordinate of the point.

HDL1BG(3) - Z coordinate of the point.

HDL1EN - The other end of the line extending through the center of the object's first handle. Define this point in drawing coordinates as follows:

HLD1EN(1) - X coordinate of the point.

HDL1EN(2) - Y coordinate of the point.

HDL1EN(3) - Z coordinate of the point.

HDL2BG - One end of the line extending through the center of the second handle of the object. The left hand will be oriented with the thumb closest to this end of the handle. Define this point in drawing coordinates as follows:

HDL2BG(1) - X coordinate of the point.

HDL2BG(2) - Y coordinate of the point.

HDL2BG(3) - Z coordinate of the point.

HDL2EN - The other end of the line extending through the center of the second handle of the object. Define this point in drawing coordinates as follows:

HDL2EN(1) - X coordinate of the point.

HDL2EN(2) - Y coordinate of the point.

HDL2EN(3) - Z coordinate of the point.

ISELVD - Selects the mode for defining the horizontal and vertical distances from the man-model and the support platform to the object. Set ISELVD to the index for the desired mode.

1 - SELECT FROM SCREEN
2 - KEY IN

- **When ISELVD = 1 (Select from screen)**, a line is selected from the screen and the program internally computes and sets the horizontal distance from the man-model to the object (DISOBJ), and the vertical distance from the support platform (HGTPLT).

- **When ISELVD = 2 (Key in)**, the following variables must be defined:

DISOBJ - Enter the horizontal distance, in drawing units, from the man-model's position reference point to the object to be pushed.

HGTPLT - Set the vertical height from the man-model's support platform to the object, in drawing units.

ISELDI - Selects the mode for defining the object dimensions. Set ISELDI to the index for the desired mode.

1 - SELECT FROM SCREEN
2 - KEY IN

- **When ISELDI = 1 (Select from screen)**, the following array must be defined:

DMSOBJ - Defines the height, width and depth of the object. Set the array DSMOBJ as follows:

DMSOBJ(1) - Element identifier of a line selected from the screen to define the height of the object.

DMSOBJ(2) - Element identifier of a line selected from the screen to define the width of the object.

DMSOBJ(3) - Element identifier of a line selected from the screen to define the depth of the object.

ISELNO - Number of line segments selected from the screen todefine object dimensions.

For irregular objects, other elements besides height, width, and depth may be needed to define the object's surface. The array SELRST is set to the identifiers of the other elements selected to define object dimensions. The element identifiers are decoded in the subroutine CCUEXF.

- **When ISELDI = 2 (Key in)**, the following array is defined.

DMKOBJ - Defines the height, width, and depth of the object. Set the array DMKOBJ to the dimensions of the object, in drawing units, as follows:

DMKOBJ(1) - Height of the object.

DMKOBJ(2) - Width of the object.

DMKOBJ(3) - Depth of the object.

NOTE: There are restrictions to minimum dimensions of the object. (Minimum for any dimension is 1 inch.)

4.2.3.4.4 Screen Display, Obstacle Avoidance Definition, and Debugging Flag Status

CNTTBL - Defines the at which point where the center of the strength table will be plotted on the screen. Set the array CNTTBL to the screen coordinates of the desired plot center as follows:

CNTTBL (1) - X coordinate of the plot center point.

CNTTBL (2) - Y coordinate of the plot center point.

IVOIDO - A flag that is set to either ignore or consider interference with obstacles in the work place during the reach portion of Task Analysis functions. Set IVOIDO to the index of the desired setting.

0 - Perform reach ignoring obstacles in work place.

1 - Reach around obstacles in work place.

NUMDSP - Defines the display type used when drawing the man-model. Set NUMDSP to the index of the desired display type.

1 - WIRE FRAME
2 - SURFACED
3 - PROFILE

UNITS - Set UNITS to the number of units per inch. For example, if the drawing unit is in centimeters, set UNITS to 2.54; for millimeters, set UNITS to 25.4.

VUMAT(3,3) - A 3x3 array containing the view matrix which describes the orientation of the user's drawing with respect to the display screen. Available in most CAD systems, this array allows presentation of an uncluttered profile image of the man-model. (If the view matrix is not available in your system, set Display Type [NUMDSP=1] to wire frame and ignore the setting VUMAT.) CREW CHIEF assumes that the screen system positive axis points in the following directions: X points to the viewer's right, Y points up, and Z points toward the viewer. Set the array VUMAT as follows:

VUMAT(1-3,1) - The screen coordinates of the direction vector defining the positive X axis of the drawing coordinate system.

VUMAT(1-3,2) - The screen coordinates of the direction vector defining the positive Y axis of the drawing coordinate system.

VUMAT(1-3,3) - The screen coordinates of the direction vector defining the positive Z axis of the drawing coordinate system.

IBGPSH - A flag that defines whether or not control variables are to be printed. Set IBGPSH to the index desired.

0 - Do not print control variables.
1 - Print control variables.

Once the required input parameters are defined, the CREW CHIEF subroutine PSHUSR is called to display the man-model peɪforming the push in the work space. If the man-model can successfully perform the push, a strength table describing the push capabilities of Air Force maintenance technicians will be displayed. If the push cannot be performed, a message is displayed explaining the reason for the unsuccessful push.

4.2.3.5 CREW CHIEF Pull Function

The CREW CHIEF Pull function evaluates the man-model's ability to pull an object at a specified height above the man-model support platform. The program displays the man-model in the final pulling position and also displays a strength table. The following input parameters must be defined (see Appendix B):

ITASK - Defines which task is to be performed (INTEGER).

IWHAND - Defines hand(s) to perform the pull (INTEGER).

MOBILE - Defines the amount of motion the man-model is allowed (INTEGER).

IOBCLR - Flag to set the criticality of object clearance (INTEGER).

FRICTN - Defines Coefficient of Friction between boots and the support surface (REAL).

NUMHDL - Defines the number of handles on the object (INTEGER).

HDL1BG (3) - Defines one end point of the first handle on the object to be pulled (REAL).

HDL1EN (3) - Defines other end of the first handle (REAL).

HDL2BG (3) - Defines one end point of the second handle on the object to be pulled (REAL).

HDL2EN (3) - Defines other end of the second handle (REAL).

ISELVD - Selects the mode for defining the object distances (INTEGER).

DISOBJ - Defines horizontal distance from the man-model position reference point to the object (REAL).

HGTPLT - Defines vertical height from man-model support platform to the object (REAL).

ISELDI - Selects the mode for defining the object dimensions (INTEGER).

DMSOBJ(3) - Element identifiers of height, width, depth selected from the screen (REAL).

SELRST(50) - Element identifiers for additional lines selected from the screen to define the object's surfaces (50 maximum) (REAL).

ISELNO - Defines number of additional line segments selected for the object (REAL).

DMKOBJ(3) - Defines dimensions of the object (height x width x depth) keyed in (REAL).

CNTTBL (2) - Defines center of region to display strength table (REAL).

IVOIDO - Obstacle Avoidance Flag (INTEGER).

NUMDSP - Defines display type (INTEGER).

UNITS - Defines drawing units per inch (REAL).

VUMAT(3,3) - A 3x3 array containing the view matrix (REAL).

IBGPLL - Debug flag for the PULL function (INTEGER).

The first variable defines the task to be performed. The next four variables define the hand(s) to be used, the mobility to be allowed to the man-model, the obstacle clearance status, and the coefficient of friction between the boots and the surface.The next thirteen variables define the handles, horizontal and vertical distances, and the object dimensions. The last six variables define the screen display, and the obstacle avoidance and debugging flag status.

4.2.3.5.1 Task Identifier

ITASK - Identifies the task to perform. ITASK must be 11 for Pull.

4.2.3.5.2 Hand(s), Mobility, Object Clearance, and Friction Definition

IWHAND - Defines which hand(s) will be used to perform the pull. Set IWHAND to the required index.

3 - BOTH

NOTE: Both hands are always used when performing the pull.

MOBILE - Defines the amount of movement allowed the man-model. Set MOBILE to the index for the mobility desired in the analysis.

1 - Full body movement allowed

2 - Upper body movement (including arms and shoulders)

3 - Arms and shoulder movement only

4 - No movement allowed. No movement is allowed when the man-model has previously performed a task and the current task analysis is to be performed without changing the man-model's position

IOBCLR - Defines whether the object clearance is critical or non-critical. Object clearance is considered "critical" when it is desirable that 95% of the resultant pulling force is contained in the horizontal component. For example, pulling an electronic box into a rack. Set IOBCLR to the required index.

1 - CRITICAL
2 - NON-CRITICAL

FRICTN - Defines the coefficient of friction between the boot and the man-model support surface. Set FRICTN to an appropriate value between 0.1 and 1.0 for the surface on which the task is being performed.

4.2.3.5.3 Handle, Horizontal and Vertical Distances, and Object Dimension Definitions

NUMHDL - Defines the number of handles on the object. Set NUMHDL to the index corresponding to the number of handles on the object.

1 - ONE
2 - TWO

NOTE: At least one handle is required for pull tasks.

- **When NUMHDL = 1 (One handle) and ISELDI = 1 (Object dimensions selected from screen)**, the following arrays must be defined.

HDL1BG - One end of the line extending through the center of the handle of the object. The hand will be oriented with the thumb closest to this end of the handle. Define this point in drawing coordinates as follows:

HDL1BG(1) - X coordinate of the point.
HDL1BG(2) - Y coordinate of the point.
HDL1BG(3) - Z coordinate of the point.

HDL1EN - The other end of the line running through the center of the object's handle. Define this point in drawing coordinate as follows:

HLD1EN(1) - X coordinate of the point.
HDL1EN(2) - Y coordinate of the point.
HDL1EN(3) - Z coordinate of the point.

- **When NUMHDL = 2 (Two handles) and ISELDI = 1 (Object dimensions selected from screen)**, the following arrays must be defined as follows:

HDL1BG - One end of the line extending through the center of the first handle of the object. The right hand will be oriented with the thumb closest to this end of the handle. Define this point in drawing coordinates as follows:

HDL1BG(1) - X coordinate of the point.

HDL1BG(2) - Y coordinate of the point.

HDL1BG(3) - Z coordinate of the point.

HDL1EN - The other end of the line extending through the center cf the object's first handle. Define this point in drawing coordinates as follows:

HLD1EN(1) - X coordinate of the point.

HDL1EN(2) - Y coordinate of the point.

HDL1EN(3) - Z coordinate of the point.

HDL2BG - One end of the line running through the center of the second handle of the object. The left hand will be oriented with the thumb closest to this end of the handle. Define this point in drawing coordinates as follows:

HDL2BG(1) - X coordinate of the point.

HDL2BG(2) - Y coordinate of the point.

HDL2BG(3) - Z coordinate of the point.

HDL2EN - The other end of the line extending through the center of the second handle of the object. Define this point in drawing coordinates as follows:

HDL2EN(1) - X coordinate of the point.

HDL2EN(2) - Y coordinate of the point.

HDL2EN(3) - Z coordinate of the point.

ISELVD - Selects the mode for defining the horizontal and vertical distances from the man-model and the support platform to the object. Set the index to the desired mode.

1 - SELECT FROM SCREEN
2 - KEY IN

- **When ISELVD = 1 (Select from screen)**, a line is selected from the screen and the program internally computes and sets the horizontal distance from the man-model to the object

(DISOBJ), and the vertical distance from the support platform (HGTPLT).

- **When ISELVD = 2 (Key in)**, the following variables must be defined:

DISOBJ - Enter the horizontal distance, in drawing units, from the man-

model's position reference point to the object to be pulled.

HGTPLT - Set the vertical height from the man-model's support platform to the object, in drawing units.

ISELDI - Selects the mode for defining the object dimensions. Set ISELDI to the index for the desired mode.

1 - SELECT FROM SCREEN
2 - KEY IN

- **When ISELDI = 1 (Select from screen)**,the following array must be defined:

DMSOBJ - Defines the height, width, and depth of the object. Set the array DSMOBJ as follows:

DMSOBJ(1) - Identifier of a line selected from the screen to define the height of the object.

DMSOBJ(2) - Identifier of a line selected from the screen to define the width of the object.

DMSOBJ(3) - Identifier of a line selected from the screen to define the depth of the object.

For irregularly shaped objects, other elements besides height, width, and depth may be needed to define the object's surface. The array SELRST is set to the identifiers of the other elements selected to define the object dimensions. The element identifiers are decoded in the subroutine CCUEXF.

ISELNO - The number of other line segments selected to define the object dimensions.

- **When ISELDI = 2 (Key in)**, the following array is defined.

DMKOBJ - Defines the height, width, and depth of the object. Set the array DMKOBJ to the dimensions of the object, in drawing units, as follows:

DMKOBJ(1) - Height of the object.
DMKOBJ(2) - Width of the object.
DMKOBJ(3) - Depth of the object.

<u>NOTE:</u> There are restrictions to minimum dimensions of the object. (Minimum for any dimension is 1 inch.)

4.2.3.5.4 Screen Display, Obstacle Avoidance Definition, and Debugging Flag Status

CNTTBL - Defines the point where the center of the strength table will be plotted on the screen. Set the array CNTTBL to the screen coordinates of the desired plot center as follows:

CNTTBL (1) - X coordinate of the plot center point.

CNTTBL (2) - Y coordinate of the plot center point.

IVOIDO - A flag that is set to either ignore or consider interference with obstacles in the work place during the reach portion of Task Analysis functions. Set IVOIDO to the index for the desired setting.

0 - Perform reach, ignoring obstacles in work place.

1 - Reach around obstacles in work place.

NUMDSP - Defines the display type used when drawing the man-model. Set NUMDSP to the index corresponding to the desired display type.

1 - WIRE FRAME
2 - SURFACED
3 - PROFILE

UNITS - Set UNITS to the number of units per inch. For example, if the drawing unit is in centimeters, set UNITS to 2.54; for millimeters, set UNITS to 25.4.

VUMAT(3,3) - A 3x3 array containing the view matrix which describes the orientation of the user's drawing with respect to thє display screen. Available in most CAD systems, this array allows presentation of an uncluttered profile image of the man-model. (If the view matrix is not available in your system, set Display Type [NUMDSP=1] to wire frame and ignore the setting VUMAT.) CREW CHIEF assumes that the screen system positive axis points in the following directions: X points to the viewer's right, Y points up, and Z points toward the viewer. Set the array VUMAT as follows:

VUMAT(1-3,1) - The screen coordinates of the direction vector defining the positive X axis of the drawing coordinate system.

VUMAT(1-3,2) - The screen coordinates of the direction vector defining the positive Y axis of the drawing coordinate system.

VUMAT(1-3,3) - The screen coordinates of the direction vector defining the positive Z axis of the drawing coordinate system.

IBGPLL - A flag that defines whether or not control variables are to be printed. Set IBGPLL to the desired index.

0 - Do not print control variables.

1 - Print control variables.

Once the required input parameters are defined, the CREW CHIEF subroutine PLLUSR is called to display the man-model performing the pull in the work space. If the man-model can successfully perform the pull, a strength table describing the pull capabilities of Air Force maintenance technicians will be displayed.

4.2.3.6 CREW CHIEF Reach Function

The CREW CHIEF Reach function evaluates the ability of the maintenance technician to reach to a specified point, or points, in the work place. If the reach cannot be accomplished without interfering with drawing elements, the interference will be displayed. The following input variables must be defined (see Appendix B).

ITASK - Determines which task is to be performed (INTEGER).

MOBILE - Defines the amount of motion the man-model is allowed (INTEGER).

NUMGRP - Defines the extent of reach (INTEGER).

IWHAND - Defines hand(s) used to perform reach (INTEGER).

REACH1 (3) - Defines first point to be reached (REAL).

REACH2 (3) - Defines second point to be reached (REAL).

NUMDSP - Defines display type (INTEGER).

IVOIDO - Obstacle Avoidance Flag (INTEGER).

UNITS - Defines drawing units per inch (REAL).

VUMAT - A 3x3 array containing the view matrix (REAL).

IBGREC - Debug flag for the REACH function (INTEGER).

The first parameter defines the task. The next three parameters define the mobility of the man-model, the extent of reach, and the hand(s) used for the task. The next two parameters define the reach point(s), and the last five parameters define the screen display, obstacle avoidance definition, and debugging flag status.

4.2.3.6.1 Task Identifier

ITASK - To perform a reach task, ITASK must be set to 14.

4.2.3.6.2 Mobility, Extent of Reach, and Hand(s) Definition

MOBILE - Defines the amount of movement allowed the man-model. Set MOBILE to the index for the mobility desired in the analysis.

1 - Full body movement allowed. (Currently not available for Reach function.)

2 - Upper body movement (including arms and shoulders).

3 - Arm and shoulder movement only.

4 - No movement allowed. No movement is allowed when the man-model has previously performed a task and the current task analysis is to be performed without changing the man-model's position.

NUMGRP - Defines the extent of reach along the hand. The reach may be performed to the grip center, functional grip center, or finger tip of the hand. Set NUMGRP to the index for the desired extent of reach.

1 - GRIP CENTER
2 - FUNCTIONAL GRIP CENTER
3 - FINGER TIP

IWHAND - Defines hand(s) used to perform the reach. Set IWHAND to the index for the desired hand selection.

1 - RIGHT
2 - LEFT
3 - BOTH

4.2.3.6.3 Reach Point Definition

- **When IWHAND = 1 OR 2 (Right hand or Left hand)**, the following variable is defined.

REACH1 - Defines the drawing coordinates of the point toward which the man-model reaches. Set the array REACH1 to the 3-D point, in drawing coordinates, as follows:

REACH1 (1) - X coordinate of the point.
REACH1 (2) - Y coordinate of the point.
REACH1 (3) - Z coordinate of the point.

- **When IWHAND = 3 (Both hands)**, the following variables are defined.

REACH1 - Defines the drawing coordinates of the point toward which the man-model's right hand reaches. Set the array REACH1 to the 3-D point, in drawing coordinates, as follows:

REACH1 (1) - X coordinate of the point.
REACH1 (2) - Y coordinate of the point.
REACH1 (3) - Z coordinate of the point.

REACH2 - Defines the drawing coordinates of the reach point for the left hand. Set the array REACH2 to the 3-D point, in drawing coordinates, as follows:

REACH2 (1) - X coordinate of the point.
REACH2 (2) - Y coordinate of the point.
REACH2 (3) - Z coordinate of the point.

4.2.3.6.4 Screen Display, Obstacle Avoidance Definition, and Debugging Flag Status

NUMDSP - Defines the display type used when drawing the man-model. Set NUMDSP to the index corresponding to the desired display type.

1 - WIRE FRAME
2 - SURFACED
3 - PROFILE

IVOIDO - A flag that is set to either ignore or consider interference with obstacles in the work place during the reach portion of Task Analysis functions. Set IVOIDO to the index of the desired setting.

0 - Perform reach ignoring obstacles in work place.

1 - Reach around obstacles in work place.

UNITS - Set UNITS to the number of units per inch. For example, if the drawing unit is in centimeters, set UNITS to 2.54; for millimeters, set UNITS to 25.4.

VUMAT(3,3) - A 3x3 array containing the view matrix which describes the orientation of the user's drawing with respect to the display screen. Available in most CAD systems, this array allows presentation of an uncluttered profile image of the man-model. (If the view matrix is not available in your system, set Display Type [NUMDSP=1] to wire frame and ignore the setting VUMAT.) CREW CHIEF assumes that the screen system positive axis points in the following directions: X points to the viewer's right, Y points up, and Z points toward the viewer. Set the array VUMAT as follows:

VUMAT(1-3,1) - The screen coordinates of the direction vector defining the positive X axis of the drawing coordinate system.

VUMAT(1-3,2) - The screen coordinates of the direction vector defining the positive Y axis of the drawing coordinate system.

VUMAT(1-3,3) - The screen coordinates of the direction vector defining the positive Z axis of the drawing coordinate system.

IBGREC - A flag that defines whether or not control variables are to be printed. Set IBGREC to the desired index.

0 - Do not print control variables.

1 - Print control variables.

Once the required parameters are defined, the CREW CHIEF subroutine RECUSR is called to perform the reach. The man-model is displayed performing the reach, or reaching as close to the reach point as possible. If too many obstacles prevent the man-model from reaching, arrows are displayed indicating the points of interference.

4.3 INTERFERENCE ANALYSIS FUNCTION

The Interference Analysis function checks for interference between the man-model (including tools) and the elements in the user's drawing. Due to the time-intensive nature of the function, there are options to check only the appropriate portions of the man-model with drawing elements. The following variables are defined and passed through the common block ITFCTL.

INFLEV - Defines level of interference checking (INTEGER).

NUMDSP - Defines display type (INTEGER).

UNITS - Defines drawing units per inch (REAL).

VUMAT(3,3) - A 3 x 3 array containing the view matrix (REAL).

IBGITF - Debug flag for the Interference Analysis function (INTEGER).

4.3.1 Level of Interference Checking Definition

INFLEV - Defines whether the complete body or only portions of the body will be checked for interference with drawing elements. Set INFLEV to the index for the desired setting.

1 - Arms Only

- Checks interference of the upper and lower arms (including the hands and tool selected).

2 - Upper Body

- Checks interference of the body from the waist up (including those portions in the Arms Only selection).

3 - Full Body

- Checks interference of the whole body (including those portions in the Upper Body and Arms Only selections).

4.3.2 Screen Display Definition and Debugging Flag Status

NUMDSP - Defines the display type used when drawing the man-model. Set NUMDSP to the index of the desired display type.

1 - WIRE FRAME
2 - SURFACED
3 - PROFILE

UNITS - Set UNITS to the number of units per inch. For example, if the drawing unit is in centimeters, set UNITS to 2.54; for millimeters, set UNITS to 25.4.

VUMAT(3,3) - A 3x3 array containing the view matrix which describes the orientation of the user's drawing with respect to the display screen. Available in mcst CAD systems, this array allows presentation of an uncluttered profile image of the man-model. (If the view matrix is not available in your system, set Display Type [NUMDSP=1] to wire frame and ignore the setting VUMAT.) CREW CHIEF assumes that the screen system positive axis points in the following directions: X points to the viewer's right, Y points up, and Z points toward the viewer. Set the array VUMAT as follows:

VUMAT(1-3,1) - The screen coordinates of the direction vector defining the positive X axis of the drawing coordinate system.

VUMAT(1-3,2) - The screen coordinates of the direction vector defining the positive Y axis of the drawing coordinate system.

VUMAT(1-3,3) - The screen coordinates of the direction vector defining the positive Z axis of the drawing coordinate system.

IBGITF - A flag that defines whether or not control variables are to be printed. Set IBGITF to the index desired.

0 - Do not print control variables
1 - Print control variables

When the variables have been defined, the CREW CHIEF subroutine ITFUSR is called. The screen displays a message indicating the level of interference selected and whether or not interference was found. The man-model and arrows indicating the points of interference, if any, are displayed in the drawing.

4.4 WORK ENVELOPE ANALYSIS FUNCTION

The CREW CHIEF Work Envelope Analysis function allows the user to evaluate the volume of space available to operate a tool. This function can take into account the sweep of the tool, the volume required by the technician, and the reach of the technician. Results are displayed graphically superimposed on the user's drawing, and a quantitative result is displayed on the CAD system message line. The following variables are defined and passed through the common block WRKCTL.

IENVTY - Desired envelope type (INTEGER).

RCHINC - Reach increment (REAL).

MOBILE - Amount of mobility the man-model is allowed (INTEGER).

NUMDSP - Defines display type (INTEGER).

UNITS - Defines drawing units per inch (REAL).

VUMAT(3,3) - A 3 x 3 array containing the view matrix (REAL).

IBGWRK - Debug flag for the Work Envelope Analysis function (INTEGER).

4.4.1 Type of Envelope Definition

IENVTY - Defines the amount of constraint to be placed on the envelope calculations.

1 - Tool Sweep.

This calculates and displays a tool's full operational envelope. Interference between the tool and work place, or the technician and work place, as well as technician reach capability is ignored.

2 - Tool Envelope.

This calculates and displays a tool's operational envelope, taking into account interference between the tool and work place. Note that no technician limitations (reach and interference) are taken into account.

3 - Work Envelope.
This calculates and displays a tool's operational envelope, taking into account both tool/work place interference and technician limitations (reach and interference).

4.4.2 Reach Characteristics

RCHINC - Reach Increment (Set only when IEVNTY =3). CREW CHIEF performs a series of reaches to calculate the work envelope. This variable determines the increment used between successive tool placements, minimum value of increment is 0.5 inches (REAL).

MOBILE - Defines the amount of movement allowed the man-model. Set MOBILE to the index for the mobility desired in the analysis (Set only when IENVTY = 2 or 3).

1 - Full body movement allowed (currently not available for Reach function).

2 - Upper body movement (including arms and shoulders.

3 - Arm and shoulder movement only.

4 - No movement allowed. No movement is allowed when the man-model has previously performed a task and the current task analysis is to be performed without changing the man-model's position.

4.4.3 Screen Display Definition and Debugging Flag Status

NUMDSP - Defines the display type used when drawing the man-model. Set NUMDSP to the index of the desired display type.

1 - WIRE FRAME
2 - SURFACED
3 - PROFILE

UNITS - Set UNITS to the number of units per inch. For example, if the drawing unit is in centimeters, set UNITS to 2.54; for millimeters, set UNITS to 25.4.

VUMAT(3,3) - A 3x3 array containing the view matrix which describes the orientation of the user's drawing with respect to the display screen. Available in most CAD systems, this array allows presentation of an uncluttered profile image of the man-model. (If the view matrix is not available in your system, set Display Type [NUMDSP=1] to wire frame and ignore the setting VUMAT.) CREW CHIEF assumes that the screen system positive axis points in the following directions: X points to the viewer's right, Y points up, and Z points toward the viewer. Set the array VUMAT as follows:

VUMAT(1-3,1) - The screen coordinates of the direction vector defining the positive X axis of the drawing coordinate system.

VUMAT(1-3,2) - The screen coordinates of the direction vector defining the positive Y axis of the drawing coordinate system.

VUMAT(1-3,3) - The screen coordinates of the direction vector defining the positive Z axis of the drawing coordinate system.

IBGWRK - A flag that defines whether or not control variables are to be printed. Set IBGWRK to the index desired.

0 - Do not print control variables
1 - Print control variables

4.5 VISIBILITY ANALYSIS FUNCTION

The Visibility Analysis function provides a plot of the visual acuity of the man-model in four different conditions: no external obstruction, and three conditions with external obstruction (chemical defense mask, cold weather parka with the hood "open," and cold weather parka with the hood "closed"). Elements of the man-model, such as the hands and arms, which interfere with visibility may be included or excluded in the plot. The plot is presented on the screen in an area selected by the user. The plot portrays the limits of visual acuity and the azimuth and elevation angles. The following variables are defined and passed through the common block VISCTL.

LOSTYP - Defines the line-of-sight type (INTEGER).

EYELOC(3) - Defines the eye location in drawing coordinates (REAL).

TRGLOC(3) - Defines the target loc1tion in drawing coordinates (REAL).

MANPLT - Indicator for including or excluding the man-model in the plot (INTEGER).

CNTTBL(2) - Defines center of visibility plot area (REAL).

UNITS - Defines drawing units per inch (REAL).

IBGVIS - Debug flag for the Visibility Analysis function (INTEGER).

4.5.1 Line-of-Sight, Eye and Target Location Definition

LOSTYP - Defines the line-of-sight type to be used. Select the index for the desired type.

1 - Use current man-model line-of-sight (the line-of-sight from the existing head orientation and task location).

2 - User defined line-of-sight (the line-of-sight for an eye and target location other than the current man-model line-of-sight).

- **When LOSTYP = 1 (Current Man-model Line-of-**

MANPLT - Defines whether the man-model is included or excluded from the Visibility Analysis plot. Select the index for the desired plotting status.

0 - Exclude the man-model from the plot.
1 - Include the man-model in the plot.

- **When LOSTYP = 2 (User Defined Line-of-Sight)**, the following variables must be defined.

EYELOC(3) - Defines the eye location in drawing coordinates. Set the array EYELOC in drawing coordinates as follows:

EYELOC(1) - X coordinate of the eye location.

EYELOC(2) - Y coordinate of the eye location.

EYELOC(3) - Z coordinate of the eye location.

TRGLOC(3) - Defines the target location in drawing coordinates. Set the array TRGLOC, in drawing coordinates, as follows:

TRGLOC(1) - X coordinate of the target location.

TRGLOC(2) - Y coordinate of the target location.

TRGLOC(3) - Z coordinate of the target location.

4.5.2 Screen Display Definition and Debugging Flag Status

CNTTBL(2) - Defines the point where the center of the plot will appear on the screen. Set the array CNTTBL to the screen coordinates of the desired plot center as follows:

CNTTBL(1) - X coordinate of the plot center point.

CNTTBL(2) - Y coordinate of the plot center point.

UNITS - Set UNITS to the number of units per inch. For example, if the drawing unit is in centimeters, set UNITS to 2.54; for millimeters, set UNITS to 25.4.

IBGVIS - A flag that defines whether or not control variables are to be printed. Set IBGVIS to the desired index.

0 - Do not print control variables
1 - Print control variables

When all variables have been defined, the CREW CHIEF subroutine VISUSR is called. The Visibility Analysis plot is displayed in the selected area of the screen.

4.6 CURRENT CONFIGURATION FUNCTION

The Current Configuration function displays the man-model parameters defined in the Initialization function, and the hand and tool parameters defined in the Task Analysis function. The configuration definitions are displayed in an area of the screen designated by the user. The three input variables are defined as follows and passed through the common block CFGCTL.

Variable Definition

XCEN - Defines X coordinate of the display center in screen coordinates.

YCEN - Defines Y coordinate of the display center in screen coordinates.

IBGCFG - A flag that defines whether or not control variables are to be printed. Set IBGCFG to the desired index.

0 - Do not print control variables
1 - Print control variables

When all variables have been defined, the CREW CHIEF subroutine CFGUSR is called to display the Current Configuration definitions in the selected display area.

SECTION 5
CAD DATABASE OUTPUT

Geometry output is used for adding geometric elements to the user's drawing. The CREW CHIEF functions output geometry through subroutine calls. The distribution tape contains a sample listing of the geometry output routines. These subroutines, called by the CREW CHIEF functions, must be modified to invoke the corresponding geometric routines that are recognized by the user's CAD system. For most CAD systems, only one line of the sample listing need be changed.

5.1 SUBROUTINE CCO3DL: GENERATE A 3-D LINE SEGMENT

When the CREW CHIEF functions need to generate 3-D lines in the user's drawing, the subroutine CCO3DL is called. For example, CREW CHIEF displays the man-model as a series of 3-D lines, output through a single call to CCO3DL. The parameters passed to CCO3DL are:

ISETNO - Geometry Attribute (Table 5.1).

POINT1 (3,2000) - The 3-D coordinates of one end of the line segments,

POINT2 (3,2000) - The 3-D coordinates of the other end of the line segments, and

NUMLIN - Number of line segments.

ICHCK - Flag variable set in CCO3DL.

0 - If line segments were generated

1 - If, for any reason, the line segment could not be generated

TABLE 5.1: GEOMETRY ATTRIBUTES USED IN CREW CHIEF

Segment #	Description
0	General Work Place Geometry
1	Hips
2	Trunk
3	Right Upper Arm
4	Right Lower Arm
5	Right Hand
6	Left Upper Arm
7	Left Lower Arm
8	Left Hand
9	Right Upper Leg
10	Right Lower Leg
11	Left Upper Leg
12	Left Lower Leg
13	Head
14	Right Boot
15	Left Boot
16	Right Main Tool
17	Right Extension
18	Left Main Tool
19	Left Extension
20	Right Socket
21	Left Socket
22	Misc. Geometry
24	CREW CHIEF Table Output
25	CREW CHIEF Geometry Output

NOTE: POINT1 and POINT2 are arrays where the first, second, and third locations of the arrays correspond to the X, Y, and Z coordinates of 3-D points. The flag variable ICHCK should be initially set to "0"; if an error occurs when a line segment is generated, set ICHCK to "1."

The subroutine call "LINE3D" in the distribution tape sample listings must be changed to call the 3-D line-generating sub-routine recognizable by the user's CAD system.

5.2 SUBROUTINE CCO3DA: GENERATE A 3-D ARROW

When the CREW CHIEF functions need to generate a 3-D arrow in the user's drawing, the subroutine CCO3DA is called. For example, if the man-model's arm incurs interference with a line segment in the user's drawing, a 3-D arrow will be displayed to indicate the point of intersection. The parameters passed to CCO3DA are:

ISETNO - Geometry Attribute (TABLE 5.1).

POINT1(3) - Head point of arrow in drawing coordinates.

POINT2(3) - Tail point of arrow in drawing coordinates.

ICHCK - Check digit for successful generation.

The subroutine call "AROW3D" in the sample listing in the distribution tape must be changed to call a 3-D arrow-generating subroutine recognized by the user's CAD system.

5.3 SUBROUTINE CCO2DS: GENERATE 2-D SPLINE

When the CREW CHIEF functions need to generate a 2-D spline in the user's drawing, the subroutine CCO2DS is called. For example, when displaying a strength table during the execution of the Lift function, several of the borders of the table are displayed using 2-D splines. The parameters passed to CCO2DS are:

ISETNO - Geometry Attribute (TABLE 5.1)

NUMPTS - Number of points in the spline.

POINTS

(2,NUMPTS) - The X and Y screen coordinates of each key point in the spline.

ICHCK - Check digit for successful generation.

The subroutine call "CADSP1," in the sample listing in the distribution tape must be changed to call a 2-D spline retrieval subroutine recognized by the user's CAD SYSTEM.

5.4 SUBROUTINE CCO2DL: GENERATE A 2-D LINE

The subroutine CCO2DL is called to generate 2-D line segments in the user's drawing for CREW CHIEF. For example, 2-D lines are also used to create strength tables for the Task Analysis functions. The parameters passed to CCO2DL are:

ISETNO - Geometry Attribute (TABLE 5.1

POINT1(2) - X and Y screen coordinates of the first end point.

POINT2(2) - X and Y screen coordinates of the second end point.

ICHCK - Check digit for successful generation.

The subroutine call "CADLN," in the sample listing in the distribution tape must also be changed to a 2-D line generating subroutine call that is recognized by the user's CAD system.

5.5 SUBROUTINE CCOTXT: GENERATE A LINE OF TEXT ON THE SCREEN

The subroutine CCOTXT is called to generate on-screen text for CREW CHIEF functions. Parameters passed to CCOTXT are:

ISETNO - Geometry Attribute (TABLE 5.1).

ITEXT - Array containing text.

LENTXT - Defines number of characters contained in the text.

X - Defines X screen coordinate of start of text.

Y - Defines Y screen coordinate of start of text.

SIZE - Defines character size in drawing units.

IOPT - Determines direction of text:

If 0, the text is horizontal.

If 90, the text is vertical.

No other values recognized.

ICHCK - Error flag:

If 0, text generated without errors.

If 1, error in text generation.

The subroutine call "CADTXT," in the sample listing in the distribution tape must also be changed to a text-generating subroutine call that is recognized by the user's CAD system.

5.6 SUBROUTINE CCOMML: MESSAGE SUBROUTINE

When the CREW CHIEF functions need to return an informational message to the user at the graphic scope, the subroutine CCOMML is called. The parameters associated with this call are:

NCHAR - Number of characters in the message.

MSG1(21) - Text of the message.

The subroutine call "MSGGEN" must be replaced with a message subroutine that the user's CAD system will recognize. For most CAD systems, this is the only line of the sample listing that will need to be changed. The maximum number of characters in the message is 60.

5.7 SUBROUTINE CCOMPT: GENERATE MESH POINTS

When the user selects a "Surface" display type, the CREW CHIEF model is output as a Finite Element Mesh. CCOMPT allows the points defining this mesh to be output to the user's CAD drawing, for later use in defining these mesh elements (see subroutine CCOMEL). The parameters passed to CCOMPT are:

ISETNO - Geometry attribute to be assigned to this entity. See TABLE 5.1 for more information on these attributes.

NUMPTS - Number of mesh points to be added with this call.

PTSMSH - The x-, y-, and z-coordinates of the mesh points to be added.

ICHK - Error flag.

= 0, if points generated without errors.

= 1, if there was an error in point generation.

5.8 SUBROUTINE CCOMEL: GENERATE MESH ELEMENTS.

When the user selects a "Surface" display type, the CREW CHIEF model is output as a Finite Element Mesh. CCOMEL defines the mesh connectivity of the points defined through a previous call to CCOMPT. The parameters passed to CCOMEL are:

ISETNO - Geometry attribute to be assigned to this entity. See TABLE 5.1 for more information on these attributes.

ISEL - Type of elements to be added.

= 1, add mesh triangles.
= 2, add mesh quadrilaterals.

NUMELN - Number of elements to be added.

MESHTR - This is a 4xNUMELN array containing triangle connectivity data. The first three locations for each element contain the mesh point numbers defining the element, while the fourth location contains a color indicator.

MESHQD - This is a 5xNUMELN array containing quadrilateral connectivity data. The first four locations for each element contain the mesh point numbers defining the element, while the fifth location contains a color indicator.

ICHK - Error flag.

= 0, if points generated without errors.
= 1, if there was an error in point generation.

SECTION 6
UTILITY SUBROUTINES

Utility subroutines are used for transferring information between CREW CHIEF and the user's CAD system. These subroutines are called only once in the execution of the CREW CHIEF function, and are common to all CREW CHIEF functions. The tape file "CREW.CHIEF.CADSI.SAMPLE.SOURCE" contains a sample listing of each utility subroutine. These subroutines must be modified to reflect calls of the user's CAD system. For most CAD systems, only one line of the sample listing need be changed.

6.1 SUBROUTINE CCUERS: ERASE CURRENT MAN-MODEL DISPLAY

Before displaying the man-model in the user's drawing, the CREW CHIEF program deletes any previous man-model display by calling the subroutine CCUERS. For example, when the Initialization function is executed, a man-model is displayed. Before the Head Orientation function can redisplay the man-model, it must first erase the man-model displayed during Initialization. The parameters passed to CCUERS are:

IOPT - Processing option

= 1 - Erase all CREW CHIEF geometry

= 2 - Erase only CREW CHIEF geometry with the attribute specified in a PARM (1)

PARM(1) - Attribute of elements to be erased

(IOPT = 2, only)

6.2 SUBROUTINE CCUEXF: CREW CHIEF GEOMETRY ELEMENT TRANSFER

One of the defining characteristics of the CREW CHIEF system of programs is its ability to interact with the user's drawings. The programs can calculate interference in the work place, alternative positions which avoid interference, or simply display the drawing as the technician would see it. To perform these functions it is necessary to transport descriptions of various drawing elements (ellipses, circles, splines, etc.) to the CREW CHIEF programs. The subroutine CCUEXF controls the transfer of elements from the user's CAD drawing to the CREW CHIEF programs. The transfer of elements is initiated when the CREW CHIEF programs call CCUEXF.

The data base format and data retrieval techniques are extremely dependent upon the resident CAD system, and can be quite complex. To allow programmers of resident systems full use of their system capabilities, CREW CHIEF transfers execution control, through CCUEXF, and uses a set of geometry-receiving subroutines to input work place geometry to the CREW CHIEF functions. Elements are passed to CREW CHIEF through subroutines called by the resident system, and control does not fully return to CREW CHIEF until the last geometric element has been passed. CREW CHIEF will call the utility subroutine CCUEXF with two parameters, ITYPE and PARM.

ITYPE - Indicates type of elements to retrieve.

1 - Retrieve all drawing elements
2 - Retrieve a specific type of drawing element
3 - Retrieve a particular drawing element
4 - Retrieve CREW CHIEF elements with
5 - Retrieve a specific CREW CHIEF element

PARM (20) - Additional parameter list.

If ITYPE = 1, all values in PARM are ignored.

If ITYPE = 2, PARM(1) contains the element type (lines, circles, etc.).

If ITYPE = 3 or 5, PARM(1) contains the element identifier number.

If ITYPE = 4, PARM(1) contains the desired attribute.

NOTE: PARM(2) through PARM(20) are reserved for future use.

6.3 SUBROUTINE CCULTP: CHANGE LINE TYPE

The CREW CHIEF programs change the line type from time to time to improve clarity and readability of screen displays and plots. The line type is designated by changing the variable ITYPE. The following values designate the line types indicated.

ITYPE = 0 - Medium solid line
1 - Light solid line
2 - Heavy solid line
3 - Heavy dashed line
4 - Light dashed line

The line type should be set to "medium solid line" (ITYPE = 0) at the start of each CREW CHIEF function. All elements will be passed as "medium solid lines" until the function is completed, or until CCULTP is called and the variable (ITYPE) is reset. Elements are always passed in the line type designated by the ITYPE value input at the last call of CCULTP in the CREW CHIEF function.

6.4 SUBROUTINE CCUCOL: CHANGE DISPLAY COLOR

The CREW CHIEF display is comprised of line segments and text. For CAD systems which support color, this display can be colored with 12 unique colors. CREW CHIEF operates on the assumption that once a color type is set, all elements subsequently generated will be of this color until the next call for change of color. CREW CHIEF colors are numbered 0 through 12, and the utility subroutine CCUCOL must translate these numbers to the correct CAD colors according to the following table:

0 - NO COLOR, RESET COLORS

1 - OLIVE DRAB

2 - PINK/FLESH TONES

3 - YELLOW/BLOND

4 - BROWN/BOOTS AND WOODEN TOOL HANDLES

5 - GRAY/STEEL ON TOOLS

6 - WHITE/FUR ON PARKA

7 - OFF WHITE/USED ON CHEMICAL DEFENSE MASK

8 - DARKER FLESH TONES

9 - PURPLE/USED FOR INTERFERENCE ARROWS

10 - CURRENTLY NOT USED

11 - CURRENTLY NOT USED

12 - CURRENTLY NOT USED

CCUCOL contains a single parameter, ICOLOR, which is set to the appropriate color.

SECTION 7
CAD DATABASE INPUT

Once the utility subroutine CCUEXF retrieves a particular drawing element, this drawing element is transferred to the CREW CHIEF programs through a call to one of several geometry-receiving subroutines. Each subroutine described in this section receives a particular type of drawing element. Thus, several subroutine calls are required to transfer an entire drawing. All parameters are passed in drawing coordinates.

7.1 SUBROUTINE CCI3DL: INPUT 3-D LINE

To transfer a 3-D line to the CREW CHIEF programs, the subroutine CCI3DL must be called. The calling format for CCI3DL is:

```
CALL CCI3DL(ISETNO,PTS), where
```

ISETNO - is the line segment set number.

PTS(1-3) - defines coordinates of one end point of the line segment.

PTX(4-6) - defines coordinates of the other end point of the line segment.

7.2 SUBROUTINE CCI3DC: INPUT 3-D CIRCLE

To transfer a 3-D circle, or ellipse, to the CREW CHIEF programs, the subroutine CCI3DC must be called. The calling format CCI3DC is:

```
CALL CCI3DC(CENTER,V1,V2,ANGLE1,ANGLE2,AMAJ,AMIN) where
```

CENTER(3) - Defines coordinates of the center of the circle or ellipse.

V1 (3) - Defines direction vector indicating the direction of the major axis of the ellipse.

V2 (3) - Defines direction vector indicating the direction of the minor axis of the ellipse.

ANGLE1 - Defines beginning angle of the arc in radians (measured with a positive angle in the direction of V1 from V2).

ANGLE2 - Defines ending angle of the arc in radians (the portion of the ellipse beginning at ANGLE1 and ending at ANGLE2).

AMAJ - Defines major axis length.

AMIN - Defines minor axis length.

NOTE: If passing a circle, then AMAJ = AMIN = RADIUS.

7.3 SUBROUTINE CCI3DS: INPUT 3-D SPLINE

The subroutine CCI3DS must be called to pass a 3-D spline to the CREW CHIEF programs. The calling format is:

```
CALL CCI3DS(NOPTS,XYZ,ABC1,ABC2)  where
```

NOPTS - Defines the number of points in the spline.

XYZ (3,2000) - Defines the key points of the spline.

ABC1 (3,2000)- Defines the direction vectors at the beginning of each bay.

ABC2 (3,2000)- Defines the direction vectors at the end of each bay.

7.4 SUBROUTINE CCIHBP: INPUT 3-D HOMOGENEOUS BEZIER PATCH

To transfer a 3-D homogeneous Bezier Patch to the CREW CHIEF programs, the subroutine CCIHBP must be called. The calling format is:

CALL CCIHBP(XYZW) where

XYZW (4,16) - Defines the 16 defining points of the Bezier Patch. The fourth value for each point gives the weighting factor (non-zero) for that point. Note also that the first, fourth, thirteenth, and sixteenth points lie on the corners of the patch.

7.5 SUBROUTINE CCIRUL: INPUT RULED SURFACE

Call subroutine CCIRUL to transfer a 3-D ruled surface to the CREW CHIEF programs. Ruled surfaces are generated by connecting two splines with straight lines. The calling format is:

CALL CCIRUL(NOPTS,XYZ,ABC1,ABC2,NOPTS2) where

NOPTS - Defines the number of points in the first spline.

XYZ (3,400) - Defines the key points of the splines. [Points for the second spline begin at XYZ(1,NOPTS+1).]

ABC1 (3,400) - Defines the direction vectors at the beginning of each bay. [Vectors for the second spline begin at ABC1(1,NOPTS+1).]

ABC2 (3,400) - Defines the direction vectors at the end of each bay. [Vectors for the second spline begin at ABC2(1,NOPTS+1).]

NOPTS2 - Defines the number of points in the second spline.

7.6 SUBROUTINE CCIREV: INPUT SURFACE OF REVOLUTION

To transfer a 3-D surface of revolution to the CREW CHIEF programs, you must call subroutine CCIREV. The calling format is:

```
CALL CCIREV(NOPTS,XYZ,ABC1,ABC2,VUMAT,ANGRAD) where
```

NOPTS - Defines the number of points in the spline.

XYZ (3,200) - Defines the key points of the spline.

ABC1 (3,200)- Defines the direction vectors at the beginning of each bay.

ABC2 (3,200)- Defines the direction vectors at the end of each bay.

VUMAT (3,4) - Defines the rotation and origin of a coordinate system. The first three columns of this array give the direction of the X, Y, and Z axes, respectively, of a rotation coordinate system. The last column gives the origin of this system.

ANGRAD - Defines the angle of rotation,about the Z axis of the system defined above, which sweeps out the surface of revolution.

APPENDIX A

TOOLS AND ACCESSSORIES USED IN CREW CHIEF

TABLE A.1

TOOL CLASSES AND MODELS

CLASS	NAME	MODEL	NAME
(KLASID)		(MODID)	
1	WRENCHES	13	TORQUE WRENCH
		14	RATCHET WRENCH
		15	BREAKER BAR
		16	STANDARD BOX END WRENCH
		17	DEEP OFFSET BOX END WRENCH
		18	RATCHETING BOX END WRENCH
		19	OPEN END WRENCH
		20	COMBINATION WRENCH OPEN END
		21	COMBINATION WRENCH BOX END
		22	SPEED HANDLE
		25	ALLEN WRENCH SHORT END
		26	ALLEN WRENCH LONG END
2	SCREWDRIVERS	12	OFFSET SCREWDRIVER
		23	REGULAR SCREWDRIVER
3	PLIERS	7	COMBINATION
		8	NEEDLE NOSE
		9	SAFETY WIRE
		10	WIRE CUTTERS
		11	ADJUSTABLE JOINT
4	MISCELLANEOUS TOOLS	1	HAMMER
		2	METAL FILE
		3	SCRAPER
		4	HACKSAW
		5	DRILL
		6	SANDER
		24	NUT DRIVER
8	USER DEFINED TOOLS		

TABLE A.2

ACCESSORY CLASSES AND MODELS

CLASS (MACCLS)	NAME	MODEL (MACMID)	NAME
5	SOCKET	34	REGULAR
		35	DEEP WELL
		36	HEX HEAD
		37	UNIVERSAL
6	EXTENSION	27	2 INCH
		28	3 INCH
		29	5 INCH
		30	6 INCH
		31	8 INCH
		32	10 INCH
		33	12 INCH
7	CHISEL	38	CHISEL

TABLE A.3

TOOL VARIABLE INTERACTIONS

MODEL	NAME	HAND	GRIP	HANDLE	STRENGTH
				DIRECTION	DATA
(MODID)		(IWHAND)	(IDRTOL)	REQUIRED[1]	AVAILABLE[2]
1	HAMMER	1,2	1	YES	NO
2	METAL FILE	1	1	NO	NO
3	SCRAPER	1,2	1	YES	NO
4	HACKSAW	1,2	1	YES	NO
5	DRILL	1,2	1	YES	NO
6	SANDER	3	1	YES	NO
7	COMBINATION PLIERS	1,2	1,2	YES	NO
8	NEEDLE NOSE PLIERS	1,2	1,2	YES	NO
9	SAFETY WIRE PLIERS	1,2	1,2	YES	NO
10	WIRE CUTTERS	1,2	1,2	YES	NO
11	ADJUSTIBLE JOINT PLIERS	1,2	1,2	YES	NO
12	OFFFSET SCREWDRIVER	1,2	1,2	YES	NO
13	TORQUE WRENCH	1-3	1,2	YES	YES
14	RATCHET WRENCH	1-3	1,2	YES	YES
15	BREAKER BAR	1-3	1,2	YES	YES
16	STANDARD BOX END WRENCH	1-3[3]	1,2	YES	YES
17	DEEP OFFSET BOX END WRENCH	1-3[3]	1,2	YES	YES
18	RATCHETING BOX END WRENCH	1-3[3]	1,2	YES	YES
19	OPEN END WRENCH	1-3[3]	1,2	YES	YES
20	COMBINATION WRENCH OPEN END	1-3[3]	1,2	YES	YES

TABLE A.3 (Continued)

TOOL VARIABLE INTERACTIONS

				HANDLE	STRENGTH
MODEL	NAME	HAND	GRIP	DIRECTION	DATA
(MODID)		(IWHAND)	(IDRTOL)	REQUIRED[1]	AVAILABLE[2]
21	COMBINATION WRENCH BOX END	1-3[3]	1,2	YES	YES
22	SPEED HANDLE	3	1,2	YES	NO
23	REGULAR SCREWDRIVER	1,2	1-3	NO	NO
24	NUT DRIVER	1,2	1-3	NO	NO
25	ALLEN WRENCH SHORT END	1,2	1,2	YES	NO
26	ALLEN WRENCH LONG END	1,2	1	YES	NO

[1] When the value in this column is NO, the variable TOLDIR does not need to be set.

[2] YES in this column indicates that strength data is available, and the array CNTTBL must be set.

[3] Sizes of these wrenches, used with numbered screws, only hands 1 and 2 (right and left) can be selected.

TABLE A.4

TOCL MODEL SIZES AND SUB-SIZES

MODEL (MODID)	NAME	TOOL SIZE* (MSZID)	SIZE DESCRIPTION	SUB-SIZE* (MSSZID)	SUB-SIZE DESCRIPTION
1	HAMMER	1	-	1	-
2	METAL FILE	1	-	1	-
3	SCRAPER	1	-	1	-
4	HACKSAW	1	-	1	-
5	DRILL	1	-	1	-
6	SANDER	1	-	1	-
7	COMBINATION PLIERS	1	-	1	-
8	NEEDLE NOSE PLIERS	1	-	1	-
9	SAFETY WIRE PLIERS	1	-	1	-
10	WIRE CUTTERS	1	-	1	-
11	ADJUSTABLE JOINT PLIERS	1	-	1	-
12	OFFSET SCREWDRIVER	1	-	1	REGULAR HANDLE
				2	LONG HANDLE

* Where only one size or sub-size is listed there is no user selection, however, the variable(s) (MSZID) or (MSSXID) must be set to 1.

TABLE A.4 (Continued)

TOOL MODEL SIZES AND SUB-SIZES

MODEL (MODID)	NAME	TOOL SIZE (MSZID)	SIZE DESCRIPTION	SUB-SIZE (MSSZID)	SUB-SIZE DESCRIPTION
13	TORQUE WRENCH	1	#1 - 10 SCREWS	1	REGULAR HANDLE
				2	LONG HANDLE
		2	3/16" TO 1/2"	1	REGULAR HANDLE
				2	LONG HANDLE
		3	9/16" TO 1"	1	REGULAR HANDLE
				2	LONG HANDLE
14	RATCHET WRENCH	1	#1 - 10 SCREWS	1	REGULAR HANDLE
				2	LONG HANDLE
		2	3/16" TO 1/2"	1	REGULAR HANDLE
				2	LONG HANDLE
		3	9/16" TO 1"	1	REGULAR HANDLE
				2	LONG HANDLE
15	BREAKER BAR	1	#1 · 10 SCREWS	1	REGULAR HANDLE
				2	LONG HANDLE
		2	3/16" TO 1/2"	1	REGULAR HANDLE
				2	LONG HANDLE
		3	9/16" TO 1"	1	REGULAR HANDLE
				2	LONG HANDLE

TABLE A.4 (Continued)

TOOL MODEL SIZES AND SUB-SIZES

MODEL	NAME	TOOL SIZE	SIZE DESCRIPTION		SUB-SIZE*
(MODID)		(MSZID)	(fastener diameter)	(head size)	(MSSZID)
16	STANDARD BOX END WRENCH	1	#3, 4 & 5 SCREWS	3/16"	1
		2	#6 & 8 SCREWS	1/4"	1
		3	#10 SCREWS	5/16"	1
		4	3/16"	3/8"	1
		5	1/4"	7/16"	1
		6	5/16"	1/2"	1
		7	3/8"	9/16"	1
		8	7/16"	5/8"	1
		9	1/2"	3/4"	1
		10	9/16"	3/16"	1
		11	5/8"	15/16"	1
		12	3/4"	1 1/4"	1
		13	7/8"	1 5/16"	1
		14	1"	1 1/2"	1

* **There is no user selection of sub-size for the Standard Box End Wrench, however, the variable (MSSZID) must be set to 1.**

TABLE A.4 (Continued)

TOOL MODEL SIZES AND SUB-SIZES

MODEL	NAME	TOOL SIZE	SIZE DESCRIPTION		SUB-SIZE*
(MODID)		(MSZID)	(fastener diameter)	(head size)	(MSSZID)
17	DEEP OFFSET BOX END WRENCH	1	#6 & 8 SCREWS	1/4"	1
		2	#10 SCREWS	5/16"	1
		3	3/16"	3/8"	1
		4	1/4"	7/16"	1
		5	5/16"	1/2"	1
		6	3/8"	9/16"	1
		7	5/8"	15/16"	1
		8	1/2"	3/4"	1
		9	9/16"	13/16"	1
		10	5/8"	15/16"	1
		11	3.4"	1 1/4"	1

* **There is no user selction of sub-size for the Deep Offset Box End Wrench, however, the variable (MSSZID) must be set to 1.**

TABLE A.4 (Continued)

TOOL MODEL SIZES AND SUB-SIZES

			SIZE		
MODEL	NAME	TOOL SIZE	DESCRIPTION		SUB-SIZE*
(MODID)		(MSZID)	(fastener diameter)	(head size)	(MSSZID)
18	RATCHETING BOX END WRENCH	1	#6 & 8 SCREWS	1/4"	1
		2	#10 SCREWS	5/16"	1
		3	3/16"	3/8"	1
		4	1/4"	7/16"	1
		5	5/16"	1/2"	1
		6	3/8"	9/16"	1
		7	7/16"	5/8"	1
		8	1/2"	3/4"	1
		9	9/16"	13/16"	1
		10	5/8"	15/16"	1
		11	3/4"	1 1/4"	1

* There is no user selection of sub-size for the Ratcheting Box End Wrench, however, the variable (MSSZID) must be set to 1.

TABLE A.4 (Continued)

TOOL MODEL SIZES AND SUB-SIZES

MODEL	NAME	TOOL SIZE	SIZE DESCRIPTION		SUB-SIZE*
(MODID)		(MSZID)	(fastener diameter)	(head size)	(MSSZID)
19	OPEN END WRENCH	1	#3, 4 & 5 SCREWS	3/16"	1
		2	#6 & 8 SCREWS	1/4"	1
		3	#10 SCREWS	5/16"	1
		4	3/16"	3/8"	1
		5	1/4"	7/16"	1
		6	5 16"	1/2"	1
		7	3/8"	9/16"	1
		8	7/16"	5/8"	1
		9	1/2"	3/4"	1
		10	9/16"	13/16"	1
		11	5/8"	15/16"	1
		12	3/4"	1 1/4"	1

* There is no user selection of sub-size for the Open End Wrench, however, the variable (MSSZID) must be set to 1.

TABLE A.4 (Continued)

TOOL MODEL SIZES AND SUB-SIZES

MODEL	NAME	TOOL SIZE	SIZE DESCRIPTION		SUB-SIZE*
(MODID)		(MSZID)	(fastener diameter)	(head size)	(MSSZID)
20	COMBINATION WRENCH, OPEN END	1	#6 & 8 SCREWS	1/4"	1
		2	#10 SCREWS	3/16"	1
		3	3/16"	3/8"	1
		4	1/4"	7/16"	1
		5	5/16"	1/2"	1
		6	3/8"	9/16"	1
		7	7/16"	5/8"	1
		8	1/2"	3/4"	1
		9	9/16"	13/16"	1
		10	5/8"	15/16"	1
		11	3/4"	1 1/4"	1

* There is no user selection of sub-size for the Combination Wrench, Open End, however, the variable (MSSZID) must be set to 1.

TABLE A.4 (Continued)

TOOL MODEL SIZES AND SUB-SIZES

			SIZE		
MODEL	NAME	TOOL SIZE	DESCRIPTION		SUB-SIZE*
(MODID)		(MSZID)	(fastener diameter)	(head size)	(MSSZID)
21	COMBINATION WRENCH BOX END	1	#6 & 8 SCREWS	1/4"	1
		2	#10 SCREWS	3/16"	1
		3	3/16"	3/8"	1
		4	1/4"	7/16"	1
		5	5/16"	1/2"	1
		6	3/8"	9/16"	1
		7	7/16"	5/8"	1
		8	1/2"	3/4"	1
		9	9/16"	13/16"	1
		10	5/8"	15/16"	1
		11	3/4"	1 1/4"	1
22	SPEED HANDLE	1	#1 - 10 SCREWS	-	1
		2	3/16" TO 1/2"	-	1
		3	9/16" TO 1"	-	1

* **There is no user selection of sub-size for the Combination Wrench, Box End, and the Speed Handle, however, the variable (MSSZID) must be set to 1.**

TABLE A.4 (Continued)

TOOL MODEL SIZES AND SUB-SIZES

			SIZE	
MODEL	NAME	TOOL SIZE	DESCRIPTION	SUB-SIZE*
(MODID)		(MSZID)	(blade length)	(MSSZID)
23	REGULAR SCREWDRIVER	1	1.5"	1
		2	3.0"	1
		3	4.0"	1
		4	6.0"	1
		5	8.0"	1
		6	10.0"	1
		7	12.0"	1

* **There is no user selection of sub-size for the Regular Screwdriver, however, the variable (MSSZID) must be set to 1.**

TABLE A.4 (Continued)

TOOL MODEL SIZES AND SUB-SIZES

MODEL	NAME	TOOL SIZE	SIZE DESCRIPTION		SUB-SIZE*
(MODID)		(MSZID)	(fastener diameter)	(head size)	(MSSZID)
24	NUT DRIVER	1	#1 & 2 SCREWS	1/8"	1
		2	#3, 4 & 5 SCREWS	3/16"	1
		3	#6 & 8 SCREWS	1/4"	1
		4	#10 SCREWS	5/16"	1
		5	3/16"	3/8"	1
		6	1/4"	7/16"	1
		7	5/16"	1/2"	1
		8	3/8"	9/16"	1
		9	7/16"	5/8"	1

* There is no user selection of sub-size for the Nut Driver, however, the variable (MSSZID) must be set to 1.

TABLE A.4 (Continued)

TOOL MODEL SIZES AND SUB-SIZES

			SIZE		
MODEL	NAME	TOOL SIZE	DESCRIPTION		SUB-SIZE*
(MODID)		(MSZID)	(fastener diameter)	(head size)	(MSSZID)
25	ALLEN WRENCH, SHORT END	1	#1 & 2 SCREWS	1/16"	1
		2	#3, 4 & 5 SCREWS	3/32"	1
		3	#6 & 8 SCREWS	1/8"	1
		4	#10 SCREWS	5/32"	1
		5	3/16"	5/32"	1
		6	1/4"	3/16"	1
		7	5/16"	1/4"	1
		8	3/8"	5/16"	1
		9	7/16"	3/8"	1
		10	1/2"	3/8"	1
		11	9/16"	7/16"	1
		12	5/8"	1/2"	1
		13	3/4"	5/8"	1

* There is no user selection of sub-size for the Allen Wrench, however, the variable (MSSZID) must be set to 1.

TABLE A.4 (Continued)

TOOL MODEL SIZES AND SUB-SIZES

MODEL	NAME	TOOL SIZE	SIZE DESCRIPTION		SUB-SIZE*
(MODID)		(MSZID)	(fastener diameter)	(head size)	(MSSZID)
		1	#1 & 2 SCREWS	1/16"	1
		2	#3, 4 & 5 SCREWS	3/32"	1
		3	#6 & 8 SCREWS	1/8"	1
		4	#10 SCREWS	5/32"	1
		5	3/16"	5/32"	1
		6	1/4"	3/16"	1
26	ALLEN WRENCH	7	5/16"	1/4"	1
	LONG END	8	3/8"	5/16"	1
		9	7/16"	3/8"	1
		10	1/2"	3/8"	1
		11	9/16"	7/16"	1
		12	5/8"	1/2"	1
		13	3/4"	5/8"	1

* **There is no user selection of sub-size for the Allen Wrench, however, the variable (MSSZID) must be set to 1.**

TABLE A-5

ACCESSORY MODEL SIZES AND SUB-SIZES

MODEL	NAME	SIZE	DESCRIPTION	SUB-SIZE	DESCRIPTION
(MACMID)		(MACSZ)	(fastener diameter)	(MACSSZ)*	
27	2" EXTENSION	1	#1 - 10 SCREWS	1	-
		2	3/16" - 1/2"	1	-
		3	9/16" - 1"	1	-
28	3" EXTENSION	1	#1 - 10 XCREWS	1	-
		2	3/16" - 1/2"	1	-
		3	9/16" - 1"	1	-
29	5" EXTENSION	1	#1 - 10 SCREWS	1	-
		2	3/16" - 1/2"	1	-
		3	9/16" - 1"	1	-
30	6" EXTENSION	1	#1 - 10 SCREWS	1	-
		2	3/16" - 1/2"	1	-
		3	9/16" - 1"	1	-

* There are no user selections of sub-sizes for the Extensions, however, the variable (MACSSZ) must be set to 1.

TABLE A-5 (Continued)
ACCESSORY MODEL SIZES AND SUB-SIZES

MODEL (MACMID)	NAME	SIZE (MACSZ)	DESCRIPTION (fastener diameter)	SUB-SIZE (MACSSZ)*	DESCRIPTION
31	8" EXTENSION	1	#1 - 10 SCREWS	1	-
		2	3/16"- 1/2"	1	-
		3	9/16" - 1"	1	-
32	10" EXTENSION	1	#1 - 10 SCREWS	1	-
		2	3/A6" - 1/2"	1	-
		3	9/16" - 1"	1	-
33	12" EXTENSION	1	#1 - 10 SCREWS	1	-
		2	3/16" - 1/2"	1	-
		3	9/16" - 1"	1	-

* There are no user selections of sub-sizes for the Extensions, however, the variable (MACSSZ) must be set to 1.

TABLE A-5 (Continued)

ACCESSORY MODEL SIZES AND SUB-SIZES

MODEL	NAME	SIZE	DESCRIPTION	SUB-SIZE	DESCRIPTION	
(MACMID)		(MACSZ)	(fastener diameter)	(MACSSZ)	(fastener diameter)	(head size)
34	REGULAR SOCKET	1	#1 - 10 SCREWS	1	#1 & 2 SCREWS	1/8"
				2	#3, 4 & 5 SCREWS	3/16"
				3	#6 & 8 SCREWS	1/4"
				4	#10 SCREWS	5/16"
		2	3/16" - 1/2"	1	3/16"	3/8"
				2	1/4"	7/16"
				3	5/16"	1/2"
				4	3/8"	9/16"
				5	7/16"	5/8"
				6	1/2"	3/4"
		3	9/16" - 1"	1	9/16"	13/16"
				2	5/8"	15/16"
				3	3/4"	1 1/4"
				4	7/8"	1 5/16"
				5	1"	1 1/2"

TABLE A-5 (Continued)

ACCESSORY MODEL SIZES AND SUB-SIZES

MODEL	NAME	SIZE	DESCRIPTION	SUB-SIZE	DESCRIPTION	
(MACMID)		(MACSZ)	(fastener diameter)	(MACSSZ)	(fastener diameter)	(head size)
35	DEEP WELL SOCKET	1	#1 - 10 SCREWS	1	#1 & 2 SCREWS	1/8"
				2	#3, 4 & 5 SCREWS	3/16"
				3	#6 & 8 SCREWS	1/4"
				4	#10 SCREWS	5/16"
		2	3/16" - 1/2"	1	3/16"	3/8"
				2	1/4"	7/16"
				3	5/16"	1/2"
				4	3/8"	9/16"
				5	7/16"	5/8"
				6	1/2"	3/4"
		3	9/16" - 1"	1	9/16"	13/16"
				2	5/8"	15/16"
				3	3/4"	1 1/4"
				4	7/8"	1 5/16"
				5	1"	1 1/2"

TABLE A-5 (Continued)

ACCESSORY MODEL SIZES AND SUB-SIZES

MODEL	NAME	SIZE	DESCRIPTION	SUB-SIZE	DESCRIPTION	
(MACMID)		(MACSZ)	(fastener diameter)	(MACSSZ)	(fastener diameter)	(head size)
36	HEX HEAD SOCKET	1	#1 - 10 SCREWS	1	#1 & 2 SCREWS	1/16"
				2	#3, 4 & 5 SCREWS	3/32"
				3	#6 & 8 SCREWS	1/8"
				4	#10 SCREWS	5/32"
		2	3/16" - 1/2"	1	3/16"	5/32"
				2	1/4"	3/16"
				3	5/16"	1/4"
				4	3/8"	5/16"
				5	7/16"	3/8"
				6	1/2"	3/8"
		3	9/16" - 1"	1	9/16"	7/16"
				2	5/8"	1/2"
				3	3/4"	5/8"

TABLE A-5 (Continued)

ACCESSORY MODEL SIZES AND SUB-SIZES

MODEL	NAME	SIZE	DESCRIPTION	SUB-SIZE	DESCRIPTION	
(MACMID)		(MACSZ)	(fastener diameter)	(MACSSZ)	(fastener diameter)	(head size)
37	UNIVERSAL SOCKET	1	#1 - 10 SCREWS	1	#3, 4 & 5 SCREWS	3/16"
				2	#6 & 8 SCREWS	1/4"
				3	#10 SCREWS	5/16"
		2	3/16" - 1/2"	1	3/16"	3/8"
				2	1/4"	7/16"
				3	5/16"	1/2"
				4	3/8"	9/16"
				5	7/16"	5/8"
				6	1/2"	3/4"
		3	9/16" - 1"	1	9/16"	13/16"
				2	5/8"	15/16"
38	CHISEL	1*	-	1*	-	-

* Although there are no user selections of size, or sub-size, for the Chisel, the variables (MACSZ)and (MACSSZ) must be set to 1.

TABLE A-6
TOOL AND ACCESSORY COMBINATIONS

TOOL		ACCESSORY			OPTIONAL/
MODEL	NAME	CLASS	NAME	MODEL	REQUIRED
(MODID)		(MACCLS)		(MACMID)	
1	HAMMER	7	CHISEL	38	OPT
13	TORQUE WRENCH	6	EXTENSION	27 28 29 30 31 32 33	OPT
		5	SOCKET	34 35 36	REQ
14	RATCHET WRENCH	6	EXTENSION	27 28 29 30 31 32 33	OPT
		5	SOCKET	34 35 36 37	REQ
15	BREAKER BAR	6	EXTENSION	27 28 29 30 31 32 33	OPT
		5	SOCKET	34 35 36 37	REQ
22	SPEED HANDLE	6	EXTENSION	27 28 29 30 31 32 33	OPT
		5	SOCKET	34 35 36	REQ

APPENDIX B

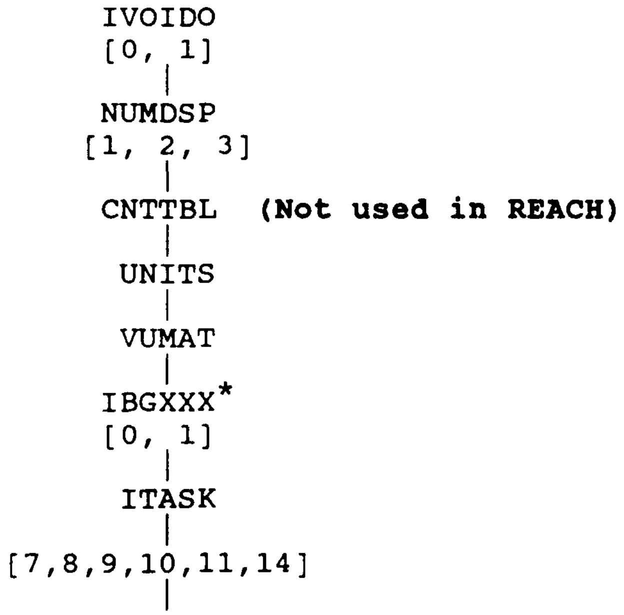
MANUAL MATERIALS HANDLING TASKS

MANUAL MATERIALS HANDLING TASKS

The logical sequence for setting the interface control variables used by the six Manual Materials Handling tasks (found in the FORTRAN common blockS XXXCTL) is dependent on the values of other variables in the common. This dependency, which includes the type of data stored or whether or not the variable needs to be set, is shown in the following variable dependency diagrams. The branches shown in the diagrams indicate the critical dependencies.

The items enclosed in brackets indicate the valid range for the variables. The items enclosed in parentheses show the condition which selects each branch. Variables shown without values are typically screen coordinates or are element identifiers (see Section 3 for details).

The Figure B.1 shows the variables common to all six Manual Materials Handling tasks (including value ranges). Figures B.2 through B.7 show the dependency of the variables specific to the individual tasks.



*XXX = CRY, HLD, PSH PLL, LFT, REC

Figure B.1. Variables Common to all Six Manual Materials Handling Tasks.

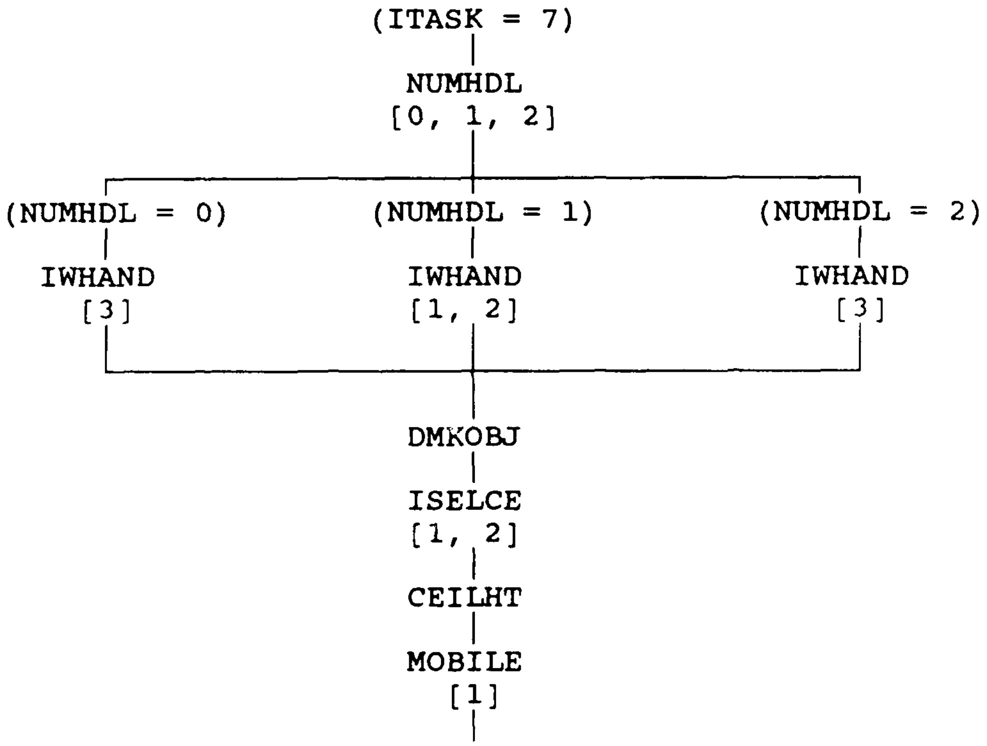


Figure B.2. Variable Dependency for CARRY.

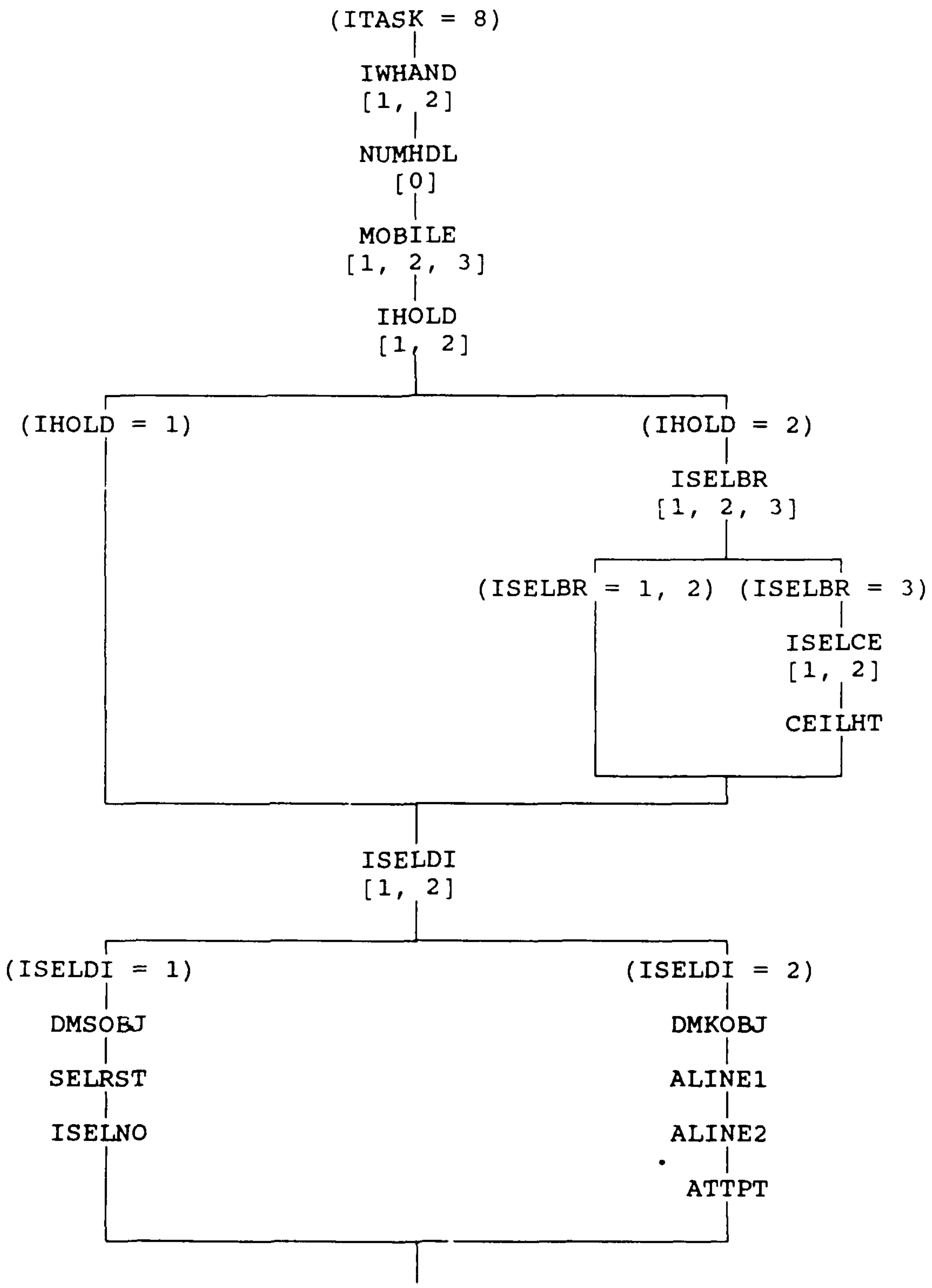


Figure B.3. Variable Dependency for HOLD.

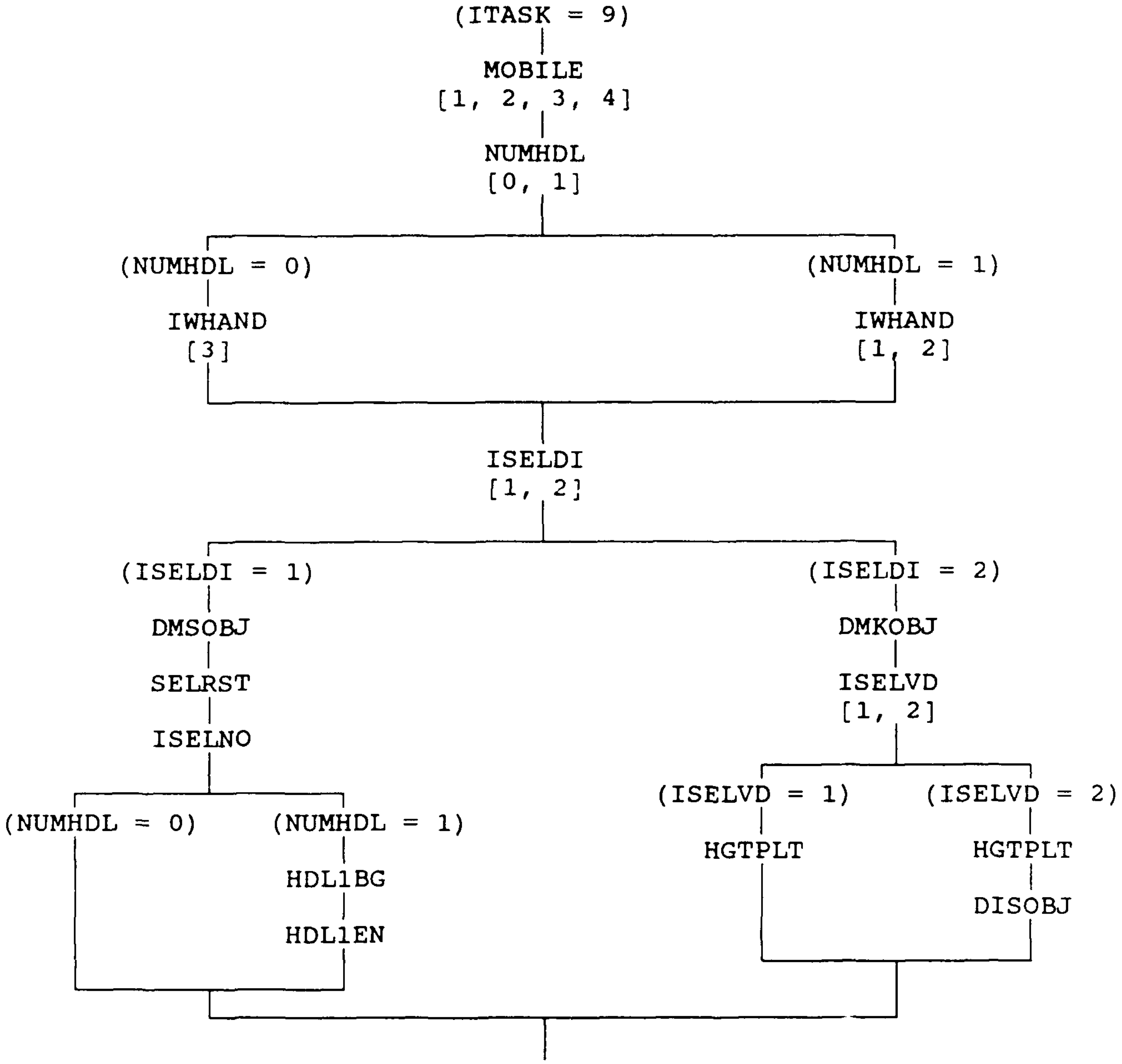


Figure B.4. Variable Dependency for LIFT.

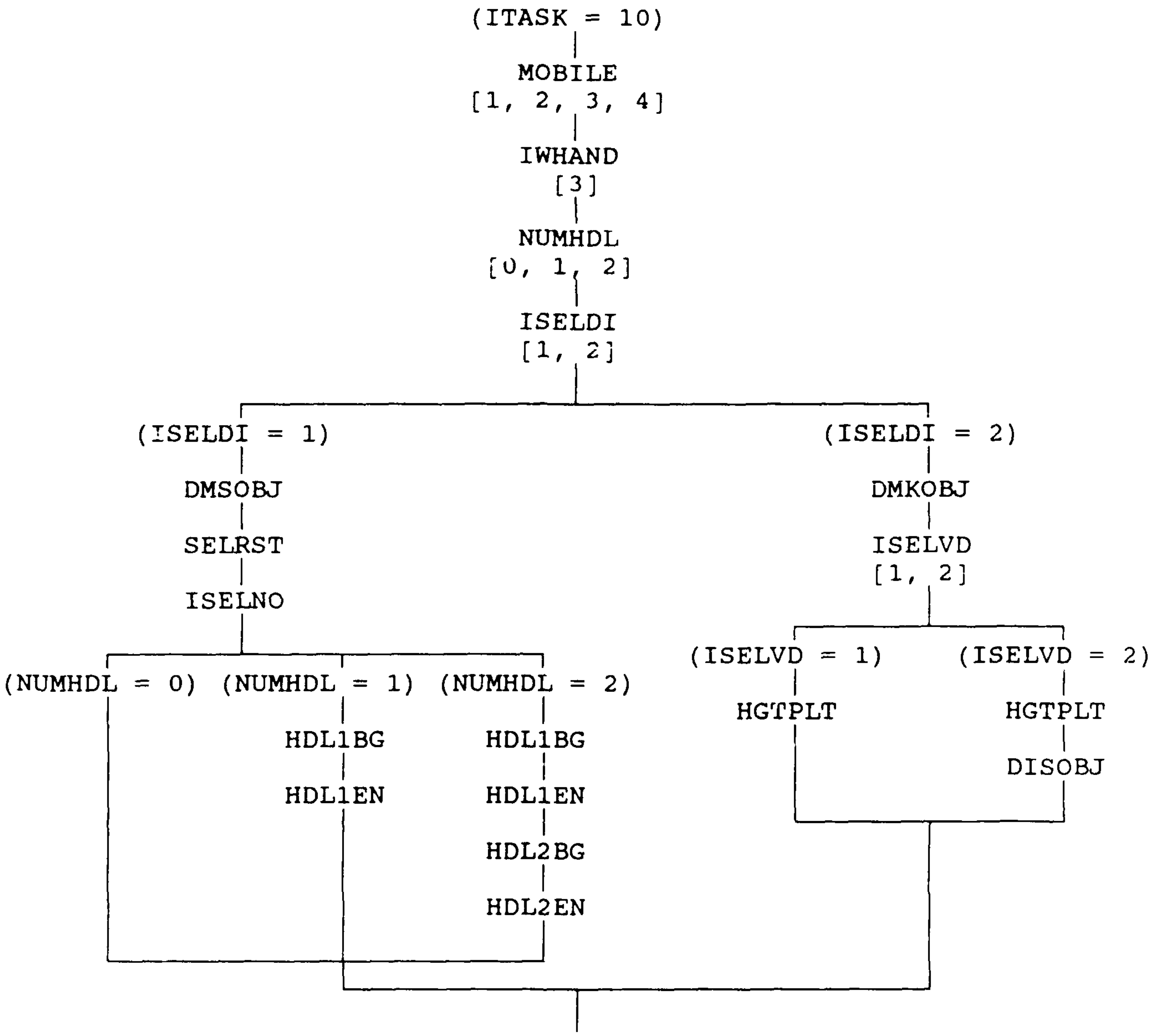


Figure B.5. Variable Dependency for PUSH.

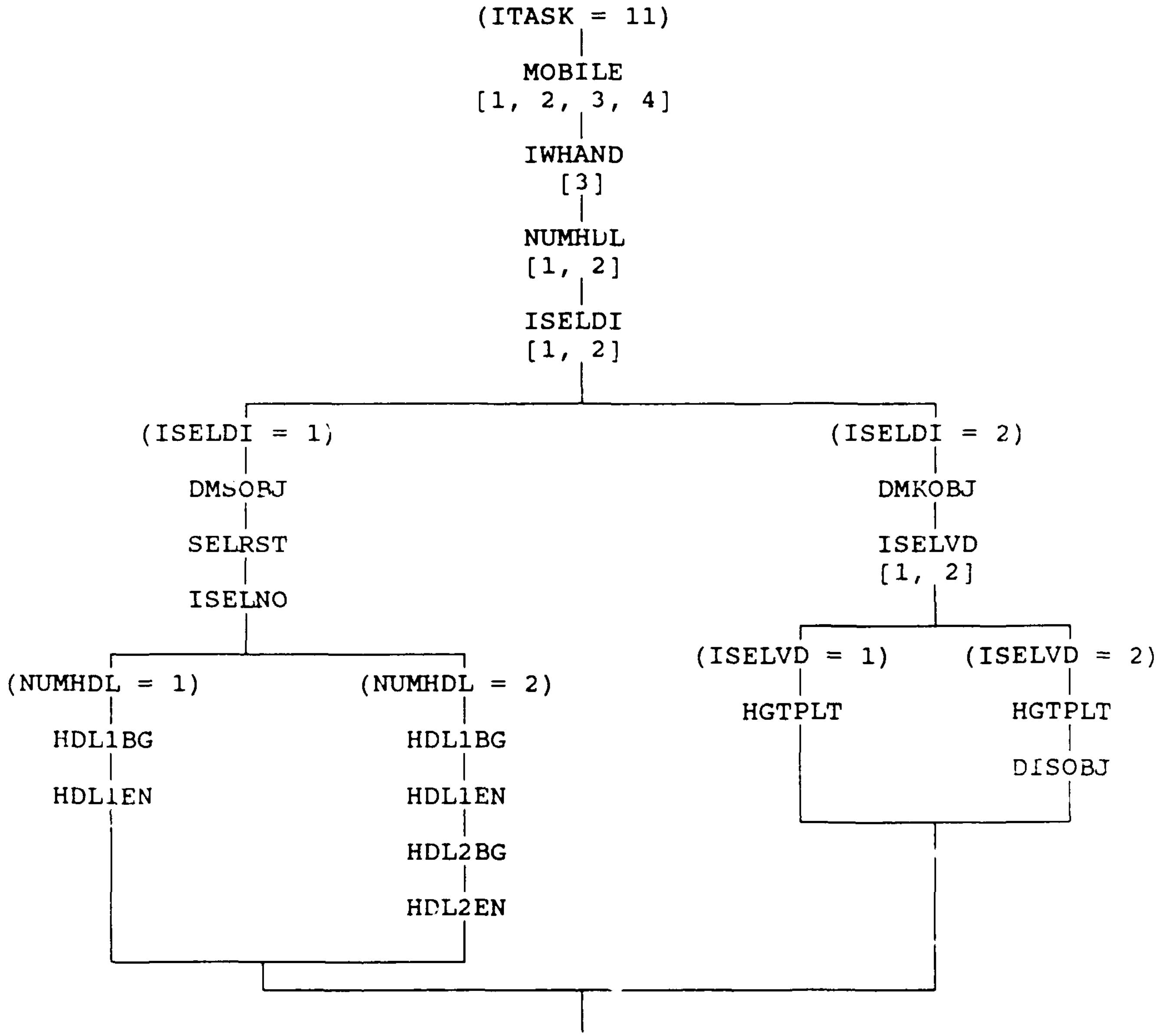


Figure B.6. Variable Dependency for PULL.

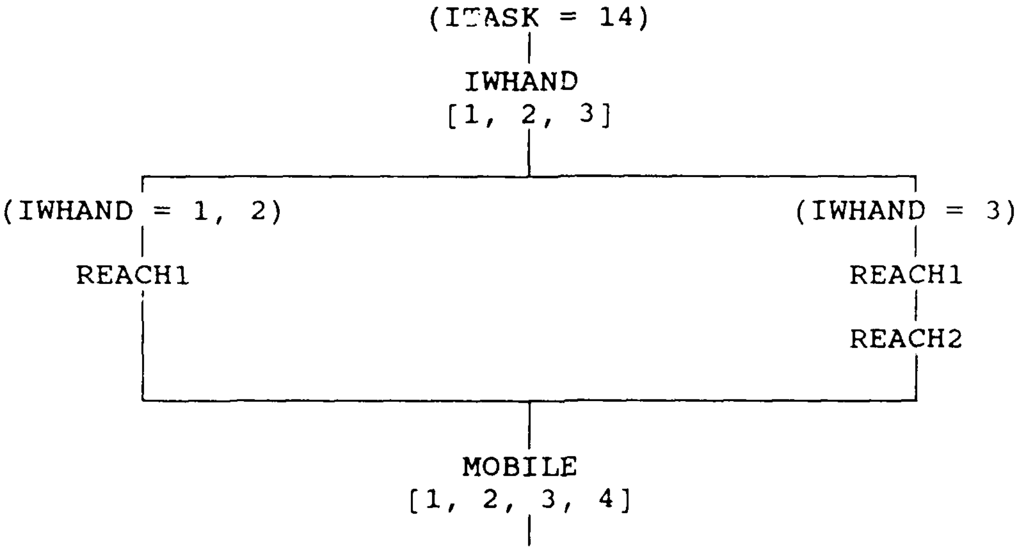


Figure B.7. Variable Dependency for REACH.

© U.S.Government Printing Office: 1990 748-002/20207