76071100

CONTROL DATA
CORPORATION

CONTROL DATA®
CYBER 70 SERIES MODELS 72/73/74/76
6000 SERIES COMPUTER SYSTEMS
7600 COMPUTER SYSTEM
CYBER 170 SERIES COMPUTER SYSTEM

NOS 1, NOS/BE 1, SCOPE 2
APPLICATION INSTALLATION HANDBOOK

REVISION RECORD

REVISION	DESCRIPTION
A	Initial printing
(6-15-75)	
B	Addition of NOS 1.0 Operating System (Part IV) and APEX-III Parametrics Option to KRONOS 2.1, SCOPE 2.1
(6-23-75)	and 3.4.
C	Addition of NOS/BE Operating System (Part V) and Graphics Product Family applications.
(3-31-76)	Other minor changes made.
D	Addition of TOTAL/ATHENA application product in Parts II, IV, and V; 274/IGS in Parts II and V; APL in
(7-13-76)	Part IV; extensive changes to GPSS V/6000 in Parts I, II, IV and V; and elimination of all ".0" version
	designations in text because of constant upgrading which resulted in excessive manual revision.
E	Addition of UNIPLOT application product in Parts IV and V; and changes to APL in Part IV. Other minor
(12-28-76)	changes made. Corresponds to PSR level 439.
F	Changes made to TOTAL in Part IV to reflect new features of Version 1.5 of TOTAL and TOTAL Extended.
(1-31-77)	Corresponds to PSR level 439.
G	Changes made to TOTAL/ATHENA in Parts II, IV, and V. Other minor corrections made. Addition of List of
(8-15-77)	Effective Pages (pages iii and iv). Corresponds to PSR levels 452 (NOS 1), and 454 (SCOPE 3.4 and NOS/BE 1).
H	Heavy revision made to APEX-III in Parts III, IV, and V. All computer printouts removed from Parts III and IV
(11-21-77)	which reduces the size of the handbook by 91 pages. Corresponds to PSR levels 258 (SCOPE 2), 460 (NOS 1),
	and 461 (NOS/BE 1).
J	Addition of APEX-III Usage Package 1 and Usage Accounting Utility application products in Parts IV and V.
(12-30-77)	The Usage Accounting Utility replaces the SURPASS 1 application product in Part IV. Also deleted are the
	PSR modification paragraphs under LCGT/IGS 2 in Parts IV and V. Other minor changes made. Corresponds
	to PSR levels 460 (NOS) and 461 (NOS/BE).
K	Complete revision of the 777/IGS application product in Part V to correspond to Version 2.2 of 777/IGS.
(1-20-78)	Corresponds to PSR level 461.
L	APL 2 in Part IV completely revised and expanded with the release of APL Version 2.1.012 under NOS 1.2 PSR
(2-27-78)	level 460. Minor changes made to APT IV 2 in Parts IV and V.
M	Handbook completely revised and reorganized. Number of Parts (operating systems) reduced from five to three
(6-12-78)	with the deletion of KRONOS 2.1 (Part I) and SCOPE 3.4 (Part II). NOS 1 (formerly PART IV) moved to Part I
	and NOS/BE 1 (formerly Part V) moved to Part II. SCOPE 2 remains Part III. Application products GODAS,
	Math Science Library, 241/IGS, and 274/IGS deleted from this handbook. New products include IMSL 6,
	TIGS 1, and XEDIT 3. Handbook completely retyped in double-column format with numerous changes and
	corrections made throughout. Corresponds to PSR level 472. This revision obsoletes all previous editions.
N	Addition of TOTAL Universal 2 and TOTAL Universal Extended 2 in Part I. Changes made to APL 2, GPSS
(9-11-78)	V/6000 1.2, and XEDIT 3 in Part I; GPSS V/6000 1.2 in Part II; and APEX-III in Part III. Other minor changes
	made. Corresponds to PSR levels 477 (NOS 1) and 278 (SCOPE 2).
P	Addition of TOTAL Universal 2 in Part II. Minor change in Usage Accounting Utility release tape name in
(10-30-78)	Part II. Corresponds to PSR level 481 (NOS/BE 1).
Publication No. 76071100	

REVISION LETTERS I, O, Q AND X ARE NOT USED

Address comments concerning this
manual to:

CONTROL DATA CORPORATION
Publications and Graphics Division
Data Services Publications
P.O. Box 0 HQC02C
Minneapolis, MN. 55440

or use Comment Sheet in the back of
this manual.

REVISION RECORD (CONT'D)

REVISION	DESCRIPTION
R	Changes to Part I, NOS 1 to reflect new products: TIGS 1 Products, PDS/MaGen, SIMSCRIPT II.5, and
(1-18-79)	UNIPLOT Products. Changes to TOTAL/ATHENA and XEDIT 3. Corresponds to PSR level 485 (NOS).
S	Changes to Part II, NOS/BE 1: TIGS 1 section completely revised including new Sanders Graphic 7 Post
(2-16-79)	Processor Option, and changes to TOTAL/ATHENA section. Corresponds to PSR level 488 (NOS/BE).

Publication No.
76071100

LIST OF EFFECTIVE PAGES

New features, as well as changes, deletions, and additions to information in this manual, are indicated by bars in the margins or by a dot near the page number if the entire page is affected. A bar by the page number indicates pagination rather than content has changed.

PAGE	REV
Front Cover	-
Title Page and ii	-
ii-a/ii-b	-
iii/iv	S
v/vi	M
vii and viii	R
ix through xi	S
Tab - NOS 1	
I-1-1 and I-1-2	R
I-2-1 through I-2-4	M
I-2-5 and I-2-6	N
I-3-1	N
I-3-2	M
I-3-3	N
I-3-4	M
I-3-5 and I-3-6	N
I-3-7	M
I-4-1 through I-4-3	M
I-5-1	M
I-5-2	N
I-5-3	M
I-6-1	N
I-6-2	M
I-7-1 through I-7-4	M
I-8-1	N
I-8-2	M
I-9-1 and I-9-2	R
I-10-1 through I-10-8	R
I-11-1	N
I-11-2	M
I-11-3 through I-11-6	N
I-12-1 and I-12-2	R
I-13-1 through I-13-8	R
I-14-1 through I-14-4	M
I-15-1	R
I-16-1	R
Tab - NOS/ BE 1	
II-1-1	S
II-2-1 through II-2-4	M
II-2-5 and II-2-6	N
II-3-1 through II-3-4	M
II-4-1 through II-4-3	N
II-5-1	M
II-5-2	N
II-6-1 through II-6-4	M
II-7-1 and II-7-2	M
II-8-1 through II-8-6	M
II-9-1 and II-9-2	M
II-10-1 through II-10-12	S
II-11-1 through II-11-3	P
II-12-1 and II-12-2	S
II-13-1 through II-13-5	M
II-14-1	P
II-14-2 through II-14-4	M
Tab - SCOPE 2	
III-1-1	M
III-2-1 through III-2-4	N
III-2-5	M
III-3-1 through III-3-3	M
III-4-1 and III-4-2	M
III-5-1 and III-5-2	M
Index-1	M
Index-2	N
Index-3	S
Comment Sheet/Mailer	-
Back Cover	-

PREFACE

This manual provides information pertaining to the installation of Control Data applications software under the following operating systems:

NOS 1	(Part I)
NOS/BE 1	(Part II)
SCOPE 2	(Part III)

As a result, this handbook is divided into parts according to operating system. Since each system relates to unique installation and modification procedures, each part in the handbook contains its own introductory material.

CONTENTS

PART I (NOS 1)

PART II (NOS/BE1)

PART III (SCOPE 2)

INDEX

FIGURES

TABLES

INTRODUCTION 1

INSTALLATION AND MODIFICATION

Part I presents information pertaining to the installation of applications software under the Network Operating System Version 1 (NOS $^{T.M.}$ 1). While the information in this part describes the application installation, users should reference the NOS 1 Reference Manual, Volume 1 and the NOS Installation Handbook for additional information.

Control Data assumes that the installation of applications software will be performed by an analyst with experience with a CDC® CYBER 70 or CDC® CYBER 170 series computer system. Control Data is not responsible for the proper functioning of undescribed features.

RELEASE TAPES

The following application products are available under NOS 1:

Application	Tape	Section
APEX-III Out-of-Core System 1	REL57B	2
APEX-III Mixed Integer Programming Option	REL57C	2
APEX-III Matrix Reduction Option	REL57D	2
APEX-III Parametrics Option	REL57E	2
APEX-III Usage Package 1	REL57F	2
APL 2	REL60	3
APT IV 2	PL59	4
GPSS V/6000 1.2	REL50	5
IMSL 6	REL63	6
LCGT/IGS 2	REL56A	7
PERT/TIME 2	REL52A	8
SIMSCRIPT II.5 V4.2	REL64	9
TIGS 1	REL62A	10
Tektronix 401X Post Processor Option Under TIGS	REL62B	10
TOTAL Universal 1	REL54A	11
TOTAL Universal Extended 1	REL54B	11
TOTAL Universal 2	REL54C	11
TOTAL Universal Extended 2	REL54D	11
TOTAL/ATHENA	REL58	12
UNIPLOT 3	REL66	13
CalComp 906 Post Processor Option Under UNIPLOT	REL66A	13
Houston Instrument BTC-7 Post Processor Option Under UNIPLOT	REL66B	13
Tektronix 401X Post Processor Option Under UNIPLOT	REL66C	13
Usage Accounting Utility	APO01N	14
XEDIT 3	REL61	15
PDS/MaGen	REL65	16

RELATED DOCUMENTATION

The following manuals contain relevant information for the operating system or application products described in part I.

Title	Publication No.
NOS 1 Installation Handbook	60435700
NOS 1 Reference Manual, Volume 1	60435400

Title	Publication No.
APEX-III Reference Manual	76070000
APL 2 Reference Manual	60454000
APT IV Internal Maintenance Specification	*
APT IV Reference Manual	17326900
Beginning Graphics User Guide	76077300
GPSS V/6000 General Information Manual	84003900
GPSS V User's Manual	76078800
IMSL Reference Manual	SMD150043*
LCGT/IGS Reference Manual	76079100
LCGT/IGS User Guide	76077400
PDS/MAGEN User Information Manual	84009900
PERT/Time Reference Manual	60133600
SIMSCRIPT II.5 User Information Manual	84000460
SIMSCRIPT II.5 Instant	84000450
TIGS V1 Reference Manual	60455940
TOTAL/ATHENA User Information Manual	84000090
TOTAL Universal Reference Manual	76070300
UNIPLOT Reference/User Guide	60454730
Usage Accounting Utility Reference Manual	84000440
XEDIT 3 Reference Manual	60455730
2550 Computer System Software Reference Manual CCP 1.0	74701200

*These manuals may be ordered by title from Control Data Corporation, Software Manufacturing and Distribution, 4201 North Lexington Avenue, St. Paul, Minnesota 55112.

APEX-III OUT-OF-CORE SYSTEM 1

RELEASE DESCRIPTIONS

APEX-III Out-of-Core System 1 runs under NOS. The FORTRAN Extended compiler is required for compiling and updating APEX-III. APEX-III Out-of-Core System 1 is designed to execute from permanent files.

Hardware Requirements

APEX-III Out-of-Core System 1 requires the same minimum hardware configuration as NOS. A minimum field length of 64K octal is required to execute APEX-III and 70K octal is required for installation.

Deficiencies

None.

Release Materials

APEX-III Out-of-Core System 1 resides on the tape known as REL57B. REL57B is a program library in UPDATE format for APEX-III Out-of-Core System 1. It has the following characteristics: 7-track (800 bits per inch [bpi]) or 9-track (800 or 1600 characters per inch [cpi]), binary recording mode, OOCAPEXIIIV1P1 as file ID in HDR1 label, and six files:

File 1 - installation deck in UPDATE PL form,
File 2 - empty,
File 3 - empty,
File 4 - source code in UPDATE PL form,
File 5 - verification deck in UPDATE PL form, and
File 6 - sample output from verification run.

Listings of the installation deck (APEXC1) and the verification deck (APEXCV) may be obtained from REL57B by executing the following job:

```
LIST,T50.
USER.           (Supply valid USER card.)
LABEL(OLDPL,R,L=OOCAPEXIIIV1P1,
        ( HY )
    D = { HD }  ,VSN=APXOOC)
        ( PE )
UPDATE(F,*==,L=7)
SKIPF(OLDPL,4,17)
UPDATE(F,*==,L=7,R)
7/8/9
6/7/8/9
```

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURE

APEX-III Out-of-Core System 1 is installed by executing deck APEXC1, contained on the first file of REL57B. This job deck reads the APEX-III system from REL57B, compiles the system, and installs an executable file of the APEX-III system on a permanent file named APEXIII with the password APEX.

This job deck also creates three permanent files which are needed when one uses the APEX-III USER feature. (Reference the APEX-III Reference Manual for detailed information.) For those installations where the USER feature is used infrequently, it may be desirable to maintain these three files on magnetic tape rather than permanent files. These three files are described below:

Name	Password	Contents
APEXPL	APEX	Partial APEX-III source code. Contains APEX-III COMMON regions and subroutines QBUFFER, QGET, QUSER, and QUSERDA.
APEXOV	APEX	Relocatable binary code of APEX-III overlay drivers.
APEXRL	APEX	Relocatable binary code of remaining APEX-III subroutines.

For those installations which have purchased any of the APEX-III options, deck APEXC1 may also be used to install the APEX-III Out-of-Core System 1 with any combination of the APEX-III Mixed Integer Programming Option (MIP), APEX-III Matrix Reduction Option, and APEX-III Parametrics Option.

The deck APEXC1 contains UPDATE IF,DEF directives to simplify installation. The names used by these directives and their meaning when DEFINEd are as follows:

Name	Effect on Installation Job
MT	Uses 7-track magnetic tapes.
HD	Uses 9-track, 800 cpi, magnetic tapes.
PE	Uses 9-track; 1600 cpi, magnetic tapes.

Name	Effect on Installation Job
MIP	APEX-III Mixed Integer Programming Option is installed with APEX-III Out-of-Core System 1.
RED	APEX-III Matrix Reduction Option is installed with APEX-III Out-of-Core System 1.
PAR	APEX-III Parametrics Option is installed with APEX-III Out-of-Core System 1.

APEXC1 may be extracted from REL57B and submitted for execution via the following job. Note that APEXC1 requires a change for execution. A valid user card must be supplied.

```
INST,T50.
USER.              (Supply valid USER card.)
LABEL(OLDPL,R,L=OOCAPEXIIIV1P1,
        ( HY )
    D=  { HD }  ,VSN=APXOOC)
        ( PE )
UPDATE(Q,*==,D,8)
SUBMIT(COMPILE,B)
7/8/9
          ( MT )
=DEFINE   { HD }
          ( PE )
=DEFINE MIP    (If Mixed Integer Programming
               Option.)
=DEFINE RED    (If Matrix Reduction Option.)
=DEFINE PAR    (If Parametrics Option.)
=D,APEXC1.3
USER.          (Supply valid user card.)
=C,APEXC1
7/8/9
6/7/8/9
```

Error Messages

A nonfatal loader error will be encountered in loading the APEX-III system. The error message is of the form:

```
NE4102///DUPLICATE ENTRY POINT NAME - EXIT
PROGRAM NAME ------ FORSYS=
LAST FILE ACCESSED - FORTRAN
```

Errors of any other form should be investigated.

Installation Verification

The following job may be used to extract job deck APEXCV from REL57B and submit for execution to verify installation of APEX-III Out-of-Core System 1.

```
VER,T50.
USER.              (Supply valid USER card.)
LABEL(OLDPL,R,L=OOCAPEXIIIV1P1,
        ( HY )
    D=  { HD }  ,VSN=APXOOC)
        ( PE )
SKIPF(OLDPL,4,17)
UPDATE(Q,*==,D,8,R)
SUBMIT(COMPILE,B)
7/8/9
=D,APEXCV.3
USER.              (Supply valid USER card.)
=C,APEXCV
7/8/9
6/7/8/9
```

File 6 of REL57B can be copied to OUTPUT and used to validate correct execution of the verification run.

APEX-III MIXED INTEGER PROGRAMMING OPTION (MIP)

RELEASE DESCRIPTIONS

The APEX-III Mixed Integer Programming Option runs under NOS when installed with APEX-III Out-of-Core System 1. The FORTRAN Extended compiler is required for compiling and updating APEX-III MIP. APEX-III MIP is designed to execute from permanent files.

Hardware Requirements

APEX-III MIP requires the same minimum hardware configuration as NOS. A minimum field length of 64K octal is required for execution and 70K is required for installation.

Deficiencies

None.

Release Materials

The APEX-III Mixed Integer Programming Option resides on the tape known as REL57C. REL57C is a program library in UPDATE format for APEX-III Mixed Integer Programming Option. It has the following characteristics: 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, MIPAPEXIIIV1P1 as file ID in HDR1 label, six files:

File 1 - not used,
File 2 - empty,
File 3 - empty,
File 4 - source code in UPDATE PL form,
File 5 - verification deck in UPDATE PL form, and
File 6 - sample output from verification run.

A listing of the verification deck (APEXMV) may be obtained from REL57C by executing the following job:

```
LIST,T50.
USER.            (Supply valid USER card.)
LABEL(OLDPL,R,L=MIPAPEXIIIV1P1,
       ⎧ HY ⎫
    D= ⎨ HD ⎬ ,VSN=APXMIP)
       ⎩ PE ⎭
SKIPF(OLDPL,4,17)
UPDATE(F,*==,L=7,R)
7/8/9
6/7/8/9
```

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURE

The APEX-III Mixed Integer Programming Option must be installed with APEX-III Out-of-Core System 1. The installation deck, APEXC1, on the APEX-III Out-of-Core System 1 tape, known as REL57B, is used to install the MIP Option. The name MIP is defined for the UPDATE IF,DEF directives in deck APEXC1 to merge the MIP Option with the out-of-core system.

Reference "Installation Procedure" for APEX-III Out-of-Core System 1 for details of installation and expected error messages.

Installation Verification

The following job may be used to extract job deck APEXMV from REL57C and submit for execution to verify installation of the MIP Option.

```
VER,T50.
USER.            (Supply valid USER card.)
LABEL(OLDPL,R,L=MIPAPEXIIIV1P1,
       ⎧ HY ⎫
    D= ⎨ HD ⎬ ,VSN=APXMIP)
       ⎩ PE ⎭
SKIPF(OLDPL,4,17)
UPDATE(Q,*==,D,8,R)
SUBMIT(COMPILE,B)
7/8/9
=D,APEXMV.3
USER.            (Supply valid USER card.)
=C,APEXMV
7/8/9
6/7/8/9
```

File 6 of REL57C can be copied to OUTPUT and used to validate correct execution of the verification run.

APEX-III MATRIX REDUCTION OPTION (REDUCE)

RELEASE DESCRIPTIONS

The APEX-III Matrix Reduction Option runs under the NOS operating system when installed with APEX-III Out-of-Core System 1. The FORTRAN Extended compiler is required for compiling and updating APEX-III REDUCE. APEX-III REDUCE is designed to execute from permanent files.

Hardware Requirements

APEX-III REDUCE requires the same minimum hardware configuration as NOS except that a minimum field length of 64K is required for execution and 70K octal is required for installation.

Deficiencies

None.

Release Materials

The APEX-III Matrix Reduction Option resides on the tape known as REL57D. REL57D is a program library in UPDATE format for APEX-III Matrix Reduction Option. It has the following characteristics: 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, REDAPEXIIIV1P1 as file ID in HDR1 label, six files:

File 1 - not used,
File 2 - empty,
File 3 - empty,
File 4 - source code in UPDATE PL form,
File 5 - verification deck in UPDATE PL form, and
File 6 - sample output from verification run.

A listing of the verification deck (APEXRV) may be obtained from REL57D by executing the following job:

```
LIST,T50.
USER.            (Supply valid USER card.)
LABEL(OLDPL,R,L=REDAPEXIIIV1P1,
       ⎧ HY ⎫
    D= ⎨ HD ⎬ ,VSN=APXRED)
       ⎩ PE ⎭
SKIPF(OLDPL,4,17)
UPDATE(F,*==,L=7,R)
7/8/9
6/7/8/9
```

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURE

The APEX-III Matrix Reduction Option must be installed with APEX-III Out-of-Core System 1. The installation deck, APEXC1, on the APEX-III Out-of-Core System 1 tape, known as REL57B, is used to install the REDUCE Option. The name RED is defined for the UPDATE IF,DEF directives in deck APEXC1 to merge the REDUCE Option with the out-of-core system.

(Reference "Installation Procedure" for APEX-III Out-of-Core System 1 for details of installation and expected error messages.)

Installation Verification

The following job may be used to extract job deck APEXRV from REL57D and submit for execution to verify installation of the REDUCE Option.

```
VER,T50.
USER.             (Supply valid USER card.)
LABEL(OLDPL,R,L=REDAPEXIIIV1P1,
       ⎧ HY ⎫
    D= ⎨ HD ⎬  ,VSN=APXRED)
       ⎩ PE ⎭
SKIPF(OLDPL,4,17)
UPDATE(Q,*==,D,8,R)
SUBMIT(COMPILE,B)
7/8/9
=D,APEXRV.3
USER.             (Supply valid USER card.)
=C,APEXRV
7/8/9
6/7/8/9
```

File 6 of REL57D can be copied to OUTPUT and used to validate correct execution of the verification run.

APEX-III PARAMETRICS OPTION

RELEASE DESCRIPTIONS

The APEX-III Parametrics Option runs under the NOS operating system when installed with APEX-III Out-of-Core System 1. The FORTRAN Extended compiler is required for compiling and updating APEX-III Parametrics. APEX-III Parametrics is designed to execute from permanent files.

Hardware Requirements

APEX-III Parametrics requires the same minimum hardware configuration as NOS except that a minimum field length of 64K octal is required for execution and 70K octal is required for installation.

Deficiencies

None.

Release Materials

The APEX-III Parametrics Option resides on the tape known as REL57E. REL57E is a program library in UPDATE format for APEX-III Parametrics Option. It has the following characteristics: 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, PARAPEXIIIV1P1 as file ID in HDR1 label, six files:

File 1 - not used,
File 2 - empty,
File 3 - empty,
File 4 - source code in UPDATE PL form,
File 5 - verification deck in UPDATE PL form, and
File 6 - sample output from verification run.

A listing of the verification deck (APEXPV) may be obtained from REL57E by executing the following job:

```
LIST,T50.
USER.             (Supply valid USER card.)
LABEL(OLDPL,R,L=PARAPEXIIIV1P1,
       ⎧ HY ⎫
    D= ⎨ HD ⎬  ,VSN=APXRED)
       ⎩ PE ⎭
SKIPF(OLDPL,4,17)
UPDATE(F,*==,L=7,R)
7/8/9
6/7/8/9
```

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURE

The APEX-III Parametrics Option must be installed with APEX-III Out-of-Core System 1. The installation deck, APEXC1, on the APEX-III Out-of-Core System 1 tape, known as REL57B, is used to install the Parametrics Option. The name PAR is defined for the UPDATE IF,DEF directives in deck APEXC1 to merge the Parametrics Option with the out-of-core system.

(Reference "Installation Procedure" for APEX-III Out-of-Core System 1 for details of installation and expected error messages.)

Installation Verification

The following job may be used to extract job deck APEXPV from REL57E and submit for execution to verify installation of the Parametrics Option.

```
VER,T50.
USER.              (Supply valid USER card.)
LABEL(OLDPL,R,L=PARAPEXIIIV1P1,
      ⎧ HY ⎫
   D= ⎨ HD ⎬  ,VSN=APXPAR)
      ⎩ PE ⎭
SKIPF(OLDPL,4,17)
UPDATE(Q,*==,D,8,R)
SUBMIT(COMPILE,B)
7/8/9
=D,APEXPV.3
USER.              (Supply valid USER card.)
=C,APEXP3
7/8/9
6/7/8/9
```

File 6 of REL57E can be copied to OUTPUT and used to validate correct execution of the verification run.

APEX-III USAGE PACKAGE 1

RELEASE DESCRIPTIONS

APEX-III Usage Package 1 runs under the NOS operating system. The FORTRAN Extended compiler is required for compiling APEX-III when using the USER feature. APEX-III is designed to execute from permanent files.

Installation of APEX-III Usage Package 1 requires that the Usage Accounting Utility be installed and that a product update run be made to add the APEX-III product to the accounting files. (Consult the Usage Accounting Utility Reference Manual for further details.)

The product activity card required for APEX-III has the following form:

```
CDAPEX III   A A 4      APEX
```

Columns	Contents
1 to 2	CD is the vendor code.
3 to 12	APEX III is the product name.
14	A signifies add product information to product name file.
16	A signifies the product is active.
18	4 is the ID value.
20 to 23 25 to 28	Represent threshhold values-consult Usage Accounting Utility Reference Manual if usage of APEX-III is to be limited.
30 to 80	APEX is the software code.

Hardware Requirements

APEX-III Usage Package 1 requires the same minimum hardware configuration as NOS except that a minimum field length of 70K octal is required for execution of APEX-III.

Deficiencies

None.

Release Materials

APEX-III Usage Package 1 resides on the tape known as REL57F. REL57F contains relocatable binary code for APEX-III Usage Package 1. It has the following characteristics: 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, AUUAPEXIIIV1P1 as file ID in HDR1 label, seven files:

File 1 - installation deck in UPDATE PL form,
File 2 - relocatable binary code of APEX-III overlay drivers,
File 3 - relocatable binary code of remaining APEX-III system routines,
File 4 - partial source code in UPDATE PL form for USER feature,
File 5 - verification deck in UPDATE PL form,
File 6 - sample output from verification run, and
File 7 - list file of documentation of selected APEX-III routines for USER feature.

Listings of the installation deck (APEXU1) and the verification deck (APEXUV) may be obtained from REL57F by executing the following job:

```
LIST,T50.
USER.              (Supply valid USER card.)
LABEL(OLDPL,R,L=AUUAPEXIIIV1P1,
      ⎧ HY ⎫
   D= ⎨ HD ⎬  ,VSN=APXAUU)
      ⎩ PE ⎭
UPDATE(F,*==,L=7)
SKIPF(OLDPL,4,17)
UPDATE(F,*==,L=7,R)
7/8/9
6/7/8/9
```

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURE

APEX-III Usage Package 1 is installed by executing deck APEXU1, contained on the first file of REL57F. This job deck reads relocatable binary code of the APEX-III system from REL57F and installs an executable file of the APEX-III system on a permanent file named UAPEX3 with the password APEX.

This job deck will optionally create three permanent files which are needed when one uses the APEX-III USER feature. (See the APEX-III Reference Manual for detailed information.) For those installations where the USER feature is used infrequently, it may be desirable to reference these files directly from the installation tape when needed rather than maintaining them on permanent files. These three files are described as follows:

Name	Password	Contents
UAPEXPL	APEX	Partial APEX-III source code. Contains APEX-III COMMON regions and subroutines QBUFFER, QGET, QUSER and QUSERDA. Same as file 4 of installation tape.
UAPEXOV	APEX	Relocatable binary code of APEX-III overlay drivers. Same as file 2 of installation tape.
UAPEXRL	APEX	Relocatable binary code of remaining APEX-III subroutines. Same as file 3 of installation tape.

The deck APEXU1 contains UPDATE IF,DEF directives to simplify installation. The names used by these directives and their meaning when DEFINED are as follows:

Name	Effect on Installation Job
MT	Uses 7-track magnetic tapes.
HD	Uses 9-track, 800 cpi magnetic tapes.
PE	Uses 9-track, 1600 cpi, magnetic tapes.
PF	Files 2, 3, and 4 of the installation tape are installed as permanent files with names UAPEXOV, UAPEXRL, and UAPEXPL respectively for the USER feature.

APEXU1 may be extracted from REL57F and submitted for execution via the following job. Note that APEXU1 requires a change for execution. A valid USER card must be supplied.

```
INST.T50.
USER.              (Supply valid USER card.)
LABEL(OLDPL,R,L=AUUAPEXIIIV1P1,
        HY
   D=   HD   ,VSN=APXAUU)
        PE
UPDATE(Q,*==,D,8)
SUBMIT(COMPILE,B)
7/8/9
            MT
=DEFINE     HD
            PE
=DEFINE PF        (If files for USER feature are to be
                  on permanent files.)
=D,APEXU1.3
USER.             (Supply valid USER card.)
=C,APEXU1
7/8/9
6/7/8/9
```

Error Messages

A nonfatal loader error will be encountered in loading the APEX-III system. The error message is of the form:

```
 NE4102///DUPLICATE ENTRY POINT NAME - EXIT
PROGRAM NAME ------ FORSYS=
LAST FILE ACCESSED- FORTRAN
```

Errors of any other form should be investigated.

Installation Verification

The following job may be used to extract job deck APEXUV from REL57F and submit for execution to verify installation of APEX-III Usage Package 1.

```
VER,T50.
USER.             (Supply valid USER card.)
LABEL(OLDPL,R,L=AUUAPEXIIIV1P1,
        HY
   D=   HD   ,VSN=APXAUU)
        PE
SKIPF(OLDPL,4,17)
UPDATE(Q,*==,D,8,R)
SUBMIT(COMPILE,B)
7/8/9
=D,APEXUV.3
USER.             (Supply valid USER card.)
=C,APEXUV
7/8/9
6/7/8/9
```

File 6 of REL57F can be copied to OUTPUT and used to verify correct execution of the verification run.

RELEASE DESCRIPTION

APL Version 2 runs under the NOS 1.3 time-sharing system or IAF, and as a NOS batch job.

This installation procedure is designed to be run as a batch job by installation personnel. No knowledge of the APL language or APL terminals is required.

NOTES AND CAUTIONS

This installation procedure defaults to a time-sharing based system. If an IAF-based system is desired, the =DEFINE IAF directive is required whenever APL is assembled using the EXTRACT job. See step 3 on the following pages.

This APL2 installation procedure destroys all existing files of the same name under user names APL0 and APL1. (These permanent file names are listed as files and records in the Release Tape Format section.) The one exception to this procedure is when the =DEFINE APLDEBG is selected (see step 3); the running production version of APL is not disturbed (file APLDEBG is created instead of changing file APLPROD both of which reside under user name APL0). To access this new test version of APL, the user merely includes the DEBUG parameter as the last parameter on the APL control statement:

```
APL,TT=...,DEBUG.
```

The validity of the APL system can be compromised and the security of locked proprietary functions in workspaces will be forfeited if the following information is disclosed to users:

- Password on files APLPROD and APLDEBG,
- Contents of file APLPL, and
- The internal APL character representations

INSTALLATION PROCEDURE

The procedures for the installation of APL2 are outlined in the following seven steps.

Step 1 - Validate User Names APL0 and APL1

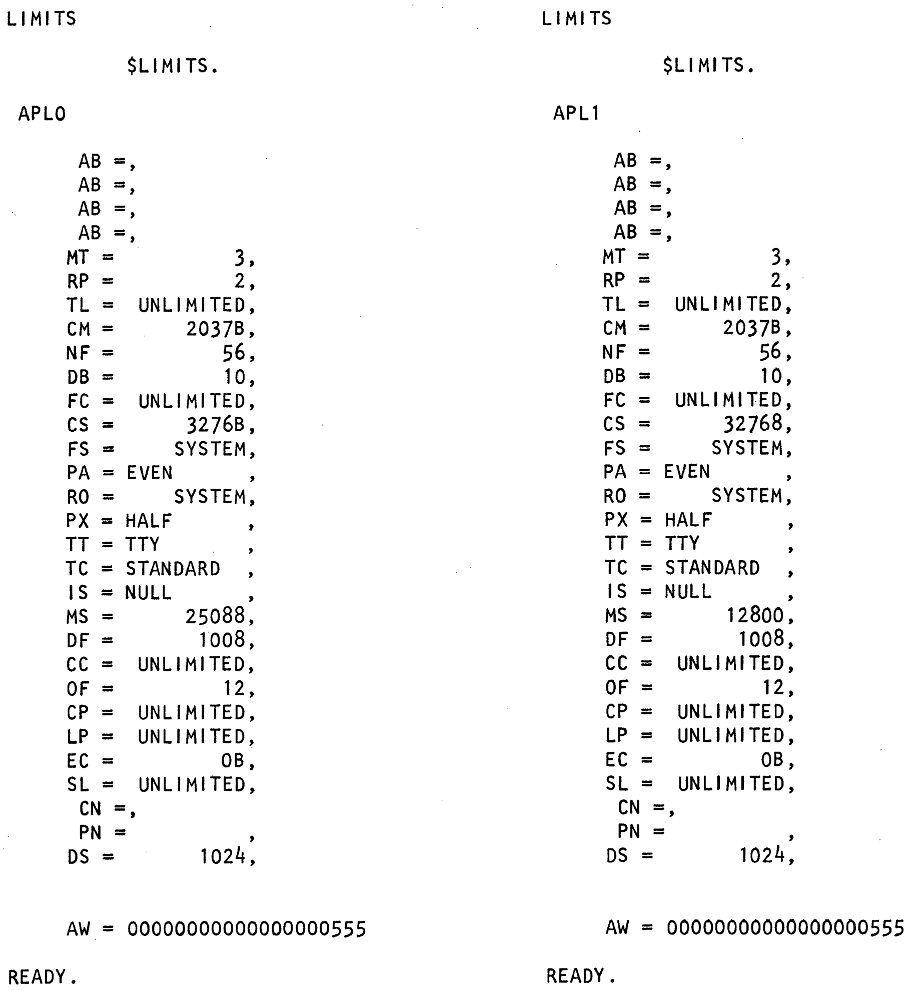
The installation of APL2 requires that two user names be validated prior to actual installation. (Check to see if user names APL0 and APL1 have been previously validated. If they have not, validate them.) APL0 must be validated for system origin privileges if a new deadstart tape is built (see step 6). Recommended LIMITS for APL0 and APL1 are shown in figure I-3-1. The LIMITS shown for APL0 assume the use of the BINARY or DISK installation directives (see step 3). If the SOURCE or IAF directive is selected, a higher value for the MS validation on LIMITS should be used or the assembly listing (file LASM) should be written directly to tape to avoid PRU LIMIT.

Step 2 - Run the LIST Job

The installation/maintenance job is file 1 on the release tape (see "Release Tape Format" section). It is used for the following functions:

- Installing the APL system from either 7- or 9-track tape,
- Installing the APL system from the released binary or source code,
- Generating an IAF-based system in place of the default time-sharing based system,
- Writing the APL system tape files to permanent disk storage,
- Installing the APL public workspaces on user name APL1,
- Enabling the APL interpreter to be installed under user name APL0 without disturbing the production processing,
- Changing the character set and hardware terminal type defaults,
- Changing or deleting the APL entry message that is displayed to users, and
- Creating a procedure file called ANEWSYS that can be used in conjunction with a tape request to create a new tape of all files on the APL system.

File 1 is a program library in UPDATE format with = (equals) as the master character, and contains all NOS jobs necessary to install and maintain APL. To perform these functions, two jobs, LIST and EXTRACT, must be executed. LIST is used to obtain an UPDATE COMPILE file listing of the installation/maintenance job. With this listing, you can determine the UPDATE directives needed to select the desired options. These directives are then applied when running the EXTRACT job which does the actual installation functions.



```
LIMITS

          $LIMITS.

 APL0

        AB =,
        AB =,
        AB =,
        AB =,
       MT =           3,
       RP =           2,
       TL =  UNLIMITED,
       CM =       2037B,
       NF =          56,
       DB =          10,
       FC =  UNLIMITED,
       CS =       3276B,
       FS =      SYSTEM,
       PA = EVEN        ,
       RO =      SYSTEM,
       PX = HALF        ,
       TT = TTY         ,
       TC = STANDARD    ,
       IS = NULL        ,
       MS =       25088,
       DF =        1008,
       CC =  UNLIMITED,
       OF =          12,
       CP =  UNLIMITED,
       LP =  UNLIMITED,
       EC =          OB,
       SL =  UNLIMITED,
        CN =,
        PN =            ,
       DS =        1024,


       AW = 00000000000000000555

READY.
```

```
LIMITS

          $LIMITS.

 APL1

        AB =,
        AB =,
        AB =,
        AB =,
       MT =           3,
       RP =           2,
       TL =  UNLIMITED,
       CM =       2037B,
       NF =          56,
       DB =          10,
       FC =  UNLIMITED,
       CS =       32768,
       FS =      SYSTEM,
       PA = EVEN        ,
       RO =      SYSTEM,
       PX = HALF        ,
       TT = TTY         ,
       TC = STANDARD    ,
       IS = NULL        ,
       MS =       12800,
       DF =        1008,
       CC =  UNLIMITED,
       OF =          12,
       CP =  UNLIMITED,
       LP =  UNLIMITED,
       EC =          OB,
       SL =  UNLIMITED,
        CN =,
        PN =            ,
       DS =        1024,


       AW = 00000000000000000555

READY.
```

Figure I-3-1. Validate LIMITS for User Names APL0 and APL1

Run the following LIST job to list the installation/maintenance job:

```
LIST.
USER.         (Any user name, password and family
              may be used here.)
CHARGE.       (Insert charge card, if required.)
LABEL(OLDPL,R,F=I,L=APL2,
 ⎧D=HY⎫                          (7-track,  800 bpi.)
 ⎨D=HD⎬  ,VSN=REL60,PO=R)        (9-track,  800 cpi.)
 ⎩D=PE⎭                          (9-track, 1600 cpi.)
UPDATE(F,*==,L=A1247,C=0)
UNLOAD(OLDPL)
6/7/8/9
```

Step 3 - Select the Desired EXTRACT Job Directives

With each EXTRACT run, the following directives must always be included:

```
=IDENT OPTIONS
=D APLINST.3,4                  (Delete USER and
                                CHARGE statements.)
USER(APL0,password,family)      (Insert your password
                                and family for APL0.)
=D,133,134                      (Delete USER and
                                CHARGE statements.)
USER(APL1,password,family)      (Insert your password
                                and family for APL1.)
```

EXTRACT directives for all other installation options are as follows:

One of the following directives must always be included to specify the format of the release tape:

```
=DEFINE HY        (To select 7-track, 800 bpi tape.)
=DEFINE HD        (To select 9-track, 800 cpi tape.)
=DEFINE PE        (To select 9-track, 1600 cpi tape.)
```

One of the following directives must always be included to select the desired installation task to be performed:

```
=DEFINE BINARY      (To install the time-sharing-
                    based ready-to-execute system.)
=DEFINE DISK        (To create disk files of the
                    release tape.)
=DEFINE SOURCE      (To compile APL from source
                    and install as time-sharing-
                    based system.)
=DEFINE IAF         (To compile APL from source
                    and install as IAF-based
                    system.)
```

One of the following directives must always be included whether the BINARY, the SOURCE, the IAF, or the DISK Option was selected above:

```
=DEFINE APLPROD     (To replace the production
                    APL interpreter.)
=DEFINE APLDEBG     (To replace the test APL
                    interpreter.)
```

If the BINARY task is selected, the default system and associated files are installed. The following permanent files are replaced:

```
APLLIB    APLPL      TAPLTST   CLEARWS
EMPTY     APLNEWS    CATALOG   WSFNS
FILESYS   FILES2     TAPLWS    NEWSF
```

See the "Release Tape Format" section, file 2, for the list of defaults values the interpreter is released as.

No further directives are allowed for the BINARY and DISK tasks except those explained in step 4.

If the APLPROD directive is selected, the old production permanent file APLPROD is replaced with the new production system.

If the APLDEBG directive is selected, the old test version permanent file APLDEBG is replaced with the new test system.

If the DISK task is selected, all release tape files are installed user name APL0 using file names given in the "Release Tape Format" section. If this option is used, the workspaces are all stored as separate records under one file named APLWS under APL0, rather than their normal APL1 location. The DISK option also creates a procedure file called ANEWSYS under APL0 which can subsequently be used to generate a new tape of the APL system, if it should be revised by the site. When this option is used, no files get installed under user name APL1 and the APL verify test job is not run.

If the SOURCE task is selected, a time-sharing-based system full assembly of all decks will be made and the following permanent files are replaced:

```
APLLIB    APLPL    APL
AWSFIX    AFIFIX   AOVFIX
```

If the IAF task is selected, an IAF-based partial assembly will be made of only those decks with *EDIT directives in the file IAFMODS and the following permanent files are replaced:

```
APLLIB    APLPL    APL
AWSFIX    AFIFIX   AOVFIX
```

The IAF task should not be selected if the user's system is time-sharing system-based.

If the SOURCE or IAF task is selected, the user should determine where the site's NOS system OPL file is located. This is the file containing the NOS common decks and macros. The APL assembly will use, by default, the following control statement to access this file:

```
ATTACH(OLDPL=OPL/UN=LIBRARY)
```

If the user's system OPL file has a name different than OPL, or a storage location different than user name LIBRARY, these defaults can be changed by using the following directives with the replacements for MYOPL and MYLIB:

```
=D 274
ATTACH(OLDPL=MYOPL/UN=MYLIB)
```

If the SOURCE or IAF task is selected, no further directives are allowed except those explained in step 4.

Step 4 - Select Directives to Change the Installation Defaults (Optional)

This optional step selects directives to change the installation defaults for character set, terminal type, and entry message.

To change the default character set and hardware terminal type defaults, include the following directives:

```
=DEFINE CHANGE               (Override default options.)
=D 104                       (Delete CALL, CHANGE
                             (OPT1=64,OPT2=TYPEP).)
CALL,CHANGE(OPT1=cs,OPT2=ttopt)
                             (Both OPT1 and OPT2 must
                             be specified.)
where:    cs=64 (default)    (Character set is 64.)
            =63              (Character set is 63.)
          ttopt=APLAS        (Corresponds to the ASCII
                             APL print definition in the
                             NOS Reference Manual,
                             Volume I, page I-A-2.)
```

=TYPEP (default)	(ASCII APL typewriter pairing.)
=BITPR	(ASCII APL bit pairing.)
=ASCII	(Full ASCII (not APL).)
=TTY33	(Teletype Model 33 compatible.)
=BATCH	(CDC 512 Printer with ASCII print chain.)
=BH501	(CDC 501 Printer.)
=TTY38	(ASCII APL Teletype Model 38 model numbers 3841/4EA, 3841/4EG, 3851/6JA and 3851/6JG.)
=CORRE	(2741 compatible terminal. User must use TERM, CORAPL or equivalent before entering APL.)
=CD713	(Full ASCII-lowercase alphas equate to uppercase alphas.)

Note that although TYPEP is the default terminal type, many installations may be better off with APLAS, BITPR or CORRE. Consult your local APL analyst as to the most used type of APL terminal at your site.

NOTE

> Even if a different default for the terminal type is preferred, it is not necessary to make the change since the APL user can specify the desired terminal type on the APL control card.

To change the APL entry message that is displayed to users, include the following directives (the default is no entry message):

```
=DEFINE CHANGE
=DEFINE ENTRY
=D 322                    (Delete ENTER )LOAD *APL1
                          APLNEWS.)
ENTRY MESSAGE OF YOUR CHOICE
                          (Insert the desired message
                          here - 80 character
                          maximum.)
```

To delete the APL entry message, include the following directives:

```
=DEFINE CHANGE
=DEFINE NOENTRY           (Default is no entry message.)
```

The CHANGE directive need only be specified once if you want to change both the entry message and default options. However, all other directives are still needed.

The CHANGE directives, if specified, must be specified for all EXTRACT runs. That is, the 64-character set, TYPEP terminal type, and no entry message defaults are used unless otherwise specified by CHANGE directives.

The CALL,CHANGE directive works by setting up a call to the program AOVFIX with the appropriate options. The default AOVFIX program statement is as follows:

```
AOVFIX(C=64, L=OUT, FO=WSIO, TT=TYPEP)
```

In general, the AOVFIX parameters are defined as follows:

```
AOVFIX(C=cs, L=list, FO=WSIO, TT=ttopt, M=mfile, B=ovls)
```

where:

cs	Defined in the CALL,CHANGE directive.
ttopt	Defined in the CALL,CHANGE directive.
list	The local file name to which the AOVFIX listing output is written.
mfile	The local file name containing the entry message.
ovls	The local file name to which the ready-to-execute APL interpreter is written.

The FO=WSIO parameter must always be specified. If the M=mfile parameter is left out, no entry message will be generated which is what the =DEFINE NOENTRY directive does. If just M is specified, the file INPUT is used. If the B=ovls parameter is left out, AOVFIX writes the new interpreter to the file APLOVLS by default.

Step 5 - Run the EXTRACT Job

The EXTRACT job automatically creates and submits another job called APL0. The main function of job APL0 is to create permanent files containing the APL interpreter and associated files under user name APL0. If the APL0 job successfully completes and the BINARY, SOURCE, or IAF tasks were selected, job APL0, in turn, automatically creates and submits a job called APL1. The primary function of the APL1 job is to install the APL public workspaces under user name APL1. When this is done, job APL1 automatically runs the APL batch test to provide output to be used when verifying the APL installation (see step 7).

Run an EXTRACT job similar to the following with the desired installation options selected:

```
EXTRACT.
USER.            (Any user name, password, and
                 family may be used here.)
CHARGE.          (Insert charge card, if required.)
LABEL(OLDPL, F=I, L=APL2,
 ⎧ D=HY ⎫                    (7-track, 800 bpi.)
 ⎨ D=HD ⎬ VSN=REL60, PO=R)   (9-track, 800 cpi.)
 ⎩ D=PE ⎭                    (9-track, 1600 cpi.)
UPDATE(F, *==, C, D, 8)
UNLOAD(OLDPL)
SUBMIT(COMPILE)
REWIND(COMPILE)
COPYSBF(COMPILE)
7/8/9
```

```
=IDENT USERS    (Include with all EXTRACT jobs.)
=D APLINST.3,4
USER(APL0,password,family)
                (Insert APL0 password and family.)
CHARGE.         (Insert charge card, if required.)
=D 133,134
USER(APL1,password,family)
                (Insert APL1 password and family.)
CHARGE.         (Insert charge card, if required.)
=/ END OF USERS
=IDENT OPTIONS
|
Insert your options or any of option sets listed in
Table I-3-1.
↓
6/7/8/9
```

TABLE I-3-1. TYPICAL EXTRACT JOB DIRECTIVE SETS

Install time-sharing system-based APL test binaries using normal defaults from a 9-track, 800 cpi tape:

```
=/ EXAMPLE OPTION SET 1.
=DEFINE BINARY
=DEFINE HD
=DEFINE APLDEBG
=/ END OF SET 1.
```

Install time-sharing system-based production binaries from a 9-track, 1600 cpi tape and change the entry message:

```
=/ EXAMPLE OPTION SET 2.
=DEFINE PE
=DEFINE BINARY
=DEFINE CHANGE
=DEFINE ENTRY
=DEFINE APLPROD
=D 322
CHECK CATALOG FOR LATEST LIBRARY CHANGES.
=/ END OF SET 2.
```

Compile only IAF modifications and install APL as an IAF-based test system from a 7-track, 800 bpi tape:

```
=/ EXAMPLE OPTION SET 3.
=DEFINE HY
=DEFINE IAF
=DEFINE APLDEBG
=/ END OF SET 3.
```

Compile the entire time-sharing system-based APL system changing the terminal type default from TYPEP to BITPR and change the name of the NOS common decks file from OPL to SYSPL.

```
=/ EXAMPLE OPTION SET 4.
=DEFINE PE
=DEFINE SOURCE
=DEFINE CHANGE
=DEFINE NOENTRY
=DEFINE APLDEBG
=D 274
ATTACH(OLDPL=SYSPL/UN=LIBRARY)
=D 104
CALL,CHANGE(OPT1=64,OPT2=BITPR)
=/ END OF SET 4.
```

Step 6 - Install the APL Loader

The final step in installing APL is to put the APL loader program on the deadstart tape. The loader is stored as private indirect access file called APL under user name APL0. It contains the entry points APL and APLUM which enable it to be accessed as a control statement under either name.

To create a new deadstart tape, run a job similar to the following (the LIBEDIT directives may be different for your system):

```
MAKEDST.
USER(APL0,password,family)
CHARGE.          (Insert charge card, if required.)
VSN(TAPE=tapenumber)
                 (Insert volume serial number for
                 new deadstart tape.)
GET(APL)
COMMON(SYSTEM)
LIBEDIT(P=SYSTEM,B=APL,L=1,N=NEWSYS)
REQUEST(TAPE,NT,PE,F=I,LB=KU,PO=W)
COPY(NEWSYS,TAPE,V)
UNLOAD(TAPE)
7/8/9
*DELETE ABS/APLSYS
*BEFORE ABS/BASIC,ABS/APL
                 (LIBEDIT directives.)
6/7/8/9
```

The NOS system should now be deadstarted again, using the newly created deadstart tape. (Refer to the NOS Installation Handbook, for the deadstart procedure.)

Step 7 - Verify the APL Installation

If the BINARY, SOURCE, or IAF EXTRACT job directive is selected, the APL batch test will be run automatically. The DAYFILE of job APL1 should show the message "*END OF VERIFY TEST" if the test ran to completion.

The output from this test should be visually compared with file 6 (TAPLOUT) of the release tape. Any discrepancies may mean that the user has installed this version of APL on the wrong level of the operating system or may mean that the system has local modifications that are not compatible with APL.

The user may also wish to verify that APL runs correctly interactively. The following procedure can be run on any non-APL ASCII terminal:

Log on to an interactive ASCII terminal using a user name other than APL0 or APL1.

Enter the command (where CR means hit the carriage return key):

```
APL,WS=APLNEWS,UN=APL1,TT=713 CR
```

In response to the question WHAT DO YOU DESIRE? enter:

```
ALL CR
```

When the printing stops, enter:

```
    CR
    )SYSTEM CR
```

This will exit APL and will return the user to the NOS command level.

Although the APL batch test is run automatically during installation, the user may use the following job to run the test again without having to reinstall:

```
VERIFY.RUN APL BATCH TEST
USER(APL1,password,family)
                   (Run under user name APL1.)
CHARGE.            (Insert charge card, if required.)
* VERIFY APL INSTALLATION
ATTACH(TAPLTST/UN=APL0)
COPYBR(INPUT,TAPLDIR)
PURGE(OUTFILE/NA)
DEFINE(OUTFILE)
SETTL(4000)
APL(I=TAPLTST,L=OUTFILE,TT=BATCH,LO=EPB,
  DEBUG)
RETURN(TAPLTST,OUTFILE)
REWIND(TAPLDIR)
APL(I=TAPLDIR,TT=BATCH,LO=PB,MX=100000B,
  DEBUG)
* END OF VERIFY TEST
EXIT.
REWIND(OUTFILE)
COPY(OUTFILE)
PURGE(OUTFILE,CLEARWS,TSTEMP,TEST1,
  TEST2/NA)
PURGE(COPYTST,PWTEST,ABC/NA)
7/8/9
)LOAD TAPLWS
OUT $QTOUTFILE$QT
)DROP OUTFILE
6/7/8/9
```

Note that the DEBUG parameter on the APL control statements should be included only if the user wishes to test file APLDEBG under user name APL0. File APLPROD is tested if the DEBUG parameter is omitted.

RELEASE TAPE FORMAT

The APL2 installation tape format is as follows:

File 1 APLINST - OLDPL - contains all decks necessary to install and maintain the interpreter under system library *APL0 and the public system oriented workspaces under library *APL1.

File 2 APLPROD(pfn) or APLDEBG(pfn) or APLOVLS(lfn), OVL, and others - contain the absolute binary of the APL system set up for use (AOVFIXed). The released system has been generated for time-sharing usage and batch usage with the 64 character set. The default translation is APL-ASCII Typewriter-Pairing. If an IAF-based system is desired, see the =DEFINE IAF directive in step 3.

File 3 APLLIB - REL and TEXT - contains the relocatable binaries of the APL system along with their associated overlay directives. This includes the APL loader, the AWSFIX program (which formats sequential workspace files into the form APL understands), the AFIFIX program (which formats sequential APL structured files into the form APL understands), and the AOVFIX program (which replaces the first record of APLPROD with user selected system defaults).

File 4 APLPL - OPL - contains the source of the APL system in MODIFY format.

File 5 TAPLTST - TEXT - contains negative and positive tests for verifying APL after it has been installed on the user's system. The EXTRACT job automatically runs this test after installation. It tests most major features of the APL system.

File 6 TAPLOUT - TEXT - contains sample demo output from running the TAPLTST which was described previously.

File 7 APLWS - workspaces - contains the following records:

Record 1 CLEARWS - workspace
This workspace is supplied for compatibility with the APL reference manual and for $QDCOPY. Note that a)CLEAR command or $QDLOAD'*APL0 CLEARWS' no longer accesses this workspace but causes an overlay to be loaded instead.

Record 2 EMPTY - workspace
See Section 12 of the APL Reference Manual.

Record 3 APLNEWS - workspace
When accessed, this workspace gives the user news about the APL system. See Section 11 of the APL Reference Manual.

Record 4 CATALOG - workspace
When accessed, this workspace guides the user to the various public libraries of APL workspaces that are installed on the user's system. See Section 11 of the APL Reference Manual.

Record 5 WSFNS - workspace
Contains functions of general interest.

Record 6 FILESYS - workspace
Contains convenient functions for use with the APL file system. See Section 10 of the APL reference manual.

Record 7 FILES2 - workspace
Contains more complex functions for use with the APL file system. See Section 10 of the APL reference manual.

Record 8 TAPLWS - workspace
Contains all functions used when running the APL test; see File 5.

File 8 NEWSF - APL Random Access File - contains the news file which is used by the APLNEWS workspace. See File 7, Record 3.

File 9 IAFMODS TEXT - contains the modsets that are used to create an IAF-based version of the APL system and are intended for usage with NOS 1.3. These modsets are automatically applied to the APLPL file when the =DEFINE IAF directive is included during the running of the EXTRACT job. See step 3.

GENERAL DESCRIPTION

APT IV is a numerical control language processor designed to generate cutter location coordinates. The user has the capability to fully describe the part to be machined and then drive the tool along the chosen path. The output from APT IV is a verification listing and a file for input to a post processor. APT IV V2 has all the capabilities of APT IV V1 in addition to new language features and extensions.

APT IV V2 runs under NOS 1 on a Control Data host computer having at least 65,000 words of central memory.

RELEASE MATERIALS

APT IV V2 for NOS 1 is contained on release tape PL59. The structure of PL59 is as follows:

File 1 - contains the source library in UPDATE format. The first 67 decks, TAPEOP through ZSSPICT are common decks called by one or more of the succeeding decks. The next 16 decks, LLDM through LBCDBIN, are the LOAD COMPLEX. The next deck, LLCDATA, is data used by the LOAD COMPLEX routines. The next 93 decks, TAPT4T through TLCANON, are the TRANSLATOR. The next 378 decks, XAPT4E through XTRANSF57 are the EXECUTION COMPLEX. The next 45 decks, CAPT4C through CTAPEWT are the CLEDITOR. The last 18 decks, PAPT4P through PTAPEWT are the POST EXEC.

File 2 - contains the relocatable binary records of all routines needed for the STANDARD mode. It includes the binary records, COD4, PRD4, PTR4, POO4, and COO4, produced by the LOAD COMPLEX. This file is ready to be loaded by the host computer system and will produce the absolute modules contained on file 3.

File 3 - contains the absolute modules of the APT IV V2 system. The file should be placed on the system library if the user does not wish to modify the APT IV V2 processor. File 2 of the release tape is used to make file 3.

File 4 - contains the decks to install APT IV V2 on the host computer system.

File 5 - contains three identical part programs. Job control cards are provided to verify correct operation with system overlays resident in the system library, permanent files, or user library. User libraries with absolute modules are not currently available on NOS 1.

INSTALLATION REQUIREMENTS

The system has four general modes of operation:

1. Standard mode - offers all capabilities except parametric and sculptured surfaces. This mode requires approximately $110,000_8$ words of field length to build the system and for execution;
2. Parametric surface mode - offers all capabilities of the standard system plus the added features of parametric surface processing. This mode does not include sculptured surface processing. Approximately $110,000_8$ words are required to build the system and ($110,000_8$ plus the length of the FORTRAN parametric surface routines) words for execution;
3. Sculptured surfaces mode - offers all capabilities of the standard system plus the added features of sculptured surface processing. This mode does not include parametric surface processing. Approximately $125,000_8$ words are required to build and execute the system;
4. Parametric and sculptured mode - both parametric and sculptured surface processing may be used. Approximately $125,000_8$ words are required to build the system and ($125,000_8$ plus the length of of the FORTRAN parametric surface routine) words for execution.

INSTALLATION OPTIONS

The user may choose one of four modes:

1. standard
2. parametric surface
3. sculptured surface
4. parametric and sculptured surface

The release system operates in Standard mode.

In addition to the above modes, the user may choose one of three residences for the APT IV system. These are:

1. system library
2. permanent files
3. user library

The released system operates from the system library.

INSTALLATION PROCEDURES

The following control cards may be used to obtain the verification and installation decks from PL59:

```
IBGET,MT1.
LABEL(GET,R,L=APT4*RELEASE)
   where RELEASE = NOS1PO
SKIPF(GET,3)
COPYBF(GET,PUNCH,2)
6/7/8/9
```

The following provides a description of all the installation decks included in file 4. The job name (MAPT41 - MAPT46) identifies the deck.

- Install APT IV from file 3 of the release tape.

  Deck MAPT41 installs APT IV into the host computer operating system from file 3 of the release tape. The first two files are skipped and the absolute modules of file 3 are copied to input files for the system library program.

- Install APT IV from file 2 of the release tape.

  Deck MAPT42 may be used to make modifications to any routines except the LOAD COMPLEX. The modified routines are exchanged with their counterparts on the supplied relocatable file, and the resultant file is installed on the host computer system.

- Install APT IV from file 1 of the release tape.

  Deck MAPT43 and MAPT44 may be used to build the entire APT IV system from the source file. This will be necessary if the user wishes to install the sculptured surface and/or parametric surface mode of operation. The LOAD COMPLEX routines are compiled and executed. The output from LOAD COMPLEX is five FORTRAN block data programs which are then compiled and the relocatable binaries are placed on the file LCBIN. The entire APT IV system (TAPT4T through PTAPEWT) is compiled and relocatable binaries are placed on APTBIN. The two files are merged to form the file APTLGO. This file is then input to the LOADER and the absolute overlay modules are placed in the overlay file APTOVL. The remainder of the control cards makes a new system release tape which may be used as input to deck MAPT41.

INSTALLATION OPTIONS

The user has the following two options:

1. The user must insert an UPDATE directive card *DEFINE SSSURF to cause the sculptured surface routines to be compiled into the system. This card is present in deck MAPT43.
2. The user must insert an UPDATE directive card *DEFINE PARSURF to cause the parametric surface routines to be compiled into the system. This card is present in deck MAPT43.

Install APT IV from file 2 with overlay residence on permanent files.

Deck MAPT45 may be used to modify the relocatable file so that the APT IV system overlays can reside on permanent files. This deck can be modified as noted with COMMENT cards for a relocatable file with or without sculptured surfaces defined. For purposes of this example, the standard and parametric modes are equivalent.

The permanent files are defined, UPDATE is run with the overlay load parameter in deck TSUPERVT set equal to 0, the routine BEGINT is compiled and replaced on the relocatable file. Each overlay deck is compiled so that the overlay will be placed on a separate file. The loader then produces absolute overlay modules on the permanent files which are saved for later use.

Install APT IV from file 2 with overlay residence on a user library.*

Deck MAPT46 may be used to modify the relocatable file so that the APT IV system overlays can reside on a user library. The deck can be modified as noted with COMMENT cards for a relocatable file with or without sculptured surfaces defined. For purposes of this discussion, the standard and parametric modes are equivalent.

UPDATE is run with the overlay load parameter in deck TSUPERVT set equal to 7LAPT4OVL. The routine BEGINT is compiled and replaced on the relocatable file. The loader then produces absolute overlay modules on the file AFTOVL which is used as input to the EDITLIB program. EDITLIB then creates a user library on file APT4OVL.

Install APT IV from file 2 with overlay residence on the system library.

Deck MAPT42 may be used to install the system overlays since the release file has the overlay load parameter in deck TSUPERVT set equal to 6LSYSTEM.

LOAD COMPLEX EXECUTION

Deck MAPT47 may be used to modify the five relocatable decks produced by the LOAD COMPLEX. The names of the decks are COD4, PRD4, PTR4, POO4, and GOO4. Once the user modifications are determined and inserted, deck MAPT47 compiles and executes LOAD COMPLEX routines LLDM through LBCDBIN.

The deck LCCDATA, which has been modified by the user, is the input data for the LOAD COMPLEX execution. The output, five FORTRAN BLOCK DATA routines, is then compiled and the relocatable decks replace their counterparts on file 2 of the system release tape. Deck MAPT42 may then be used to install the system.

*Not currently available on NOS 1.

VERIFICATION

Deck MAPT48, MAPT49, or MAPT50 from file 5 should be used to verify that APT IV V2 has been correctly installed. Deck MAPT48 is used when the system overlays reside on the host computer system library. Deck MAPT49 is used when the system overlays reside on permanent files. Deck MAPT50 is used when the system overlays reside on a user library.

RELEASE DESCRIPTION

HARDWARE REQUIREMENTS

A minimum field length of 100K octal is required to load GPSS V 1.2. Once loaded, it requires a minimum field length of 45K octal to execute. Other requirements are the same as the minimum hardware configuration for NOS 1.

RELEASE MATERIALS

The release materials for GPSS V Version 1.2 are included on the release tape REL50. The release tape contains six files:

File 1 - GPSS V installation job decks GPS1 and GPS2, also the verification job deck VGPS in UPDATE format. The master control character for this UPDATE file is an equal sign (=).

File 2 - GPSS V preloaded absolute binaries.

File 3 - GPSS V relocatable binaries.

File 4 - GPSS V program library in UPDATE format.

File 5 - sample GPSS V model which is used as input by the verification deck VGPS.

File 6 - output from verification job VGPS.

INSTALLATION PROCEDURES

File 1 of REL50 contains the installation decks GPS1, GPS2, and the verification deck VGPS.

GPS1 updates the GPSS V program library (file 4), reassembles the system, and creates a new REL50 in the same format as the original release REL50. GPS1 may be used to configure the original release REL50 for a particular installation and also to install PSR corrective code.

GPS2 installs the preloaded absolute binaries on a direct access permanent file named GPSS. GPSS V is intended to run from a local file. It cannot be executed from the running system. The local file name for GPSS V must be GPSS.

VGPS attaches the direct access permanent file GPSS and executes the model on file 5 of REL50. Output from VGPS should be similar to the output on file 6 of REL50.

GPS1, GPS2, or VGPS may be obtained from REL50 by executing a job of the following form:

```
EXTRACT,CM45000,T100.
USER.             (Supply valid USER card.)
VSN(TAPE=REL50)
                                     ( HY )
LABEL(TAPE,L=GPSSV1P2,F=SI,D=        { HD } )
                                     ( PE )
UPDATE(Q,P=TAPE,*==,C=PUNCH)
7/8/9
=DEFINE X
=/ UPDATE CARDS TO MODIFY INSTALL DECKS
   GO HERE.
=C Y
6/7/8/9
```

where:

X = HY or HD or PE
Y = the deck name

HY selects 800 bpi 7-track tapes
HD selects 800 cpi 9-track tapes
PE selects 1600 cpi 9-track tapes

If a card punch is not available, or if the user desires to execute either GPS1, GPS2, or VGPS directly from REL50, the following two cards should be substituted for the UPDATE card in the previous deck:

```
UPDATE(Q,P=TAPE,*==)
SUBMIT(COMPILE,B)
```

In order to list GPS1, GPS2, or VGPS as they appear on REL50, a job of the following form should be executed:

```
LIST,CM45000,T100.
USER.             (Supply valid USER card.)
VSN(TAPE=REL50)
                                     ( HY )
LABEL(TAPE,FI=GPSSV1P2,F=SI,D=       { HD } )
                                     ( PE )
UPDATE(Q,L=A12347,*==,P=TAPE)
7/8/9
=C GPS1
=C GPS2
=C VGPS
6/7/8/9
```

INSTALLATION PARAMETERS

For maximum efficiency, several installation options are provided to configure GPSS V to individual installations. To select an option, an update *DEFINE option name card should be included in the configuration run. Options are:

Option — Definition

CRM — All I/O requests will be processed by the CYBER record manager. If not selected, standard CIO requests will be used. CYBER record manager will be used for I/O on the following files: INPUT, OUTPUT, JOBTA1, JOBTA2, and JOBTA3. All internal file operations on files not available to the HELP block user still use CIO directly for speed purposes.

CP64 — Code will be optimized for a 6400 type processor. If not selected, a 6600 type processor is assumed.

XJOP — Monitor requests will be executed via the XJ instruction. If not selected, the XJ instruction is not used.

IMUL — Integer multiply hardware is installed.

FILELOAD — If defined, the preloader will create a random file containing the overlays. This version of GPSS cannot be placed on the system, but must be run from a permanent file. However, the field length required to start a GPSS run is reduced to what is needed to load the (0,0) overlay plus the longest primary overlay, which is about 30K instead of approximately 75 to 105K when FILELOAD is not defined. No ECS load options are available when FILELOAD is defined.

REPRIEVE — If defined, the error termination printout section is executed when GPSS execution is halted due to operator drop, exceeding CP or I/O time limit, or exceeding mass storage limit. Thus, the results of the simulation up to the point at which GPSS execution is halted for one of the above reasons is printed. If REPRIEVE is not defined, GPSS execution is halted without an error termination printout.

FTRACE — If defined, then in addition to the standard one-line trace of GPSS V/6000, the current transaction is also listed. This option slows down a simulation which uses the TRACE feature by a factor of 2 to 10 depending upon the amount of tracing done. It also increases the minimum field length for all runs by 2 to 3 K.

Option — Definition

LISALL — If defined, LIST and UNLIST cards are listed on the output file when encountered. Otherwise, they are never listed.

CMR, XJOP, IMUL, and REPRIEVE are included by default in the install deck GPS1.

Other parameters that may be changed for an installation are the operating system name and installation name. Cards of the following type may be used as input to update to change these parameters:

```
*D,GPSSTXT.419.GPSSTXT.420
 OPSYSTEM MICRO  1,,$NOS 1.X$
 INSTALL     MICRO  1,,$installation name$
```

CONFIGURATION RUN

A job of the following form may be run to generate a new REL50 tape which is configured for a particular installation.

```
CONFIG.CM45000,T100.
USER.              (Supply valid USER card.)
VSN(TAPE=REL50)
                                          ⎧ HY ⎫
LABEL(TAPE,FI=GPSSV1P2,F=SI,D=  ⎨ HD ⎬)
                                          ⎩ PE ⎭
UPDATE(Q,*==,P=TAPE)
SUBMIT(COMPILE,B)
7/8/9
=DEFINE X
=D,GPS1.3
=/ USER CARD GOES HERE
=I,GPS1.50
*D,GPSSTXT.419,GPSSTXT.420
 OPSYSTEM MICRO  1,,$NOS 1.X$
 INSTALL     MICRO  1,,$installation name$
  LENGTH
=C,GPS1
6/7/8/9
```

where:

X = HY or HD or PE

PSR MODIFICATION RUN

A job of the following form may be run to update GPSS V with PSR corrective code and generate a new REL50. The PSR corrective code should be placed on a direct access file name USER previous to this run.

```
PSR,CM45000,T100.
USER.            (Supply valid USER card.)
VSN(TAPE=REL50)
                                 HY
LABEL(TAPE,FI=GPSSV1P2,F=SI,D= { HD })
                                 PE
UPDATE(Q,*==,P=TAPE)
SUBMIT(COMPILE,B)
7/8/9
=DEFINE X
=D,GPS1.3
=/ USER.         (Supply valid USER card.)
=I,GPS1.10
ATTACH(USER)
=D,GPS1.44,GPS1.51
*READ USER
=C,GPS1
6/7/8/9
```

where:

X = HY or HD or PE

NOTES AND CAUTIONS

The deck GPSSTXT on the source OLDPL (file 4) of REL50 contains documentation which is an overview of the system. It is not required but may be helpful for the user to obtain a listing of GPSSTXT by assemblying it using COMPASS.

When loading GPSS V, five nonfatal errors are noted. These errors are of the form:

DUPLICATE ENTRY POINT NAME - INIT

Errors of any other form should be investigated.

RELEASE DESCRIPTION

IMSL 6 (International Mathematical and Statistical Library, Version 6) consists of more than 400 FORTRAN functions and subprograms. IMSL 6 is on REL63.

HARDWARE REQUIREMENTS

IMSL 6 requires the minimum hardware configuration for NOS.

NOTES AND CAUTIONS

The installation process does not provide program listings. If a program listing is needed (it is very long), the FTN card in the install deck may be appropriately modified (remove L=0).

RELEASE MATERIALS

IMSL 6 is contained on the tape known as REL63. REL63 is a program library in UPDATE format. It has the following characteristics: 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, F=SI, seven files:

File 1 - install deck in UPDATE OLDPL format,

File 2 - empty file,

File 3 - empty file,

File 4 - program deck in UPDATE OLDPL format,

File 5 - verify job control language deck in UPDATE OLDPL format,

File 6 - expected verify output ((COPYBF,OUTPUT) format), and

File 7 - verify program deck in UPDATE OLDPL format.

INSTALLATION PROCEDURES

The procedure for the installation of IMSL 6 is outlined in the following four steps.

Step 1. Generate Essential Listings

This step includes:

- Installation deck,
- Verification deck, and
- Expected verify output.

NOTE

The user must supply appropriate job and accounting cards.

```
JOB.
USER(*,*)
CHARGE(*,*)
               ( HY )
LABEL(TAPE,D=  { HD }  ,F=SI,VSN=IMSLIB)
               ( PE )
COPYBF(TAPE,OLDPL,1)
REWIND(OLDPL)
UPDATE(F,D,8,*=/,L=7)
REWIND(TAPE,OLDPL)
SKIPF(TAPE,4)
COPYBF(TAPE,OLDPL,1)
REWIND(TAPE,OLDPL)
UPDATE(F,D,8,*=/,L=7)
SKIPF(TAPE,5)
COPYBF(TAPE,OUTPUT,1)
6/7/8/9
```

Step 2. Install IMSL 6

Installation characteristics are:

- Direct access permanent file,
- Permanent file name = IMSLIB,
- Public access rights, and
- User number of the installer.

```
JOB.
USER(*,*)
CHARGE(*,*)
               ( HY )
LABEL(TAPE,D=  { HD } ,F=SI,VSN=IMSLIB)
               ( PE )
REWIND(TAPE,OLDPL)
COPYBF(TAPE,OLDPL)
REWIND(OLDPL)
UPDATE(F,D,8,*=/)
SUBMIT(COMPILE,B)
7/8/9
/IDENT  IMSL       (Optional.)
/DEFINE NOS        (Required.)
```

Note that one of the following three tape parameters is required.

```
/DEFINE MT          (7-track, 800 bpi.)
/DEFINE NT          (9-track, 800 cpi.)
/DEFINE NTPE        (9-track, 1600 cpi.)

/DELETE IMSLIL.XX,YY            (Required.)
USER(*,*)           (Supply correct data for these
                    cards.)
CHARGE(*,*)

6/7/8/9
```

Step 3. Run Verification Tests

Refer to the following.

```
JOB.
USER(*,*)
CHARGE(*,*)
                  ⎧ HY ⎫
LABEL(TAPE,D=     ⎨ HD ⎬  ,F=SI,VSN=IMSLIB)
                  ⎩ PE ⎭
REWIND(TAPE,OLDPL)
SKIPF(TAPE,4)
COPYBF(TAPE,OLDPL)
REWIND(OLDPL)
UPDATE(F,D,8,*=/)
SUBMIT(COMPILE,B)
7/8/9

/IDENT IMSL         (Optional.)
/DEFINE NOS         (Required.)
```

Note that one of the following three tape parameters is required.

```
/DEFINE MT          (7-track, 800 bpi.)
/DEFINE NT          (9-track, 800 cpi.)
/DEFINE NTPE        (9-track, 1600 cpi.)

/DELETE IMSLVL.XX,YY            (Required.)
USER(*,*)           (Supply correct data for these
                    cards.)
CHARGE(*,*)

6/7/8/9
```

Step 4. Verification

Do a minimal visual comparision of the expected verify output and the actual verify output.

RELEASE DESCRIPTION

LCGT/IGS V2 (Low Cost Graphics Terminal Interactive Graphics System Version 2) runs under NOS and Telex to provide both a remote interactive graphic capability in graphic mode and timesharing capability in interactive mode.

The LCGT/IGS subroutines are callable from user application programs only under FORTRAN Extended, and provide the user with the capability of interacting with his program from the console using the keyboard and cross-hairs.

The operation of LCGT/IGS V2 requires the installation of Telex and all other associated products with it under NOS.

HARDWARE CONFIGURATION

The minimum hardware configuration required by NOS and Telex is required to support the LCGT terminal. LCGT supports only asynchronous communications under NOS.

NOTES AND CAUTIONS

All limitations applicable to NOS and Telex also apply to LCGT/IGS V2.

All subroutines in LCGT/IGS V2 are callable from user application programs in FORTRAN Extended only.

Under Telex, the LCGT operates asynchronously in a TTY compatible mode at 110 or 300 bauds using a 103A modem.

RELEASE MATERIALS

LCGT/IGS is contained on the tape known as REL56A.

REL56A is a program library in UPDATE format. It has the following characteristics: labeled, 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, LCGTIGSV2 as file ID in HDR1 label, F=SI, six files:

File 1 - installation job program library,

File 2 - empty,

File 3 - empty,

File 4 - program library of LCGT/IGS in UPDATE format,

File 5 - verification Job Program Library, and

File 6 - empty.

INSTALLATION PROCEDURES

The installation of LCGT/IGS V2 requires creation of a user library, LCGTVR2. The job required to install LCGT/IGS is released as an installation deck program library. Telex is required before attempting to utilize LCGT/IGS V2 but is not necessary for its installation.

The installation job is file 1 on the release tape. This job is a program library in UPDATE format with = (equals) as the master character. A dummy USER card is a part of this job. The user should first perform an UPDATE run and list the compile file. As a part of the EXTRACT procedure, the user should modify the JOB/ACCOUNT cards appropriately for his installation.

To obtain the installation job, execute a procedure of the form shown below:

```
EXTRACT,T10,CM60000.
USER.            (Supply valid USER card.)
LABEL(LCGT,R, F=SI, L=LCGTIGSV2,
        ( HY )
    D=  { HD }  ,VSN=LCGTV2)
        ( PE )
COPYBF(LCGT,OLDPL)
UPDATE(F,*==,C,D,8)
UNLOAD(LCGT)
SUBMIT(COMPILE)
7/8/9
=IDENT,XXX
=DEFINE,NOS
          ( MT   )  (7-track, 800 bpi.)
=DEFINE,  { NT   }  (9-track, 800 bpi.)
          ( NTPE )  (9-track, 1600 bpi.)
=D,LCGT1.yyy        (To change JOB and user cards
USER.                as needed.)
=COMPILE,LCGT1
6/7/8/9
```

File 5 is the verification job. This job is also in UPDATE format and may require changes in the JOB/USER cards of the deck. A preliminary UPDATE run should be made and this deck listed for inspection.

To save the binary of the verification job, the following XTRACT procedure can be used:

```
XTRACT,T10.
USER.              (Supply valid USER card.)
LABEL(LCGT,R,F=SI,L=LCGTIGSV2,
     { HY }
  D= { HD }   ,VSN=LCGTV2)
     { PE }
SKIPF(LCGT,4,17)
COPYBF(LCGT,OLDPL)
UPDATE(F,*==,C,D,8)
SUBMIT(COMPILE)
7/8/9
=IDENT,XX
=DEFINE,NOS
=D,LCGT2.yyy       (To change JOB and USER cards
USER.               as needed.)
=COMPILE,LCGT2
6/7/8/9
```

VERIFICATION PROCEDURE

The verification program, VRFY, is created as a permanent file as described earlier. This program can be run to verify the proper installation of LCGT/IGS as follows:

1. Follow the "Terminal Operating Procedures" to set up the terminal for asynchronous operation.

2. LOGIN with valid username and password, and proper CHARGE number.

3. Execute the following procedure:

   ```
   NEW,CONTROL  (CR)
   TEXT  (CR)
   ATTACH(VRFY)
   ATTACH(LCGTVR2)
   FILE(TAPE62,BT=I,RT=S)
   LDSET(LIB=LCGTVR2,FILES=TAPE62)
   (Break Key)  EXIT TEXT MODE.
   PACK  (CR)
   SAVE,CONTROL  (CR)
   -CONTROL or CALL,CONTROL
   ```

4. You will be asked to enter terminal type (figure I-7-1).

   1 = 4010 synchronous
   2 = 4010 asynchronous
   3 = 4014 synchronous
   4 = 4014 asynchronous
   5 = 4014 with EGM, synchronous
   6 = 4014 with EGM, asynchronous

5. The screen will be cleared and figure I-7-2 will be displayed on the screen.

6. Type in any number in the range 5 through 25 followed by (CR) .

7. Figure I-7-3 resulted by typing in 13.

8. Type BYE when finished.

TERMINAL OPERATING PROCEDURES

LCGT can properly communicate with the following two types of ports on a 6671 or 6676 multiplexer under Telex.

1. ASCII, asynchronous mode at 110 baud rate, and

2. ASCII, asynchronous mode at 300 baud rate.

To operate the terminal:

1. Turn the terminal power on by pressing the POWER switch to the ON position. The POWER lamp will come on red.

2. Set the ASCII/BCD switch to the ASCII position.

3. Set the baud rate switch to one of the following positions:

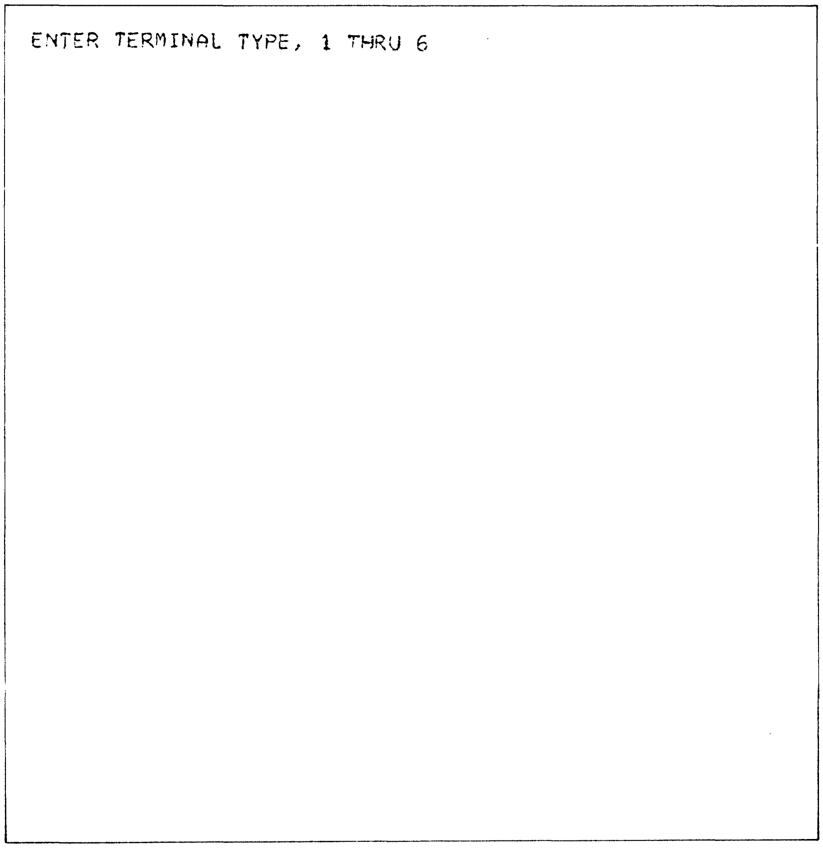
   - 300 position if asynchronous at 300-baud rate.

   - PROG position if asynchronous at 110-baud rate.

4. Turn the CODE EXPANDER switch OFF.

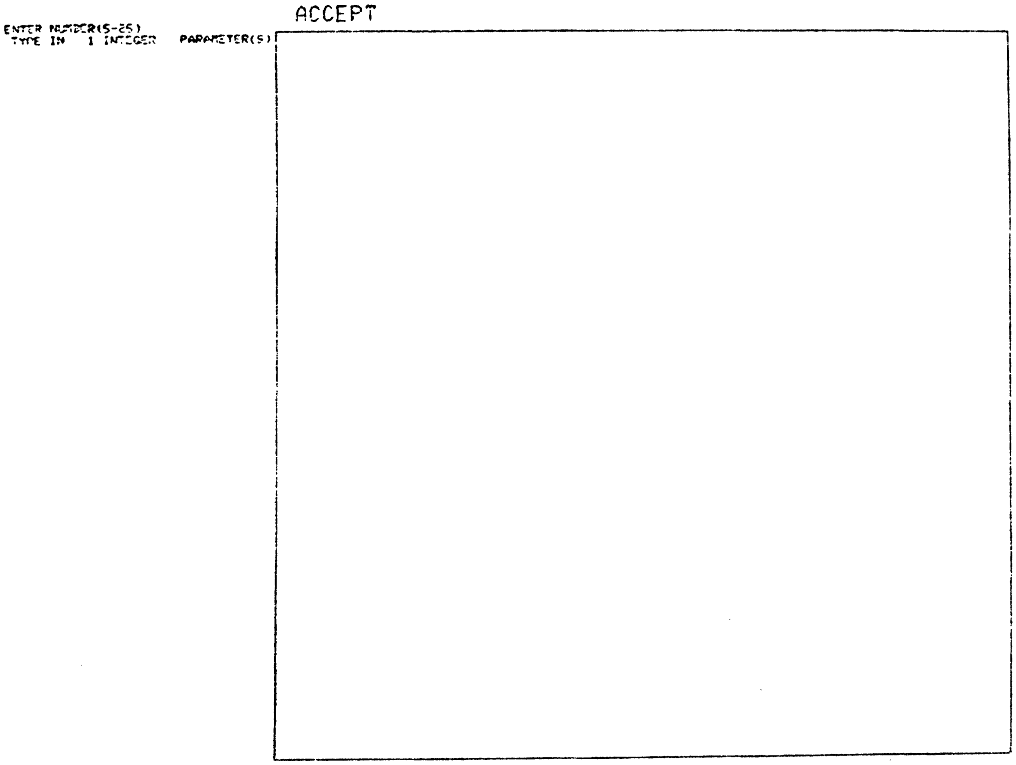
5. Dial the appropriate phone number.

6. Telex will then send a LOGIN message when it is ready for interactive command processing.



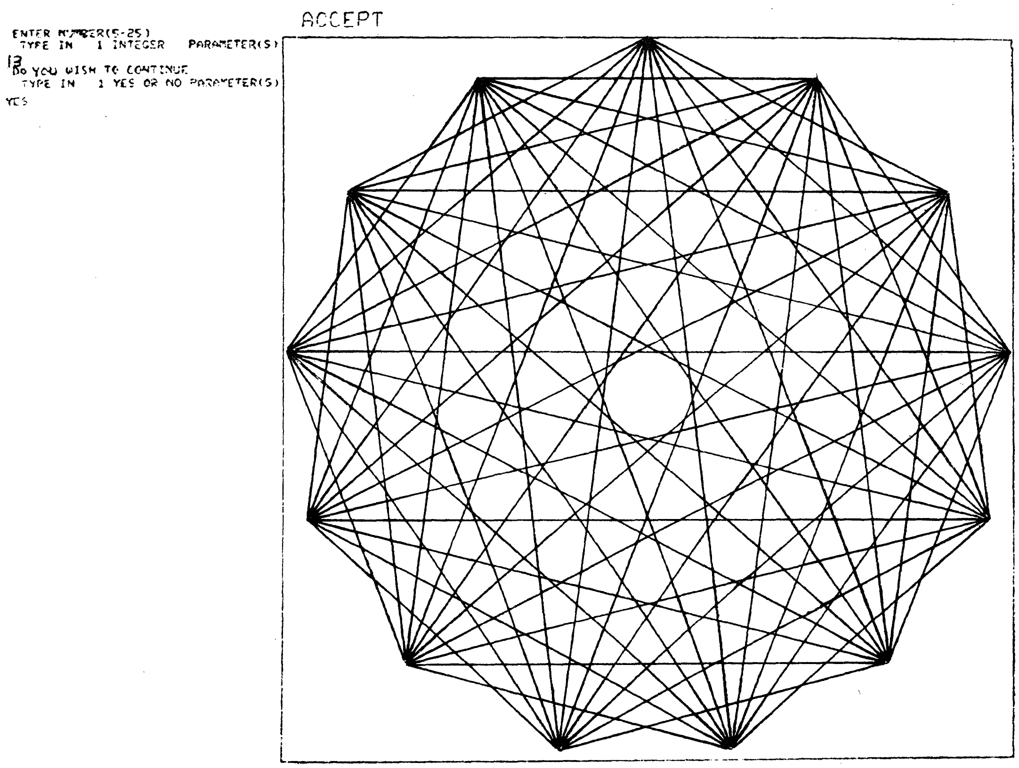
```
ENTER TERMINAL TYPE, 1 THRU 6
```

Figure I-7-1. Entering Terminal Type



```
                                    ACCEPT
ENTER NUMBER(5-25)
 TYPE IN   1 INTEGER   PARAMETER(S)
```

Figure I-7-2. Terminal's Response



```
ENTER NUMBER(5-25)
 TYPE IN   1 INTEGER   PARAMETER(S)
13
DO YOU WISH TO CONTINUE
 TYPE IN   1 YES OR NO PARAMETER(S)
YES
```

Figure I-7-3. Result of Typing in a 13

RELEASE DESCRIPTION

HARDWARE REQUIREMENTS

PERT/TIME requires up to three tape units and 65K central memory to run.

RELEASE MATERIALS

PERT/TIME is contained on release tape REL52A. REL52A is a program library in UPDATE format. It has the following characteristics: labeled, 7-track (800 bpi) or 9-track (800 or 1600 cpi), PERT2P0 as file ID in HDR1 label, six files:

File 1 - installation deck program library in UPDATE format,

File 2 - PERT/TIME in absolute binary format,

File 3 - PERT/TIME in relocatable binary format,

File 4 - source program library in UPDATE format,

File 5 - empty, and

File 6 - empty.

LIMITATIONS

No user file can be named TAPE1 through TAPE7 because these file names are used internally.

INSTALLATION PROCEDURES

Release tape REL52A contains PERT/TIME.

To get a listing and a punched-card copy of the installation decks from the first file of REL52A, run a job similar to the following (if a card punch is not on-line, the following decks should be keypunched):

```
JOB.
USER.              (Supply valid USER card.)
VSN(TAPE=REL52A)
LABEL(TAPE,FI=PERT2P0,F=I,
        ( HY )     ( MT )
   D=   { HD }  ,  {    }  )
        ( PE )     ( NT )
UPDATE(F,L=A12347,P=TAPE)
REWIND(COMPILE)
COPYBF(COMPILE,PUNCH)
7/8/9
6/7/8/9
```

The following job installs PERT/TIME executable binary on a permanent file:

```
JOB.
USER.              (Supply valid USER card.)
VSN(TAPE=REL52A)
LABEL(TAPE,FI=PERT2P0,F=I,
        ( HY )     ( MT )
   D=   { HD }  ,  {    }  )
        ( PE )     ( NT )
DEFINE(PERT78/CT=S,M=R)
SKIPF(TAPE,1)
COPYBF(TAPE,PERT78)
6/7/8/9
```

To execute from the release tape REL52A, run a job similar to the following:

```
PERT,T1000,CM10000.
USER(usernumber,password)
VSN(TAPE=REL52A)

                           ( MT )
LABEL(TAPE,FI=PERT2P0,     {    }  ,
                           ( NT )
       ( HY )
   D=  { HD }  ,F=I)
       ( PE )
SKIPF(TAPE,1)
COPYBF(TAPE,PERT78)
PERT78.
7/8/9
  .
  .             )
  .             }    PERT networks
  .             )
  .
6/7/8/9
```

If a master data tape is made, insert the following after the LABEL statement:

```
REQUEST,TAPE6,MT.
```

In an old master data tape is input and a new master data tape saved, insert the following after the LABEL statement:

```
REQUEST,TAPE4,MT.     OLD MASTER
REQUEST,TAPE6,MT.     NEW MASTER
```

If either option is used, insert a RESOURC statement before the REQUEST statement. An entry must be placed on the Y-card for each additional tape used.

Successive PERT/TIME networks may be processed by batching as follows (a higher FL will be required):

```
K
L
M
X
Y
W
A                    PERT BATCHED NETWORKS
•
•
•
A
Z
K
L
•
•
•
Z
```

(Input for each network starts with the K control statement and ends with the Z control statement.)

The following job compiles and installs PERT/TIME from the source PL to a permanent file.

```
JOB.
USER.              (Supply valid USER card.)
VSN(TAPE=REL52A)
                                            { MT }
LABEL(TAPE,FI=PERT2PO,F=I,                  { NT }  ,
           { HY }
    D=     { HD }  )
           { PE }
SKIPF(TAPE,3)
UPDATE(F,P=TAPE,R)
UNLOAD(TAPE)
REWIND(COMPILE)
DEFINE(PERT78/CT=S,M=R)
FTN(I=COMPILE,B=PERT,L=O)
LOAD(PERT)
NOGO(PERT78)
7/8/9
6/7/8/9
```

SIMSCRIPT II.5 VERSION 4.2

RELEASE DESCRIPTION

SIMSCRIPT II.5 runs under NOS and is designed to execute from permanent files.

HARDWARE REQUIREMENTS

The SIMSCRIPT II.5 system requires the same minimum hardware configuration as NOS. The compiler requires a minimum 60K octal field length; however, core requirements for compilation are dynamic and dependent on source program length. A 70K octal field length is sufficient for most programs.

DEFICIENCIES

None.

RELEASE MATERIALS

SIMSCRIPT II.5 Version 4.2 is contained on the tape known as REL64. REL64 has the following characteristics: 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, SIMII5V4P2 as file ID in the HDR1 label and six files:

File 1 - Installation deck INSTALL in UPDATE PL format,

File 2 - Absolute binaries of the SIMSCRIPT II.5 compiler,

File 3 - Relocatable binaries of SIMSCRIPT II.5 execution library and error messages,

File 4 - empty

File 5 - Verification deck in UPDATE PL format, and

File 6 - Sample output from verification run.

Listings of the installation deck and the verification program may be obtained by running the following job:

```
LIST,T50.
USER. (Supply valid USER statement.) (HY)
LABEL(OLDPL,R,L=SIMII5V4P2,D=        {HD} ,
  VSN=REL64.                         (PE)
UPDATE,F,*=  =,L=7.
SKIPF,OLDPL,4.
UPDATE,F,*=  =,L=7,R.
6/7/8/9
```

INSTALLATION PROCEDURES

Installation deck INSTALL on file 1 of REL64 installs the SIMSCRIPT II.5 binaries on four permanent files:

SIMII5 - SIMSCRIPT II.5 compiler,

SIM2LIB - SIMSCRIPT II.5 execution library,

SIMERRC - SIMSCRIPT II.5 compiler error message file, and

SIMERRE - SIMSCRIPT II.5 execution library error message file.

SIMII5 and SIM2LIB are direct access files. SIMERRC and SIMERRE are indirect access files.

The error message files must be SAVED under user name LIBRARY. This is because the SIMSCRIPT II.5 compiler or a running SIMSCRIPT II.5 program will attempt to GET these files internally from user name LIBRARY. If the error message files do not exist, or are on a different user name, an error number is printed in the job dayfile, but no error message is printed.

To execute INSTALL, a job of the following form should be run:

```
INST,T50.
USER. (Supply valid USER statement.) (HY)
LABEL,ØLDPL,R,L=SIMII5V4P2,D=       {HD} ,
  VSN=REL64.                        (PE)
UPDATE,Q,*=  =,D,8.
SUBMIT,CØMPILE,B.
7/8/9      (HY)
=DEFINE    {HD}
           (PE)
=DELETE INSTALL.3
USER. (Supply valid USER statement.)
=CØMPILE INSTALL
6/7/8/9
```

NOTES AND CAUTIONS

PERMANENT FILES

The installation deck on file 1 of REL64, INSTALL, does not check for existing permanent files with the names SIMII5, SIM2LIB, SIMERRC, or SIMERRE. If files already exist under these names, they must either be purged or renamed.

INSTALLATION VERIFICATION

The following job may be run to verify correct installation of the SIMSCRIPT II.5 system:

```
VERIFY,T50.
USER.  (Supply valid USER statement.)   (HY)
LABEL,OLDPL,R,L=SIMII5V4P2,D=           {HD}
  VSN=REL64.                            (PE)
SKIPF,OLDPL,4.
UPDATE,Q,D,8,*= =,R.
SUBMIT,COMPILE,B.
7/8/9
=D VERIFY.3
USER.  (Supply valid USER statement).
=C VERIFY
6/7/8/9
```

File 6 of REL64 can be copied to OUTPUT and compared with the output from the submitted job to verify correct execution.

TIGS 1 OVERVIEW

RELEASE DESCRIPTION

TIGS 1 (Terminal Independent Graphics System Version 1) runs under NOS and either a time-sharing system or NAM/IAF communications package to provide the user with both a remote interactive graphic capability in graphic mode and a time-sharing capability in interactive mode.

TIGS 1 consists of a preprocessor and a post processor. The preprocessor writes a neutral display file that is then traversed by the post processor to produce graphics output to a specific terminal.

The TIGS subroutines are callable from user application programs that are coded in FORTRAN Extended.

The operation of TIGS 1 under NOS requires the installation of either the Time-sharing module or NAM/IAF and all associated products.

HARDWARE CONFIGURATION

The minimum hardware configuration required for NOS and the communications package (either the Time-sharing Module or NAM/IAF) is required to support TIGS 1.

RELEASE MATERIALS

The TIGS 1 release requires both the preprocessor magnetic tape, REL62A, and an appropriate post processor tape. Refer to the individual post processor sections for installation, verification, and terminal operation of TIGS 1.

TIGS 1 PREPROCESSOR

The TIGS 1 preprocessor resides on the tape known as REL62A. REL62A has the following characteristics: 7-track (800-bits per inch (bpi)) or 9-track (800 or 1600-characters per inch (cpi)), binary recording mode, TIGSPREV1 as the file ID in the HDR1 label, and six files:

File 1 - empty,
File 2 - empty,
File 3 - empty,
File 4 - TIGS 1 preprocessor source code program library in UPDATE format,
File 5 - empty, and
File 6 - empty.

TEKTRONIX 401X POST PROCESSOR OPTION

RELEASE MATERIALS

The Tektronix 401X post processor runs under NOS when installed with the TIGS 1 preprocessor.

The Tektronix 401X post processor resides on the tape known as REL62B. REL62B is a program library in UPDATE format and has the following characteristics: 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, TIGSTEKV1 as file ID in HDR1 label, six files:

File 1 - installation job program library in UPDATE format,
File 2 - empty,
File 3 - empty,
File 4 - TIGS 1 Tektronix post processor source code program library in UPDATE format,
File 5 - verification job in UPDATE PL format, and
File 6 - empty.

NOTES AND CAUTIONS

All limitations applicable to NOS and the communications package, (the Time-sharing Module or NAM/IAF) also apply to TIGS 1 with the Tektronix 401X post processor.

All TIGS 1 subroutines are callable from application programs that have been written in FORTRAN Extended.

A CDC Synchronous interface (Tektronix product number 021-0135-01) is required on each Tektronix terminal to communicate in synchronous mode with the 255X host communications processor on NAM/IAF.

Communications between the terminal and the central site are supported for TIGS 1 with the Tektronix 401X post processor according to table I-10-1.

TABLE I-10-1. TIGS/TEKTRONIX COMMUNICATIONS CAPABILITIES

Communications Mode	Communications Package	Baud Rate	Compatible Modem Type
Synchronous	NAM/IAF	4800	208B
	NAM/IAF	2400	201B
	NAM/IAF	2000	201A
Asynchronous	NAM/IAF	1200	212A
	NAM/IAF and Time-sharing	300	103A

INSTALLATION DESCRIPTION

The installation of TIGS 1 with the Tektronix 401X post processor allows the user the option of generating either a segment/overlay loader library, TIGSLIB, or both a segment/overlay loader library, TIGSLIB, and a basic (nonsegment/nonoverlay) loader library, TIGSSFL.

The segment/overlay loader library, TIGSLIB, can be used in those situations where memory requirements of graphic applications exceed the available NOS field length. The user can employ the segment/overlay loader library, TIGSLIB, in conjunction with the CYBER loader to segment or to overlay a graphic application and thereby reduce memory requirements. Note that the use of the segment/overlay loader library, TIGSLIB, without segmentation or overlays greatly increases an application program field length requirement.

The basic (nonsegment/nonoverlay) loader library, TIGSSFL, can be used for graphic applications with memory requirements that do not exceed the NOS available field length.

The segment/overlay loader library, TIGSLIB, is installed by default by the installation job. Both the segment/overlay loader library, TIGSLIB, and the basic (nonsegment/nonoverlay) loader library will be installed by the installation job by specifying =DEFINE,BASICLDR as an UPDATE directive.

To provide optimum data output efficiency on NOS, the user must specify an output buffer size when installing TIGS 1 with the Tektronix 401X post processor libraries. Table I-10-2 lists the selectable output buffer sizes and the factors which influence buffer size selection.

TABLE I-10-2. OUTPUT BUFFER SIZES

Baud Rate	Condition	Output Buffer Size in Characters	Required Installation Job UPDATE Directive
300	Field length is important concern.	500	=DEFINE, NOS500
1200, 2000		1500	=DEFINE, NOS1500
1200, 2000, 2400	Much data is transmitted.	4000	=DEFINE, NOS4000
4800		8000	=DEFINE, NOS8000

In general, if field length is not a primary concern and the libraries are to be used in a 300-baud hardware configuration, then define an output buffer size of 1500 characters (=DEFINE, NOS1500). If the libraries are to be used in 1200-, 2000-, or 2400-baud hardware configurations, then specify an output buffer size of 4000 characters (=DEFINE, NOS4000). If the libraries are to be used in a 4800-baud hardware configuration, then specify an output buffer size of 8000 characters (=DEFINE, NOS8000).

INSTALLATION PROCEDURE

The procedure for installing TIGS 1 with the Tektronix 401X post processor is outlined in the following three steps.

Step 1

The installation job is file 1 on the Tektronix post processor release tape, (REL62B). File 1 is a program library in UPDATE format with = (equals) as the master control character.

A dummy USER card has been included as part of the installation job. Perform an UPDATE run against file 1 of REL62B and list the compile file to obtain the sequence number which corresponds to the USER card (and any other control cards) which must be modified for installation. A listing of the compile file may be obtained by executing the following procedure:

```
LISTI,T10,CM65000.
USER(*)          (Supply appropriate accounting
                 information.)
LABEL(TIGSPL,R,L=TIGSTEKV1,
      ⎧ HY ⎫                          (7-track, 800 bpi.)
   D= ⎨ HD ⎬ ,VSN=REL62B,F=SI)  (9-track, 800 cpi.)
      ⎩ PE ⎭                          (9-track, 1600 cpi.)
COPYBF(TIGSPL,OLDPL)
UNLOAD(TIGSPL)
UPDATE(F,U,*==,L=17)
COPYSBF(COMPILE,OUTPUT)
7/8/9
=IDENT,INSTALL
=DEFINE,NOS
           ⎧ HY ⎫                     (7-track, 800 bpi.)
=DEFINE,   ⎨ HD ⎬                     (9-track, 800 cpi.)
           ⎩ PE ⎭                     (9-track, 1600 cpi.)
=DEFINE,BASICLDR
                 (This is an optional UPDATE direc-
                 tive. If the user wishes to install
                 both a segment/overlay loader
                 library (TIGSLIB) and a basic (non-
                 segment/nonoverlay) loader library
                 (TIGSSFL), then this DEFINE state-
                 ment must be included. If this
                 DEFINE statement is not included,
                 then only the control cards required
                 to install a segment/overlay loader
                 library will be listed.)
           ⎧ NOS500  ⎫
=DEFINE,   ⎪ NOS1500 ⎪
           ⎨ NOS4000 ⎬
           ⎩ NOS8000 ⎭
                 (Specify an output buffer size on
                 NOS to improve the efficiency of
                 the output of data.)

6/7/8/9
```

Step 2

Perform a full UPDATE on file 1 of the Tektronix post processor release tape, (REL62B), modifying the installation job program library with the appropriate accounting information and DEFINE UPDATE directives required for installation. Route the resulting installation job to the NOS input queue by executing the following procedure. NOTE: Only the Tektronix post processor release tape (REL62B) is required for the EXTRACT job, but both the Tektronix post processor release tape (REL62B) and the TIGS 1 preprocessor release tape (REL62A) are required for the installation job routed to the NOS input queue.

```
EXTRACT,T10,CM65000.
USER(*)              (Supply appropriate accounting
                      information.)
LABEL(TIGSPL,R,L=TIGSTEKV1,
      { HY }                         (7-track, 800 bpi.)
   D= { HD } ,VSN=REL62B,F=SI)       (9-track, 800 cpi.)
      { PE }                         (9-track, 1600 cpi.)
COPYBF(TIGSPL,OLDPL)
UNLOAD(TIGSPL)
UPDATE(F,*==,L=17,D,8)
ROUTE(COMPILE,DC=IN)
7/8/9
=IDENT,INSTALL
=DEFINE,NOS
          { HY }                     (7-track, 800 bpi.)
=DEFINE,  { HD }                     (9-track, 800 cpi.)
          { PE }                     (9-track, 1600 cpi.)
=DEFINE,BASICLDR
                     (Optional UPDATE directive.  If the
                      user wishes to install both a segment/
                      overlay loader library (TIGSLIB)
                      and a basic (nonsegment/nonoverlay)
                      loader library (TIGSSFL), the
                      =DEFINE,BASICLDR directive must
                      be included as part of the UPDATE.
                      If the =DEFINE,BASICLDR direc-
                      tive is not included as part of the
                      UPDATE, then only a segment/
                      overlay loader library (TIGSLIB)
                      will be installed.)
          { NOS500  }
=DEFINE,  { NOS1500 }
          { NOS4000 }
          { NOS8000 }
                     (Specify an output buffer size on
                      NOS.)
=D,TIGS1.XX          (Modify the installation job program
USER(*)               library with appropriate accounting
                      information.  The dummy USER
                      card sequence number, XX, was
                      obtained from the COMPILE file
                      listing resulting from step 1 of
                      the "Installation Procedures".)
6/7/8/9
```

Step 3

Define the TIGS 1 library (or libraries) under user number LIBRARY by entering the following commands at the NOS console.

- Define the segment/overlay loader library, TIGSLIB, under user number LIBRARY:

```
ATTACH,A=TIGSLIB/PW=TEK
SUI,377776.
DEFINE,B=TIGSLIB/UN=LIBRARY,CT=PU
COPY,A,B,V.
```

- If a basic (nonsegment/nonoverlay) loader library, TIGSSFL, was not generated by step 2 of the "Installation Procedures," then omit this step.

Define the basic (nonsegment/nonoverlay) loader library, TIGSSFL, under user number LIBRARY:

```
ATTACH,C=TIGSSFL/PW=TEK
SUI,377776.
DEFINE,D=TIGSSFL/UN=LIBRARY,CT=PU
COPY,C,D,V.
```

VERIFICATION PROCEDURE

Use the following steps for verification:

Step 1

The verification job is file 5 on the Tektronix post processor release tape, (REL62B). File 5 is a program library in UPDATE format with = (equals) as the master control character.

A dummy USER card has been included as part of the verification job. Perform an UPDATE run against file 5 of REL62B and list the COMPILE file to obtain the sequence number which corresponds to the USER card (and any other control cards) which must be modified for installation. A listing of the compile file may be obtained by executing the following procedure:

```
LISTV,T10,CM65000.
USER(*)              (Supply appropriate accounting
                      information.)
LABEL(TIGSPL,R,L=TIGSTEKV1,
      { HY }                         (7-track, 800 bpi.)
   D= { HD } ,VSN=REL62B,F=SI)       (9-track, 800 cpi.)
      { PE }                         (9-track, 1600 cpi.)
SKIPF(TIGSPL,4,17)
COPYBF(TIGSPL,OLDPL)
UNLOAD(TIGSPL)
UPDATE(F,U,*==,L=17)
COPYSBF(COMPILE,OUTPUT)
7/8/9
=IDENT,VERIFY
=DEFINE,NOS
6/7/8/9
```

Step 2

Perform a full UPDATE on file 5 of REL62B modifying the verification job program library with the appropriate accounting information and DEFINE UPDATE directives required for installation. Route the resulting verification job to the NOS input queue by executing a procedure of the following form:

```
XTRACT,T10,CM65000.
USER(*)             (Supply appropriate accounting
                    information.)
LABEL(TIGSPL,R,L=TIGSTEKV1,
     ⎧ HY ⎫                        (7-track, 800 bpi.)
  D= ⎨ HD ⎬,VSN=REL62B,F=SI)   (9-track, 800 cpi.)
     ⎩ PE ⎭                        (9-track, 1600 cpi.)
SKIPF(TIGSPL,4,17)
COPYBF(TIGSPL,OLDPL)
UNLOAD(TIGSPL)
UPDATE(F,*==,L=17,D,8)
ROUTE(COMPILE,DC=IN)
7/8/9
=IDENT,VERIFY
=DEFINE,NOS
=D,TIGS2.YY         (Modify the verification job program
USER(*)             library with appropriate accounting
                    information. The dummy USER
                    card sequence number, YY, was ob-
                    tained from the compile file listing
                    resulting from step 1 of the
                    "Verification Procedure.")
6/7/8/9
```

Step 3

Run the verification program, catalogued as permanent file VRFY by the verification job, to verify the proper installation of TIGS 1. The verification program may be run as follows:

1. Refer to "Terminal Operating Procedures" section to properly set up the terminal.

2. Log in to NOS with a valid family name, user name, password, and charge number.

3. Create a procedure file by entering the following:

```
NEW,SEGLIB CR
TEXT CR
ATTACH(VRFY) CR
ATTACH(TIGSLIB/UN=LIBRARY) CR
LDSET(LIB=TIGSLIB) CR
VRFY. CR
Depress BREAK key for the Time-sharing
Module or press user break 2 key for IAF
Computer responds with EXIT TEXT MODE.
PACK CR
SAVE,SEGLIB CR
```

4. Call the SEGLIB procedure file which will execute the verification program, created in step 2 of the "Verification Procedure", to verify the segment/overlay loader library, TIGSLIB, by the following:

```
CALL,SEGLIB
```

5. The terminal operator will be asked to enter the terminal type. Enter the number which corresponds to the terminal type in use. (See figure I-10-1.)

6. The terminal operator will be asked to enter the baud rate. Enter the baud rate that is in use (see figure I-10-1).

7. The terminal operator will be asked if the tablet is to be used (see figure I-10-1). Enter N. The tablet is not required for the verification program.

8. The terminal operator will be asked if a hardcopy unit is available (see figure I-10-1). Enter N. The verification program does not require the availability of a hardcopy unit.

9. The terminal screen will be cleared and figure I-10-2 will be displayed if terminal type 4, 5, 6, or 7 was selected in step 5. Figure I-10-3 will be displayed if terminal type 2 or 3 was selected. Figure I-10-4 will be displayed if terminal type 1 was selected.

10. If terminal type 1 was selected in step 5, operations 11 through 14 will be skipped.

11. The operator is allowed to pick one of three buttons: WINDOW, RESTORE, or QUIT. The button is picked by moving the crosshairs over the text of the button and depressing an alpha key on the keyboard.

12. If WINDOW is picked, the terminal operator will be asked to locate the lower-left and then the upper-right corner of a new window to display. This is done by moving the crosshairs to the appropriate corner and depressing the T key on the keyboard. When both corners have been located, the new window will be displayed.

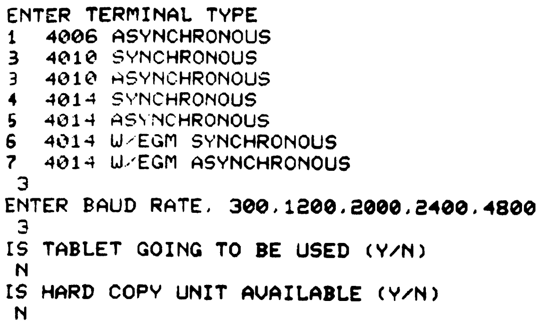
13. If RESTORE is picked, the original display will be displayed.

14. If QUIT is picked, the verification program will terminate.

15. If a basic (nonsegment/nonoverlay) loader library, TIGSSFL, was not installed, then go to operation 19.

16. Create a procedure file to utilize the basic (nonsegment/nonoverlay) loader library, TIGSSFL, by entering the following:

```
NEW,SFLLIB CR
TEXT CR
ATTACH(VRFY) CR
ATTACH(TIGSSFL/UN=LIBRARY) CR
LDSET(LIB=TIGSSFL) CR
VRFY. CR
Press BREAK key for the Time-sharing
Module or press user break number 2
key for IAF. Computer responds with
EXIT TEXT MODE.
PACK CR
SAVE,SFLLIB CR
```



```
ENTER TERMINAL TYPE
1  4006 ASYNCHRONOUS
3  4010 SYNCHRONOUS
3  4010 ASYNCHRONOUS
4  4014 SYNCHRONOUS
5  4014 ASYNCHRONOUS
6  4014 W/EGM SYNCHRONOUS
7  4014 W/EGM ASYNCHRONOUS
 3
ENTER BAUD RATE, 300,1200,2000,2400,4800
 3
IS TABLET GOING TO BE USED (Y/N)
 N
IS HARD COPY UNIT AVAILABLE (Y/N)
 N
```

Figure I-10-1. Terminal User is Asked to Enter Terminal Type, Baud Rate, if Tablet is to be Used, and if Hardcopy is Available

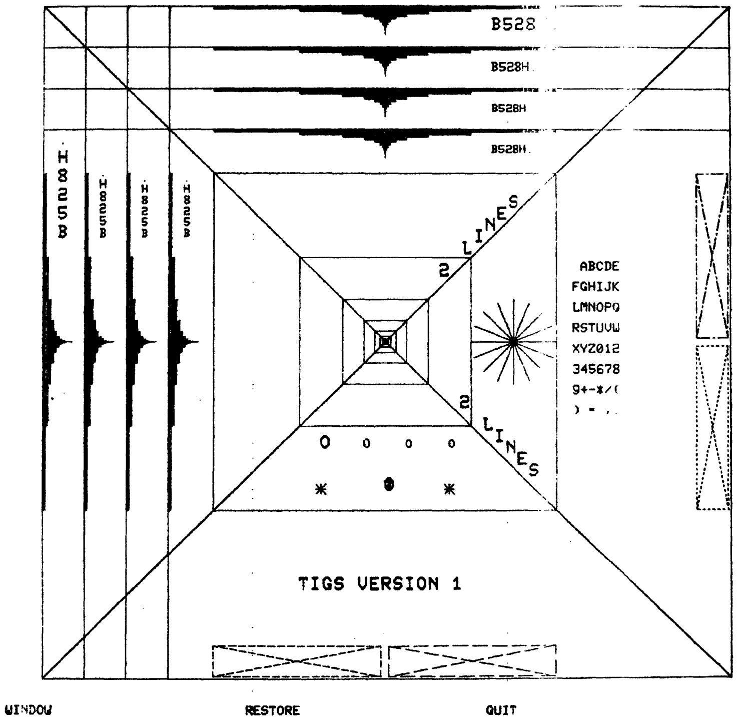


Figure I-10-2. Figure Displayed on Terminal Types 4, 5, 6, or 7

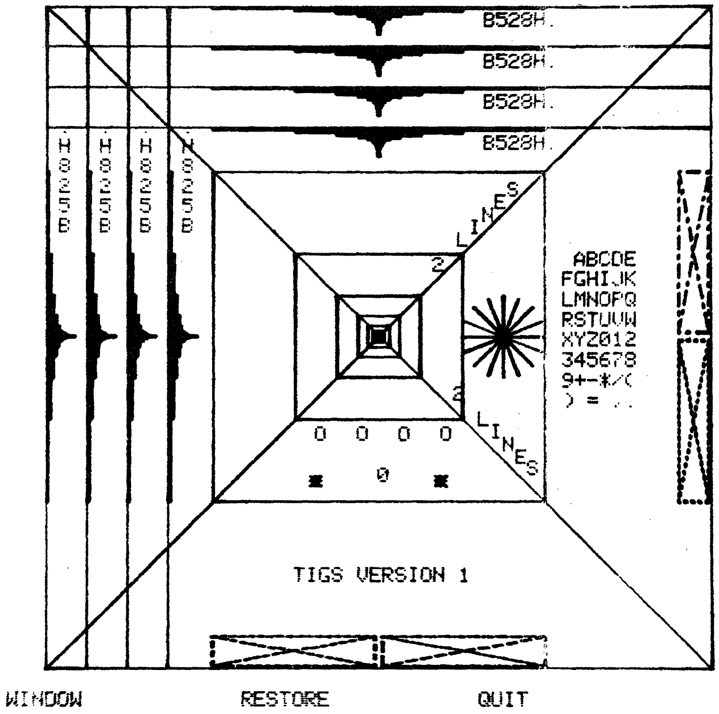


Figure I-10-3. Figure Displayed on Terminal Types 2 or 3

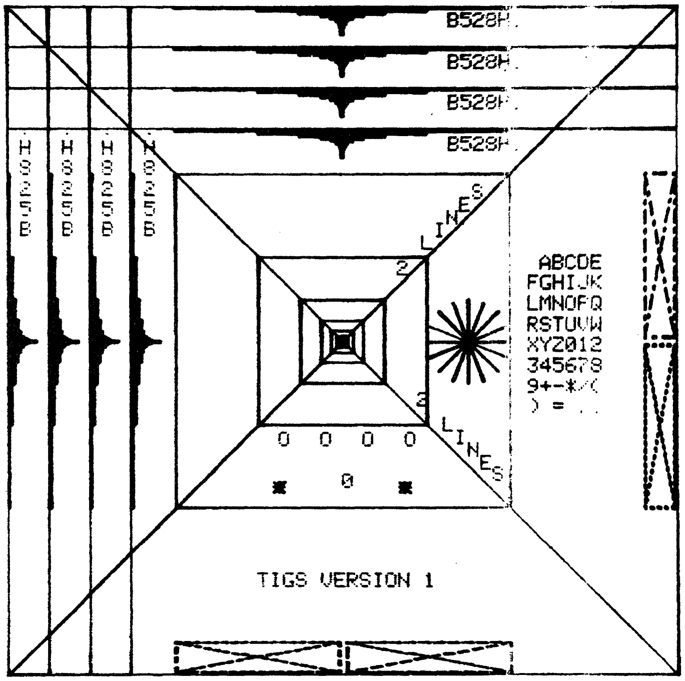


Figure I-10-4. Figure Displayed on Terminal Type 1

17. Call the SFLLIB procedure file which will execute the verification program to verify the basic (nonsegment/nonoverlay) loader library, TIGSSFL, by the following:

    CALL, SFLLIB

18. Repeat operations 5 through 14 to verify the basic (nonsegment/nonoverlay) loader library, TIGSSFL.

19. Purge any unnecessary files generated during the installation or verification of TIGS 1.

20. Log off the system by entering BYE when finished.

TERMINAL OPERATING PROCEDURES

TEKTRONIX 4006 TERMINAL

The Tektronix 4006 terminal operating procedures are as follows:

1. Turn the terminal power on by turning the POWER switch, located at the rear of the display unit, to the ON position.

2. Allow the terminal to warm up.

3. Press the PAGE key to erase the screen and to position the cursor to home position (the upper left-hand corner of the screen).

4. Select the appropriate transmit and receive baud rates using the baud rate switches located at the rear of the display unit.

5. The setting of the FULL/HALF DUPLEX switch located at the rear of the display unit is NOS installation dependent.

6. Turn the modem power on and verify that the modem is connected to the terminal.

7. Dial the appropriate telephone number.

8. The Time-sharing Module will respond with a LOGIN message when ready for interactive command processing. NAM/IAF, however, requires the user to press the RETURN key after the terminal is polled before it will respond with a LOGIN message.

TEKTRONIX 4010/4014 TERMINALS WITH SYNCHRONOUS INTERFACE

The Tektronix 4010 and 4014 terminals (equipped with CDC synchronous interface) synchronous/asynchronous operating procedures are as follows:

1. Turn the terminal power on.

   - Tektronix 4010 terminal: The POWER switch is located beneath the keyboard at the top of the pedestal stand. The red POWER indicator, located at the top of the keyboard, will illuminate when power has been applied to the terminal.

   - Tektronix 4014 terminal: The POWER switch is located on the front lower right-hand corner of the pedestal stand. The green POWER indicator, located on the upper left-hand corner of the keyboard, will illuminate when power has been applied to the terminal.

2. Allow the terminal to warm up.

3. Press the PAGE key to erase the screen and to position the cursor to home position (the upper left-hand corner of the screen).

4. Set the ASCII/ALT switch to the ASCII position.

5. Select the appropriate baud rate switch settings. The switch should select the EXT position if the terminal is to be used for synchronous communications at 2000 and 4800 baud. Otherwise, the terminal baud rate switch setting will correspond to the baud rate in use.

   - Tektronix 4010 terminal: Select the appropriate transmit and receive baud rates using the baud rate switches located at the rear of the pedestal stand.

   - Tektronix 4014 terminal: Select the appropriate baud rate using the baud rate switch located at the rear of the pedestal stand.

6. The setting of the FULL/HALF DUPLEX switch is NOS installation dependent.

7. Set the CODE EXPANDER switch to one of the following positions:

   - Synchronous mode communications: Turn the switch ON.

   - Asynchronous mode communications: Turn the switch OFF.

8. The following keyboard operations establish proper operating mode:

   - Synchronous mode communications: Place the terminal in local mode. Press the SHIFT key. While the SHIFT key is pressed, press the RESET PAGE key. Place the terminal in line mode and repeat the keyboard procedure.

- Asynchronous mode communications: Place the terminal in local mode. Press and keep depressed the SHIFT key. Press and keep depressed the CNTL key. While both the SHIFT and CNTL keys are pressed, press the P key. Place the terminal in line mode.

9. Dial the appropriate telephone number.

10. The Time-sharing Module will respond with a LOGIN message when ready for interactive command processing. NAM/IAF, however, requires the user to press the RETURN key after the terminal is polled before it will respond with a LOGIN message.

TEKTRONIX 4010/4014 TERMINALS WITHOUT SYNCHRONOUS INTERFACE

If the Tektronix 4010 and 4014 terminals are not equipped with the CDC synchronous interface option, then the terminal can be operated in asynchronous mode only.

The Tektronix 4010 and 4014 terminals asynchronous mode operating procedures are essentially the same as for the synchronous mode operating procedures. However, step 8 of the synchronous mode operating procedures can be omitted. No special keyboard operations are required to establish mode switching since the terminals will always be in asynchronous mode when not equipped with the CDC synchronous interface option.

TOTAL UNIVERSAL 1

RELEASE DESCRIPTION

TOTAL Universal 1 consists of the following relocatable binaries:

DBGEN (Data base generation program) - this program reads user-prepared DBDL statements and generates COMPASS source statements which in turn produce the data base descriptor module.

DBFMT (Data set format program) - this program reads format parameter statements, and, utilizing a data base descriptor module, preformats the data sets.

DATBAS (Data base interface module) - this module serves as an interface between the user application program and the TOTAL and data base descriptor modules.

TOTAL (Data base management module) - this module provides the data management capability of the system, interpreting and executing the various DML commands from the user application program.

DBRCV (Data base recovery module) - this module, optional in use, provides the ability to recover record images from the TOTAL logging file.

These files are to be installed on the user's permanent file library (refer to the following section on "Installation Procedures").

Hardware Requirements

TOTAL Universal 1 can be maintained and run on the same minimum hardware requirements as NOS 1.

Deficiencies

None.

INSTALLATION PROCEDURES

TOTAL Universal 1 resides on the tape REL54A. REL54A is a program library in binary format. It has the following characteristics: labeled, 7- or 9-track, 800 bpi, binary recording mode, TOTALU1P0 as file ID in HDR1 label, six files:

File 1 - installation and verification deck in UPDATE format,

File 2 - empty,

File 3 - five TOTAL relocatable binaries (described earlier),

File 4 - empty,
File 5 - empty, and
File 6 - sample output of installation and validation job.

Installation consists of the following steps:

1. Determine update directives to modify the install deck (as described in the next section).

2. Create the following job to call the install/validate job from the install tape and submit it as a batch job:

```
JOB/ACCOUNT.     (Supply valid JOB/
                 ACCOUNT card.)
LABEL,TOT,R,L=TOTALU1P0,D=HY,
   VSN=REL54A.
UPDATE,Q,*=$$$$,P=TOT,L=F.
SUBMIT(COMPILE,B)
7/8/9
UPDATE directives
6/7/8/9
```

3. Validate the install validate job output against the sample output copied from the install tape.

INSTALLATION PARAMETERS

Following is a sample UPDATE directive deck to be used as described in the previous section. All options are shown and explained.

```
$INST
$COMPILE INSTALL
{$DEFINE MT }          (7-track install tape.)
{$DEFINE NT }          (9-track install tape.)
$DELETE INSTALL.2,INSTALL.3
 JOB/ACCOUNT.          (Site/user dependent.)
```

INSTALLATION JOB

The installation/validation job performs the following:

1. Issue LABEL to have install tapes assigned to job.

2. DEFINE's permanent file space for the five TOTAL files, and copies them from tape to the permanent files. An example of the DEFINE:

   ```
   DEFINE,DBGEN/CT=PU,M=R.
   ```

3. Validation is begun by defining a data base using DBGEN.

4. The test data base is formatted with DBFMT.

5. Sample data is placed in the data base with a COBOL program, and retrievals are performed. DATBAS and TOTAL are verified in this phase.

6. DBRCV is next run against the data base for its validation.

7. Finally, the sample output of the install/validation job is copied from the install tape to output.

TOTAL UNIVERSAL EXTENDED 1

RELEASE DESCRIPTION

TOTAL Universal Extended 1 resides on REL54B. REL54B is a program library in binary format. It has the following characteristics: labeled, 7- or 9-track, 800 bpi, binary recording mode, eight files:

File 1 (Installation deck) - this deck reflects the latest materials for each release level. The tape ID field shows the latest level number.

File 2 - TOTAL (Data base management module) - this module provides the data mangement capability of the system, interpreting and executing the various DML commands from the user application program.

File 3 - DATBAS (Data base interface module) - this module serves as an interface between the user application program and the TOTAL and data base descriptor modules.

File 4 - DBGEN (Data base generation program) - this program reads user-prepared DBDL statements and generates COMPASS source statements which in turn produce the data base descriptor module.

File 5 - DBFMT (Data set format program) - this program reads format parameter statements, and utilizing a data base descriptor module, preformats the data sets.

File 6 - DBRCV (Data base recovery module) - this module, optional in use, provides the ability to recover record images from the TOTAL logging file.

File 7 - DATBASE (Data base interface module for transaction processing) - this module serves as an interface between the user application program and the TOTAL Universal Extended 1 and data base descriptor modules through TRANEX.

File 8 - TOTALE (Data base management module for transaction processing) - this module provides the data management capability of the system, interpreting and executing the various DML commands from the user application program.

Hardware Requirements

Requirements are the same as the minimum hardware requirements for NOS 1.

INSTALLATION PROCEDURES

REL54B contains TOTAL Universal Extended 1. The following job installs TOTAL under the TRANEX user number.

```
JOB.              (Supply valid JOB card.)
USER.             (Supply valid USER card.)
REQUEST,OLDPL,D=800,F=I,LB=KL,VSN=REL54B.
UPDATE,F,*=/,D,8.
SUBMIT,COMPILE.
7/8/9
/DEFINE,NT.       (Optional.)
/DEFINE,LIB.      (Optional.)
6/7/8/9
DEFINE OPTIONS
NT                (Track tape [7 track assumed].)
LIB               (Generates a move deck to install
                  files under user number LIBRARY.)
```

NOTE

If the option to place files under user number LIBRARY is chosen, a procedure file (MOVLIB) is provided under the installation account (TRANEX user number) to allow the user to put all the files except TOTALE under user number LIBRARY, and also leave them under the TRANEX user number. To do this, the installation job above is run. Then the user must enter the following from the console:

```
X.DIS
USER,KB100DC.    (TRANEX user number.)
CALL,MOVLIB,C.
```

TOTALE must always reside under the TRANEX user number. If the user elects to install under another account number, the file TOTALE must be transferred to the TRANEX user number.

To list the installation deck, do the following:

```
JOB.              (Supply valid JOB card.)
USER.             (Supply valid USER card.)
REQUEST,OLDPL,D=800,F=I,LB=KL,VSN=REL54B.
UPDATE,F,L=F,C=0,*=/.
6/7/8/9
```

TOTAL UNIVERSAL 2

RELEASE DESCRIPTION

TOTAL Universal 2 consists of the following relocatable binaries:

DBGEN (Data base generation program) - this program reads user-prepared DBDL statements and generates COMPASS source statements which in turn produce the data base descriptor module.

DBFMT (Data set format program) - this program reads format parameter statements, and, utilizing a data base descriptor module, preformats the data sets.

DATBAS (Data base interface module) - this module serves as an interface between the user application program and the TOTAL and data base descriptor modules.

TOTAL (Data base management module) - this module provides the data management capability of the system, interpreting and executing the various DML commands from the user application program.

DBRCV (Data base recovery module) - this module, optional in use, provides the ability to recover record images from the TOTAL logging file.

TOTUTIL (TOTAL utilities) - this program provides TOTAL users with a generalized utility package capable of performing file loading/unloading, statistics, and other data base functions.

These files are to be installed on the user's permanent file library (refer to the following section on "Installation Procedures").

Hardware Requirements

TOTAL Universal 2 can be maintained and run on the same minimum hardware requirements as NOS 1.

Deficiencies

None.

RELEASE MATERIALS

TOTAL UNIVERSAL 2 release materials are contained on program library tape REL54C. REL54C has the following characteristics: labeled, 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, TOTALU2P0 as file ID in HDR1 label. REL54C contains seven files:

File 1 - installation and verification deck in UPDATE format,

File 2 - empty,

File 3 - five TOTAL relocatable binaries: DBGEN, DBFMT, DATBAS, TOTAL, DBRCV,

File 4 - source code for TOTAL utilities in UPDATE format,

File 5 - empty,

File 6 - COBOL compilation output from the validation job, and

File 7 - execution output from the validation job.

INSTALLATION PROCEDURES

Installation consists of the following steps:

1. Determine UPDATE directives to modify the install deck (as described in the next section).

2. Create the following job to call the install/validate job from the install tape and submit it as a batch job:

```
JOB/USER/CHARGE CARDS.  (Supply
    valid JOB/USER/CHARGE cards.)
LABEL,TOT,R,L=TOTALU2P0,D= ⎧HY⎫
    VSN=REL54C.            ⎨HD⎬ ,
                           ⎩PE⎭
UPDATE,Q,*  $$$$,P=TOT,L=F.
SUBMIT(COMPILE,B)
7/8/9
UPDATE directives
6/7/8/9
```

3. Validate the install/validate job output against the sample output copied from the install tape.

INSTALLATION PARAMETERS

Following are the UPDATE directives to be used as described in the previous section. All options are shown and explained.

```
 $IDENT INST
 $DEFINE CBL4 †   (Compiles utilities with COBOL 4 -
                  Default is COBOL 5)
⎧$DEFINE HY⎫      (7-track 800 bpi)  ⎫Select one to
⎨$DEFINE HD⎬      (9-track 800 cpi)  ⎬ correspond to
⎩$DEFINE PE⎭      (9-track 1600 cpi) ⎭ the installation
                                       tape density.
 $DELETE INSTALL.2,INSTALL.4
 JOB/USER/CHARGE CARDS.  (Supply valid JOB/
                          USER/CHARGE cards)
 $COMPILE INSTALL
```

[†] The ability to compile the utilities using COBOL 4 is provided as a user convenience at this time, but may not be supported in future releases.

INSTALLATION JOB

The installation/validation job performs the following:

1. Issues LABEL card to have install tape assigned to job.

2. DEFINE's permanent file space for the five TOTAL files, and copies them from tape to the permanent files.

3. Compiles TOTAL utilities and creates the relocatable binary on the permanent file TOTUTIL.

4. Validation is begun by defining a data base using DBGEN.

5. The test data base is formatted with DBFMT.

6. Sample data is placed in the data base with a COBOL program, and retrievals are performed. DATBAS and TOTAL are verified in this phase.

7. DBRCV is run against the data base for its validation.

8. The TOTAL utilities are validated by executing the utilities statistics option.

9. Finally, the sample output of the install/ validation job is copied from the install tape to output for verification.

FILES CREATED

The installation/validation job purges and defines the following files:

DBGEN DBFMT DATBAS TOTAL DBRCV	Five TOTAL binaries
TOTUTIL -	TOTAL utility binary
CUSTDB -	Validation DBMOD
CUCUST CUINVF	Validation data base files

The user should insure that these files will not conflict with already existing files prior to installation.

TOTAL UNIVERSAL EXTENDED 2

RELEASE DESCRIPTION

TOTAL Universal Extended 2 consists of the following relocatable binaries:

DBGEN (Data base generation program) - this program reads user-prepared DBDL statements and generates COMPASS source statements which in turn produce the data base descriptor module.

DBFMT (Data set format program) - this program reads format parameter statements, and utilizing a data base descriptor module, preformats the data sets.

DATBAS (Data base interface module) - this module serves as an interface between the user application program and the TOTAL and data base descriptor modules.

TOTAL (Data base management module) - this module provides the data management capability of the system, interpreting and executing the various DML commands from the user application program.

DBRCV (Data base recovery module) - this module, optional in use, provides the ability to recover record images from the TOTAL logging file.

DATBASE (Data base interface module for transaction processing) - this module serves as an interface between the user application program and the TOTAL Universal Extended 2 and data base descriptor modules through TRANEX.

TOTALE (Data base management module for transaction processing) - this module provides the data management capability of the system, interpreting and executing the various DML commands from the user application program.

TOTUTIL (TOTAL utilities) - this program provides TOTAL users a generalized utility package capable of performing file loading/unloading, statistics and other data base functions.

Hardware Requirements

Requirements are the same as the minimum hardware requirements for NOS 1.

Deficiencies

None.

RELEASE MATERIALS

TOTAL Universal Extended 2 release materials are contained on program library tape REL54D. REL54D has the following characteristics: labeled, 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, TOTALE2P0 as file ID in HDR1 label. REL54D contains seven files:

File 1 - installation and verification deck in UPDATE format,

File 2 - empty,

File 3 - seven TOTAL relocatable binaries: DBGEN, DBFMT, DATBAS, TOTAL, DBRCV, DATBASE, TOTALE,

File 4 - source code for TOTAL utilities in UPDATE format,

File 5 - empty,

File 6 - COBOL compilation output from the validation job, and

File 7 - execution output from the validation job.

INSTALLATION PROCEDURES

Installation consists of the following steps:

1. Determine UPDATE directives to modify the install deck (as described in the next section).

2. Create the following job to call the install/validate job from the install tape and submit it as a batch job:

```
JOB/USER/CHARGE. (Supply valid JOB/USER/
                        CHARGE cards.)
LABEL,TOT,R,L=TOTALE2P0,D= (HY)
    VSN=REL54D.            {HD} ,
                           (PE)
UPDATE,Q,*=$$$$,P=TOT,L=F.
SUBMIT(COMPILE,B)
7/8/9
UPDATE directives
6/7/8/9
```

3. Validate the install/validate job output against the sample output copied from the install tape.

INSTALLATION PARAMETERS

Following are the UPDATE directives to be used as described in the previous section. All options are shown and explained.

```
 $IDENT INST
 $DEFINE CBL4[†] (Compiles utilities with COBOL 4 -
               Default is COBOL 5)
($DEFINE HY)  (7-track 800 bpi)  )Select one to
{$DEFINE HD}  (9-track 800 cpi)  }correspond to
($DEFINE PE)  (9-track 1600 cpi) )the installation
                                  tape density
 $DELETE INSTALL.2,INSTALL.4
 JOB/USER/CHARGE CARDS. (Supply valid JOB/
                          USER/CHARGE cards)
 $DEFINE LIB (If selected, the MOVLIB procedure
     file is installed and can be used later to install
     TOTAL files under user number LIBRARY.)
 $COMPILE INSTALL
```

INSTALLATION JOB

The installation/validation job performs the following:

1. Issues LABEL card to have install tape assigned to job.

2. DEFINE's permanent file space for the seven TOTAL files, and copies them from tape to the permanent files.

3. Compiles TOTAL utilities and creates the relocatable binary on the permanent file TOTUTIL.

4. Validation is begun by defining a data base using DBGEN.

5. The test data base is formatted with DBFMT.

6. Sample data is placed in the data base with a COBOL program, and retrievals are performed. DATBAS and TOTAL are verified in this phase.

7. DBRCV is next run against the data base for its validation.

8. The TOTAL utilities are validated by executing the statistics option.

9. Finally, the sample output of the install/validation job is copied from the install tape to output for verification.

FILES CREATED

The installation/validation job purges and defines the following file:

DBGEN DBFMT DATBAS TOTAL DBRCV DATBASE TOTALE	- Seven TOTAL binaries.
TOTUTIL	- TOTAL utility binary
MOVLIB	- If $DEFINE LIB was selected during installation (MOVLIB is an indirect file)

[†] The ability to compile the utilities using COBOL 4 is provided as a user convenience at this time, but may not be supported in a future release.

CUSTDB	- Validation DBMOD
CUCUST	- Validation data base files
CUINVF	

The user should insure that these files will not conflict with already existing files prior to installation.

RELEASE DESCRIPTION

TOTAL/ATHENA consists of four relocatable binaries:

ATB1 ATB2	Modules that interface with the TOTAL data base management system.

ATRP = the Report Writer module.

ATPLT = the Plotter module.

HARDWARE REQUIREMENTS

TOTAL/ATHENA can be maintained and run on the same minimum hardware requirements as NOS 1.

SOFTWARE REQUIREMENTS

TOTAL Universal 2 must be installed before TOTAL/ATHENA installation is attempted.

INSTALLATION PROCEDURES

REL58 contains six files:

File 1 - install deck in UPDATE format,

File 2 - empty,

File 3 - four relocatable binaries:

-ATB1 -ATB2	TOTAL/ATHENA query update binaries,
-ATRP	TOTAL/ATHENA Report Writer, and
-ATPLT	TOTAL/ATHENA Plotter;

File 4 - TOTAL/ATHENA, Report Writer, and Plotter source code deck in UPDATE format,

File 5 - demonstration decks in UPDATE format:

-TOTAL/ATHENA,
-REPORT WRITER,
-PLOTTER, and
-TEXT INFORMATION;

File 6 - sample output from demonstration jobs.

Installation consists of the following steps:

1. Choose the UPDATE directives that are required in step 2 to modify the install deck.

   a. Select a DEFINE,MT or DEFINE,NT or DEFINE,NTPE for install tape processing.

   b. Select a DEFINE,BINARY or DEFINE,SOURCE. For execution of TOTAL/ATHENA, Report Writer and Plotter compiled binaries, select the DEFINE,BINARY. For users to compile their own copies of TOTAL/ATHENA, Report Writer and Plotter, select the DEFINE,SOURCE control directive.

   c. Select DEFINE,DEBUG if COBOL DEBUG mode is desired.

2. Create and submit for batch processing the following job to call the install deck from the install tape:

```
JOB.            (Supply valid JOB card.)
USER.           (Supply valid USER card.)
CHARGE.
LABEL,TA1,R,L=TOTALATHENA,
      ( HY )
   D= { HD } , VSN=REL58.
      ( PE )
UPDATE(Q,P=TA1,L=F,W,D,8,R,*=/)
SUBMIT(COMPILE,B)
7/8/9
/ID INST
        (MT   ((7-track,  800 bpi.) )  Select MT,
/DEFINE{NT   {(9-track,  800 cpi.) }  NT, or
        (NTPE((9-track, 1600 cpi.) )  NTPE
/DEFINE,(SOURCE      Select BINARY or SOURCE
        {BINARY      (see instructions.)
[/DEFINE,(DEBUG]
/D INSTALL.2,INSTALL.4
JOB.
USER.           (Supply valid USER card.)
CHARGE.         (Supply valid CHARGE card.)
/D INSTALL.99,INSTALL.101
JOB.
USER.           (Supply valid USER card.)
CHARGE.         (Supply valid CHARGE card.)
/COMPILE,INSTALL
6/7/8/9
```

3. Verify that the output from the demonstration job matches the listing of the install tape sample output file. Verify that the output obtained from interactively executing the ATHENA plotter matches the output listed in the demonstration job text file.

INSTALLATION JOB

The installation jobs perform the following:

1. Issue LABEL to have REL58 assigned to job.

2. For installation of the T/A binary files (/DEFINE, BINARY), the install job defines the two TOTAL/ ATHENA Report Writer and Plotter files and copies them from REL58 to the permanent files (no passwords are assigned).

3. For the user to compile binaries from TOTAL/ ATHENA source code (/DEFINE,SOURCE) the install job defines the two TOTAL/ATHENA files, the TOTAL/ATHENA Report Writer, and Plotter files, copies and updates the TOTAL/ATHENA source code from tape to disk, and compiles the source code generating the TOTAL/ATHENA, Report Writer and Plotter binaries.

4. A TOTAL/ATHENA demonstration is begun by defining a data base using DBGEN (TOTAL module).

5. A test data base is formatted with DBFMT (TOTAL module).

6. Sample data is placed in the data base with TOTAL/ ATHENA and then TOTAL/ATHENA retrievals and functions are performed.

7. A TOTAL/ATHENA Report Writer demonstration is then processed with SUBSET and COMMAND file data supplied on the install tape.

8. A TOTAL/ATHENA plotter demonstration should then be processed by the installer. The plotter data base and a text file of instructions to the installer is supplied on the install tape.

9. The output of the demonstration is then compared against the validation output copied from the install tape earlier.

This job lists the install deck as it appears on file 1 of REL58:

```
JOB.        (Supply valid JOB card.)
USER.       (Supply valid USER card.)
CHARGE.
LABEL,TA1,R,L=TOTALATHENA,
       ( HY )
    D= { HD } ,VSN=REL58.
       ( PE )
UPDATE(Q,P=TA1,L=F,*=/)
7/8/9
/ID ATHLIST
/COMPILE INSTALL
6/7/8/9
```

DEMONSTRATION JOB

The demonstration deck in UPDATE format includes the demonstration job and the necessary input records to the demonstration job. The demonstration job performs these operations:

1. Generates a TOTAL DBMOD and formats a TOTAL data base.

2. Creates and executes the TOTAL/ATHENA query/ update executable binary.

3. Executes the TOTAL/ATHENA report generator module.

4. Defines and copies a TOTAL/ATHENA plotter SUBSET file.

5. Prints a text file that describes how to verify the TOTAL/ATHENA plotter.

This job lists the demonstration deck as it appears on file 5 of the install tape:

```
JOB.                (Supply valid JOB card.)
USER.               (Supply valid USER card.)
CHARGE.
LABEL,TA1,R,L=TOTALATHENA,
       ( HY )
    D= { HD } ,VSN=REL58.
       ( PE )
SKIPF,TA1,4.
UPDATE(Q,L=F,P=TA1,*=SSSS,R)
7/8/9
SID ATHLIST
SCOMPILE TADEMO
6/7/8/9
```

NOTE

The binary files on this release were generated on a non-CMU CYBER system.

UNIPLOT 3 PRODUCT OPTIONS 13

UNIPLOT 3 OVERVIEW

RELEASE DESCRIPTION

UNIPLOT 3 (Universal Plotting Software Version 3) consists of two components: a library of FORTRAN Extended routines which are CalComp call-compatible (UNIPLOT) and a post processor (UNIPOST). The UNIPLOT routines write a neutral picture file, NPFILE, which can then be read by the post processor to produce graphic output on a variety of plotting devices. UNIPLOT 3 is designed to be used by all application programs which make CalComp calls to produce graphic displays.

The operation of UNIPLOT 3 requires the installation of the batch and/or interactive facility (depending on the post processor used) and all other associated products under NOS 1.

HARDWARE CONFIGURATION

The minimum hardware configuration required by NOS is required to support UNIPLOT 3.

HARDWARE OPTIONS

A graphic terminal or plotting device is required if displaying of graphic data is to be done on site. The terminals currently supported by UNIPLOT 3 are:

1. Tektronix 401X series graphic terminals,
2. Houston Instrument BTC-7 controllers and compatible plotters,
3. CalComp 906 controllers and compatible plotters.

NOTES AND CAUTIONS

All limitations applicable to NOS also apply to UNIPLOT 3.

All UNIPLOT 3 neutral picture routines are callable from a user's application program written in FORTRAN Extended only.

All three supported devices operate asynchronously only.

UNIPLOT 3

RELEASE MATERIALS

UNIPLOT 3 resides on the tape known as REL66. REL66 has the following characteristics: 7-track (800 bits per inch [bpi]) or 9-track (800 or 1600 characters per inch [cpi]), binary recording mode, UNIPLOTV3 as file ID in the HDR1 label, and six files:

File 1 - Installation job program library in UPDATE PL form,

File 2 - Absolute binaries for UNIPOST main overlay and one secondary overlay (DUMP) which does a dump of the NPFILE,

File 3 - Relocatable binaries for the UNIPLOT neutral picture routines,

File 4 - UNIPLOT 3 program library in UPDATE PL form consisting of two records; the first is UNIPOST and the second is UNIPLOT,

File 5 - Verification job program library in UPDATE PL form, and

File 6 - Empty.

INSTALLATION PROCEDURE

To obtain a listing of the installation job and the verification job execute the following:

```
LIST.
USER.                    (Supply valid USER card.)

LABEL(OLDPL,L=UNIPLOTV3,PO=R,R,
         ⎧ HY ⎫
      D= ⎨ HD ⎬ , VSN=REL66)
         ⎩ PE ⎭
UPDATE(F,*==,L=7)
SKIPF(OLDPL,4)
UPDATE(F,*==,L=7,R)
6/7/8/9
```

The installation of UNIPLOT 3 saves the library of UNIPLOT neutral picture routines, the relocatable binaries for the UNIPOST post processor, the absolute binaries for UNIPOST and the UNIPLOT program library. The installation job resides in UPDATE program library format as file 1 of the UNIPLOT 3 release tape, REL66, with a master character of equals (=). As part of the following job, a dummy USER statement on the installation program library must be deleted and the appropriate USER statement inserted.

CAUTION

The following direct access permanent files are saved when this job is run. If files already exist under these names, they must either be purged or renamed:

1. UNIPLOT/CT=PU, PW=UNIPLOT - relocatable binaries for UNIPLOT neutral picture routines
2. UNIPOST/CT=PU, PW=UNIPLOT - absolute binaries for UNIPOST post processor
3. POSTLGO/CT=PU, PW=UNIPLOT - relocatable binaries for UNIPOST post processor
4. UNV30PL/CT=PU, PW=UNIPLOT - UPDATE program library

NOTE

A backup tape is not created by this job. UNIPOST and the UNIPLOT program library on REL66 are not replaced with versions containing the post processor code. However, the direct access files created by the installation job do contain the integrated post processor routines.

If the Tektronix 401X post processor option is installed, the following permanent file is also created as direct access:

PLOT10/CT=PU, PW=PLOT10 - relocatable binaries for Tektronix driver routines.

To install UNIPLOT 3 with the desired post processor(s) run the following job:

```
EXTRACT.
USER.                (Supply valid USER statement)
LABEL(OLDPL, L=UNIPLOTV3, PO=R, R,
          ⎧ HY ⎫
       D= ⎨ HD ⎬ , VSN=REL66)
          ⎩ PE ⎭
UPDATE(Q, *==, D, 8)
SUBMIT(COMPILE)
7/8/9
=IDENT ANY
=DEFINE HI200     (If installing Houston Instruments
                   post processor)
=DEFINE CAL906    (If installing CalComp post
                   processor)
=DEFINE TEK       (If installing Tektronix post
                   processor)
        ⎧ MT   ⎫  (If 7-track tape)
=DEFINE ⎨ NT   ⎬  (If 9-track 800 cpi tape)
        ⎩ NT16 ⎭  (If 9-track 1600 cpi tape)
=DELETE. INSTALL.XX
USER.                (Supply valid USER statement)
=COMPILE INSTALL
6/7/8/9
```

NOTE

REL66 is the only tape needed for the EXTRACT job. However, the job submitted to the NOS input queue requires REL66 and all post processor tapes being installed (REL66A, REL66B and/or REL66C). See the individual post processor sections for descriptions of these tapes.

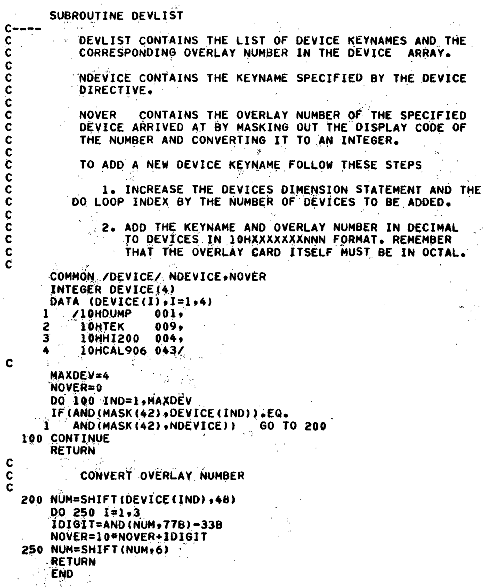
INSTALLING A NEW DEVICE

CREATING A NEW POST PROCESSOR DEVICE OVERLAY

The post processor reads vector and character strings from the UNIPLOT neutral picture file, modifies that information according to user directives, and calls subroutines to produce vector and character commands acceptable to the selected device. The subroutines that produce acceptable commands are different for each plotting device, so if many devices are supported, the post processor core requirements could grow very large. To avoid this large core requirement, the subroutines specific to a given device are placed in one overlay, which is called into core when the device is selected and kept there until the post processor execution is terminated.

A labeled common block, /PPREQ/, is used by the post processor main overlay to communicate with the device routines in the primary overlay. Each time control is returned to the primary overlay, /PPREQ/ contains information which directs the overlay's main program to call a specific device routine with parameter values also contained in /PPREQ/.

In order to support a new plotting device, the programmer must write a new overlay main program that calls initialization, vector plotting, pen select, termination and possible character plotting routines specific to the device being supported. Then the post processor relocatables must be reloaded along with the new overlay.



```
      SUBROUTINE DEVLIST
C----
C         DEVLIST CONTAINS THE LIST OF DEVICE KEYNAMES AND THE
C         CORRESPONDING OVERLAY NUMBER IN THE DEVICE  ARRAY.
C
C         NDEVICE CONTAINS THE KEYNAME SPECIFIED BY THE DEVICE
C         DIRECTIVE.
C
C         NOVER   CONTAINS THE OVERLAY NUMBER OF THE SPECIFIED
C         DEVICE ARRIVED AT BY MASKING OUT THE DISPLAY CODE OF
C         THE NUMBER AND CONVERTING IT TO AN INTEGER.
C
C         TO ADD A NEW DEVICE KEYNAME FOLLOW THESE STEPS
C
C            1. INCREASE THE DEVICES DIMENSION STATEMENT AND THE
C        DO LOOP INDEX BY THE NUMBER OF DEVICES TO BE ADDED.
C
C            2. ADD THE KEYNAME AND OVERLAY NUMBER IN DECIMAL
C               TO DEVICES IN 10HXXXXXXXNNN FORMAT. REMEMBER
C               THAT THE OVERLAY CARD ITSELF MUST BE IN OCTAL.
C
      COMMON /DEVICE/ NDEVICE,NOVER
      INTEGER DEVICE(4)
      DATA (DEVICE(I),I=1,4)
     1   /10HDUMP   001,
     2    10HTEK    009,
     3    10HHI200  004,
     4    10HCAL906 043/
C
      MAXDEV=4
      NOVER=0
      DO 100 IND=1,MAXDEV
      IF(AND(MASK(42),DEVICE(IND)).EQ.
     1   AND(MASK(42),NDEVICE))   GO TO 200
  100 CONTINUE
      RETURN
C
C         CONVERT OVERLAY NUMBER
C
  200 NUM=SHIFT(DEVICE(IND),48)
      DO 250 I=1,3
      IDIGIT=AND(NUM,77B)-33B
      NOVER=10*NOVER+IDIGIT
  250 NUM=SHIFT(NUM,6)
      RETURN
      END
```

WRITING A NEW DEVICE OVERLAY MAIN PROGRAM

The design requirements for an overlay main program are as follows:

1. That it call device-dependent routines in order to plot the graphic data specified in common block /PPREQ/. (Details are given on the following pages.)

2. That upon being entered with an initialization request, it enters into /PPREQ/ a list of properties and parameters of the device that the overlay supports before the first call to NPFDATA.

3. That the overlay makes calls to a main overlay subroutine NPFDATA to obtain data from the neutral picture file. The NPFDATA subroutine will set the /PPREQ/ common block and return to the overlay. Thus, the overlay will loop, calling NPFDATA until processing is complete.

The name of the overlay must be UNIPOST and the number must be in octal and correspond to the decimal number of the name of the device in table DEVICE in subroutine DEVLIST.

If the device dependent routines use FORTRAN I/O, TAPE4 should be used as the file to which the plot commands will be written. If the device routines open their own file and use CIO, the filename used for the plot output is specified in IPRM(2) when the overlay is called with IREQ=0.

INTERFACE WITH THE DEVICE ROUTINES

The manufacturer-supplied device dependent subroutines for each different plotting device are loaded in different overlays. Each overlay has a custom-written main program that calls a working routine in the main (0,0) overlay to pick up request codes and parameters from the neutral picture file. Upon return to the primary overlay, these codes and respective parameters are in the PPREQ common block. The codes direct the primary overlay's main program to call a specific device dependent routine with the parameters supplied.

The PPREQ common block is defined as follows:

```
COMMON/PPREQ/IREQ,IPRM(31)
DIMENSION FPRM(31)
EQUIVALENCE (IPRM,FPRM)
```

IREQ is an integer whose value specifies the kind of request being made. IPRM is an array that contains integer parameters to be used in carrying out the request. FPRM is an array that contains floating point parameters to be used in carrying out the request. The requests are as follows:

INITIALIZATION REQUEST (IREQ=0)

The device overlay returns to the main program certain information about the display or plotting device. It is usually a requirement of the device dependent subroutines that they be initialized only once in a run unless PLOT (X,Y,999) has been called.

Upon entry into the overlay with IREQ=0, the IPRM array is set as follows:

IPRM(1) = device keyname

IPRM(2) = name of the device dependent plot file to be used only with CIO; default is PLOTF

The primary overlay main program must then set the values in PPREQ as follows:

IPRM(1) = device type: 1 = drum plotter
2 = table plotter
3 = storage tube display
4 = microfilm device

FPRM(2) = display surface extent in X direction in inches or centimeters

FPRM(3) = display surface extent in Y direction in inches or centimeters

IPRM(4) = number of pens

IPRM(5) = is there hardware character generation?
0 = no 1 = yes

IPRM(6) = are there hardware generated dotted lines?
0 = no 1 = yes

IPRM(7) = are there hardware generated dashed lines?
0 = no 1 = yes

IPRM(8) = is there hardware arc generation?
0 = no 1 = yes

IPRM(9) = is there hardware cubic generation?
0 = no 1 = yes

IPRM(10) = number of increments per inch (i.e., 100,200,400,...)

IPRM(11) = normal mode of operation
0 = batch processing
1 = interactive processing

IPRM(12) = post processor device unit of measure
0 = inch
1 = centimeter

The device overlay then calls the working routine, NPFDATA, in the main overlay which transfers the above information into the appropriate variables and transfers control to the main processor.

The main processing unit of the main overlay reads another buffer from NPFILE and fills the PPREQ common area and returns to the working routine which returns to the device overlay.

Upon return to the device overlay, PPREQ contains one of the following requests:

VECTOR REQUEST (IREQ=1)

The vector request provides the arguments necessary for a call to the CalComp standard subroutine PLOT or its equivalent. A value of 999 in IPRM(3) indicates that the current request is the final request in the run, and buffers should be purged, files closed, as required, and the program ended. A dashed or dotted vector (IPRM(3) = 4 or 5) is requested only if the presence of those features was indicated in the return from the initialization request.

FPRM(1) = X-coordinate

FPRM(2) = Y-coordinate

IPRM(3) = vector type:

2	for solid line	
3	for invisible	
4	for dash line	Only if hardware
5	for dotted line	dashed/dotted
-3	for new picture	lines are avail-
999	for end of run	able.

NEWPEN REQUEST (IREQ=2)

A call to the CalComp standard subroutine NEWPEN or the equivalent is requested.

IPRM(1) = pen number

IPRM(2) = maximum number of pens

The request may be made for a single pen plotter if the application creating the neutral picture file calls NEWPEN with a negative pen number. If IPRM(1) is negative on a single pen plotter, the following calls should be made if applicable;

```
CALL PLOT(0.0,0.0,999)
CALL PLOTS(IBUF,ISIZE,IUNIT)
```

ALPHA STRING REQUEST (IREQ=3)

This request provides the overlay with the parameters needed for a character string call to the CalComp standard subroutine SYMBOL (only if hardware character generation is available on the device). Up to 100 characters may be present in each request. The X and Y coordinates usually have values of 999.0, indicating that the string is to be started at the current pen position. Centered symbols are not included in this type of request.

FPRM(1) = X-coordinate of start of character string

FPRM(2) = Y-coordinate of start of character string

FPRM(3) = character height

FPRM(4) = angle of baseline of character string in degrees

IPRM(5) = number of characters

IPRM(6)-IPRM(15) = BCD character string

DEVICE MODE REQUEST (IREQ=4)

Arguments to device independent subroutines are passed unchanged through the neutral plot file. If the user of the post processor selects the proper display device, a plotter mode request will be made to the overlay. The overlay should inspect IPRM(1) which will contain the name of the device specific routine in left-justified A10 format, and then call the specified routine using the subsequent words of IPRM as the arguments (use FPRM if an argument is of type REAL). The words of the array are used in ascending order from left to right in the subroutine argument list. (See Appendix E, UNIPLOT V2.1 Reference/User Guide, Publication No. 60454730, Revision C, for a description of Device Mode Subroutine)

IPRM(1) = name of subroutine; A10 format, left-justified

IPRM(2)-IPRM(13) = subroutine arguments

FPRM(2)-FPRM(13) = subroutine arguments

REORIGIN REQUEST (IREQ=5)

This request is part of the overplot feature and is intended to be used by CRT-type displays only. Whenever a new drawing (reorigin) occurs in the neutral picture file, the overlay for a CRT display must decide whether or not to erase the current contents of the display. This decision is made by testing the overplot flag.

IPRM(1) = overplot flag
0 = do not overplot; erase
1 = overplot; do not erase

INCREMENT REQUEST (IREQ=6)

This request passes the value of the plotter increment size, as input by an INCREMENT post processor directive to the overlay. The overlay may, on the basis of this value, call FACTOR to adjust the size of the finished plot. If no INCREMENT directive has been processed, the value of the plotter increment will be the default increment as set by the device dependent routines. If IREQ=6, the variable containing the increments per inch should be modified accordingly for any future initialization.

FPRM(1) = plotter increment size
(i.e., .01, .005, or .0025)

PEN RESTORE (IREQ=7)

This request instructs the device overlay to call the device dependent PLOT routine to move the pen to the bottom of the plotter as follows:

```
    CALL PLOT(0.0,-30.0,-3)
```

There are two ways a user can add a new device overlay to UNIPOST. The first adds the overlay to UNIPOST source code and the second merges relocatable binaries for UNIPOST and the new device overlay. Both methods require that an installation job be run first, even if no post processors are to be installed.

NOTE

> Neither job writes a back-up tape of REL66 with the new device code merged into the binaries or program library.

The first method requires running the following job:

```
ADDEV1.
USER.                    (Supply valid USER statement.)
LABEL(OLDPL,L=UNIPLOTV3,PO=R,R,
            ⎧ HY ⎫
         D= ⎨ HD ⎬ , VSN=REL66)
            ⎩ PE ⎭
UPDATE(Q,*==,D,8)
SUBMIT(COMPILE)
7/8/9
=IDENT ANY
=DEFINE TEK (Must be included if Tektronix post-
                  processor is installed)
=DELETE ADD.XX
USER.                    (Supply valid USER statement.)
=DELETE ADD.XX
*IDENT device
*DELETE DEVLIST.XX
          .
          :              (Insert changes to DEVLIST.)
*ADDFILE
*DECK device
          .   (Insert interface program and device
          :    dependent routines here.)
6/7/8/9
```

The second method requires running the following job:

```
ADDEV2.
USER.                    (Supply valid USER statement.)
ATTACH(OLDPL=UNV30PL)
UPDATE(Q)
ATTACH,UNI3=POSTLGO.  (Saved during the install
                       procedure.)
ATTACH,LIB1=PLOTLIB.  (Attach your device depend-
                       ent routines.)
FTN,I.
FTN,B=OVL.
REWIND(LGO,OVL)
COPYL,UNI3,LGO,UNI3A.
ATTACH(UNIPOST/M=W)
FILE(PLOTF,RT=S,BT=C)
LDSET(FILES=PLOTF)
LOAD(UNI3A)
LOAD(OVL)
SATISFY(LIB1)
NOGO.
7/8/9
*IDENT device
*DELETE DEVLIST.XX
          .
          :              (Insert changes to DEVLIST.)
*COMPILE DEVLIST
7/8/9
          .
          :              (Insert interface program.)
6/7/8/9
```

CALCOMP 906 POST PROCESSOR OPTION

RELEASE DESCRIPTION

The CalComp 906 post processor option runs under the NOS operating system when installed with UNIPLOT 3.

HARDWARE CONFIGURATION

UNIPLOT 3 with the CalComp 906 post processor option requires the same minimum hardware configuration as NOS. A CalComp 906 controller and a compatible plotter must be available for displaying graphic data.

RELEASE MATERIALS

The CalComp 906 post processor option resides on the tape known as REL66A. REL66A has the following characteristics: 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, CAL906 as file ID in the HDR1 label, six files:

File 1 - empty,

File 2 - empty,

File 3 - empty,

File 4 - CAL906 source for the device dependent routines,

File 5 - empty, and

File 6 - empty.

INSTALLATION PROCEDURE

The CalComp 906 post processor option must be installed with UNIPLOT 3. The installation procedure is contained on File 1 of the UNIPLOT 3 tape, REL66. The name CAL906 is defined in the installation job to add the CalComp 906 post processor code into the UNIPLOT system.

(Reference "Installation Procedure" for UNIPLOT 3.)

INSTALLATION VERIFICATION

To verify the installation of UNIPLOT 3 with the Cal Comp 906 post processor option, the following job should be run:

```
VERIFY.
USER.                                   (HY)
LABEL(OLDPL,L=UNIPLOTV3,PO=R,R,D=       {HD} ,
  VSN=REL66)                            (PE)
SKIPF(OLDPL,4)
UPDATE(Q,*= =,D,8,R)
REWIND(COMPILE)
SUBMIT(COMPILE)
7/8/9
=IDENT ANY
=DELETE VERIFY.XX
USER. (Supply valid USER statement.)
=CØMPILE VERIFY
6/7/8/9
```

The verification procedure saves a direct access neutral picture file (NPFILE) for subsequent post processing. To post process the neutral picture file, run the following job interactively:

```
ATTACH,NPFILE
ATTACH,UNIPØST
ASSIGN,TT,PLOTF
UNIPOST,D=CAL906,O=PLOTF
```

NOTE

When running under NAM/IAF, the cancel character and page width must be changed as follows before executing the above job: %CN=[and %PW = 131.

Figure I-13-1 shows the first picture generated when plotting begins.

HOUSTON INSTRUMENT BTC-7 POST PROCESSOR OPTION

RELEASE DESCRIPTION

The Houston Instrument BTC-7 post processor option runs under the NOS operating system when installed with UNIPLOT 3.

HARDWARE CONFIGURATION

UNIPLOT 3 with the Houston Instrument BTC-7 post processor option requires the same minimum hardware configuration as NOS. A Houston Instrument BTC-7 controller and a compatible plotter must be available for displaying graphics data.

RELEASE MATERIALS

The Houston Instrument BTC-7 post processor option resides on the tape known as REL66B. REL66B has the following characteristics: 7-track (800 bpi) or 9-track (800 or 1600 cpi, binary recording mode, HI200 as file ID in the HDR1 label, six files:

File 1 - empty,

File 2 - empty,

File 3 - empty

File 4 - HI200 source for the device dependent routines,

File 5 - empty, and

File 6 - empty.

INSTALLATION PROCEDURE

The Houston Instrument BTC-7 post processor option must be installed with UNIPLOT 3. The installation procedure is contained on file 1 of the UNIPLØT 3 tape, REL66. The name HI200 is defined in the installation job to add the Houston Instrument BTC-7 post processor code into the UNIPLOT system.

(Reference "Installation Procedure" for UNIPLOT 3.)

INSTALLATION VERIFICATION

To verify the installation of UNIPLOT 3 with the Houston Instrument BTC-7 post processor option, the following job should be run:

```
VERIFY.
USER. (Supply valid USER statement.)    (HY)
LABEL(OLDPL,L=UNIPLOTV3,PO=R,R,D=       {HD} ,
  VSN=REL66)                            (PE)
SKIPF(OLDPL,4)
UPDATE(Q,*= =,D,8,R)
REWIND(COMPILE)
SUBMIT(COMPILE)
7/8/9
=IDENT ANY
=DELETE VERIFY.XX
USER. (Supply valid USER statement.)
=COMPILE VERIFY
6/7/8/9
```

The verification procedure saves a direct access neutral picture file (NPFILE) for subsequent post processing. To post process the neutral picture file run the following job via batch:

```
JOB.
USER. (Supply valid USER statement.)
ATTACH,NPFILE.
ATTACH,UNIPOST.
UNIPOST,D=HI200.
REWIND,PLOTF.
COPYBF,PLOTF.
6/7/8/9
```

Figure I-13-1 shows the first picture generated when plotting begins.

TEKTRONIX 401X POST PROCESSOR OPTION

RELEASE DESCRIPTION

The Tektronix 401X post processor option runs under the NOS operating system when installed with UNIPLOT 3 using either the Time-sharing Module or NAM/IAF.

HARDWARE CONFIGURATION

UNIPLOT 3 with the Tektronix 401X post processor option requires the same minimum hardware configuration as NOS. A Tektronix 401X series terminal must be available for displaying graphics data.

RELEASE MATERIALS

The Tektronix 401X post processor option resides on the tape known as REL66C. REL66C has the following characteristics: 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, TEKTRN as file ID in the HDR1 label, six files:

File 1 - empty,

File 2 - empty,

File 3 - relocatable binaries of the Tektronix device dependent code,

File 4 - TEKTRN source for the device dependent routines,

File 5 - empty, and

File 6 - empty.

INSTALLATION PROCEDURE

The Tektronix 401X post processor option must be installed with UNIPLOT 3. The installation procedure is contained on File 1 of the UNIPLOT 3 tape, REL66. The name TEK is defined in the installation job to add the Tektronix 401X post processor code into the UNIPLOT system. (Reference "Installation Procedure" for UNIPLOT 3.)

INSTALLATION VERIFICATION

To verify the installation of UNIPLOT 3 with the Tektronix 401X post processor option, the following job should be run:

```
VERIFY.
USER. (Supply valid USER statement.)         ⎧ HY ⎫
LABEL(OLDPL,L=UNIPLOTV3,PO=R,R,D=           ⎨ HD ⎬ ,
  VSN=REL66)                                 ⎩ PE ⎭
SKIPF(OLDPL,4)
UPDATE(Q,*=  =,D,8,R)
REWIND(COMPILE)
SUBMIT(COMPILE)
7/8/9
=IDENT ANY
=DELETE VERIFY.XX
USER. (Supply valid USER statement.)
 =COMPILE VERIFY
6/7/8/9
```

The verification procedure saves a direct access neutral picture file (NPFILE) for subsequent post processing. To post process the neutral picture file, run the following job interactively:

```
ATTACH,NPFILE
ATTACH,UNIPOST
UNIPOST,D=TEK
```

NOTE

When running under NAM/IAF, the cancel character must be changed to [before executing UNIPOST. Type in %CN=[to change the cancel character before running the above job.

Figure I-13-1 shows the first picture generated when plotting begins.

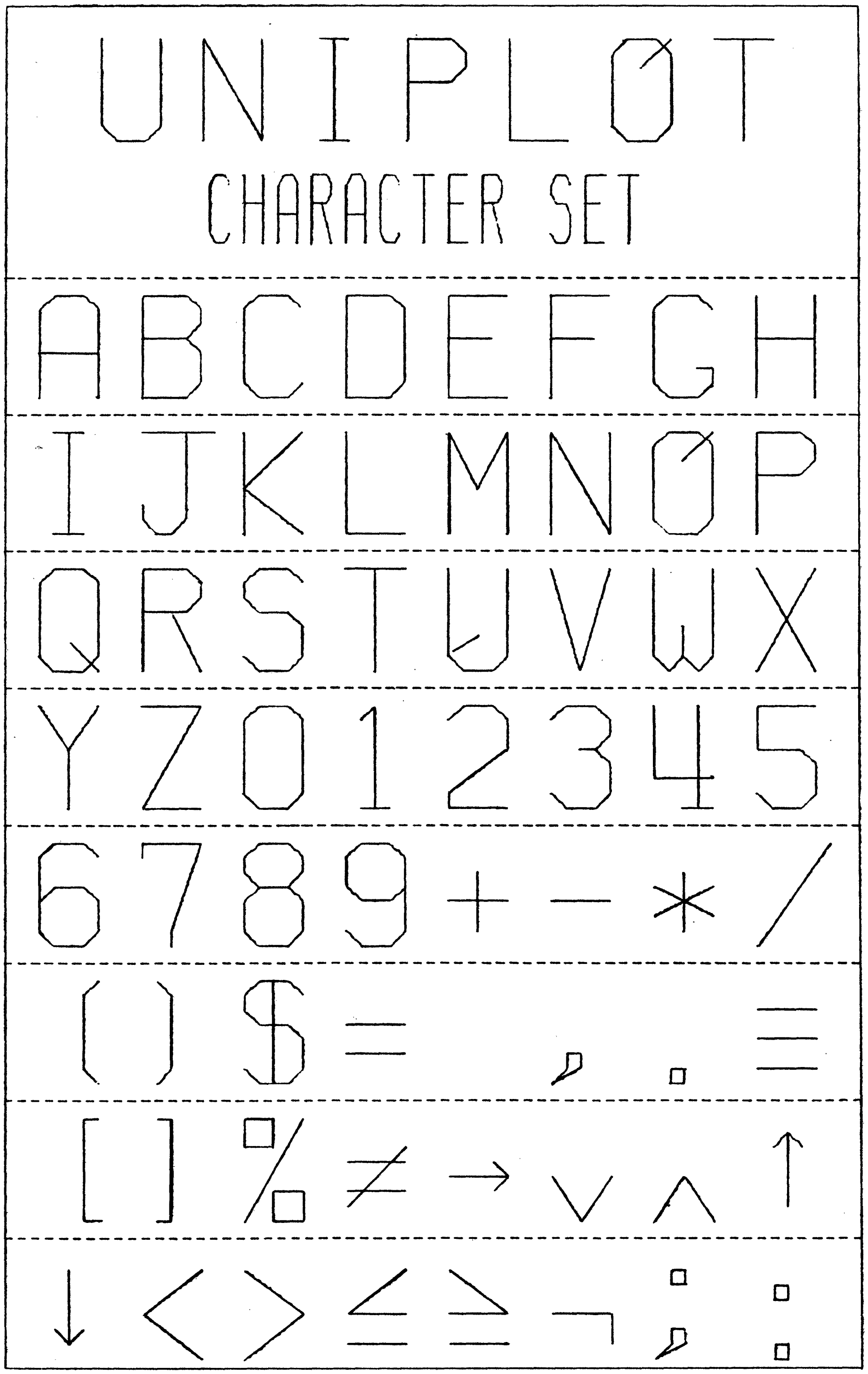


Figure I-13-1. The First Picture Generated When Plotting Begins

RELEASE DESCRIPTION

The Usage Accounting Utility release tape (AP001N) for NOS 1 consists of the following three files:

File 1 - the installation deck PL containing the UAUINS job which installs the software necessary to provide accounting for usage priced software.

File 2 - RANDR (Update/data reduction/report generation program) - this program reads user prepared input directive cards and performs the functions specified by the directives. The functions, all relating to the product file and the usage detail file are:

1. Maintain product file data base.
2. Reduce data on the usage detail file and produce detail usage summary/billing reports as per directives.
3. Generate system status reports.

File 3 - ACOUNTX - (Usage pricing accounting module) - this set of subroutines provides the accounting interface to usage priced applications. ACOUNTX tracks application usage and logs that usage on the usage detail file.

RANDR and ACOUNTX are to be installed on the SYSTEM library SYSLIB.

HARDWARE REQUIREMENTS

RANDR can be maintained on the same minimum hardware configuration as NOS 1. The amount of core over and above that required by the calling application depends upon whether or not the calling application uses the CYBLR record manager (CRM), the common memory manager (CMM), and/or is written in FORTRAN Extended Version 4 (FTN-4) (see table I-14-1).

TABLE I-14-1. ADDITIONAL CORE REQUIREMENTS

Application Characteristics			Maximum Additional Core Required for ACOUNTX
CRM	FTN-4	CMM	(In Octal)
YES	YES	YES	2.5K
YES	YES	NO	3.1K
YES	NO	YES	3.6K
YES	NO	NO	4.2K
NO	YES	YES	12.5K
NO	YES	NO	13.1K
NO	NO	YES	13.6K
NO	NO	NO	14.2K

CORRECTIONS

There are no applicable PSR summaries.

DEFICIENCIES

It is imperative that at least one billing run be made in the month of January to prevent corruption of the year-to-date totals.

Notes and Cautions

When loading an application calling ACOUNTX, nonfatal load errors may occur due to duplicate system entry points. Other errors should be investigated.

INSTALLATION PARAMETERS

Upon installation (creation) of the two necessary files, the product file and the usage detail file, various protection parameters may be specified on the input card (ID=1) specifying UPDATE (see Usage Accounting Utility Reference Manual).

Column	Contents
1 to 4	CPU model code; this is a four-character code associated with the machine type and will not be changed after the files are established. (The codes are found in appendix C of the Usage Accounting Utility Reference Manual.) The customer, in the event of a CPU model upgrade (for example, a CYBER 172 with a 10316-1 upgrade is a CYBER 173), is obligated to perform a final billing run on the existing product and detail files, remove those files, and recreate them with the appropriate CPU model code.
5	Blank; reserved.
6	Threshold protection indicator; Y specifies that special usage accumulator entries will be built, updated, and maintained in the usage detail file; accumulators, thresholds and product status (active or inactive) are checked and/or updated at start and end of an application. N specifies no special threshold checking is to be done. Blank defaults to N.

Column Contents

This parameter may be changed at any time. However, altering it from Y to N and vice versa between billing runs corrupts the effectiveness of threshold protection.

The special usage accumulator entries contain a usage accumulator. During a billing run, which covers a period of at least one full month, the accumulator is decremented by the invoiced amount of usage.

The special usage accumulator entries are lost when a billing run is made with the threshold protection installed as N. They can be reestablished (initialized) via an UPDATE run with threshold protection set to Y.

NOTE

> Threshold protection is merely to assist the site in limiting their usage and is not intended in any way to limit the sites' liability for usage as reported in monthly billing runs.

7 to 9 Blank, reserved.

10 to 14 Right-justified number (blanks treated as zero). It is multiplied by 1000 and represents the size (in number of words) at which the operator is warned that the detail file size protect has been reached. No other action is performed. It is the site responsibility to make a billing run (see Usage Accounting Utility Reference Manual) to reduce and report the usage detail file.

This parameter may be changed at any time.

15 Blank, reserved.

16 User number field protection; Y specifies that the user number will be blanked on all detail usage reports.

N specifies that user numbers for each accounting job step will appear on the detail usage reports.

Blank defaults to N.

This parameter may be changed at any time.

The following information represents directives to RANDR and is not considered installation parameters.

Column Contents

17 Blank, reserved.

18 1; represents parameter type card.

19 1 to 9; represents number of copies of report.

20 to 25 UPDATE

26 to 80 Blank, reserved.

NOTE

> After installation creation, any of the above installation parameters may be changed except the CPU model code without removing and recreating files.

Also upon installation (creation) of the two necessary files (product and usage detail), a Control Data regional address must be installed. The Control Data address to be used is specified on the front of the software contract. In addition, the customer is obligated to supply their address and the name of an individual as a primary contact for resolution of any matters which may arise pertaining to the preparation, content and submission of the usage reports. The address information is specified on the input cards (ID=2,3) to the creation run and may be altered in part or whole by update runs.

For each specific usage priced product, there is additional installation information to be specified on the product activity card (ID=4); vendor code, product name, transaction codes, threshold protection values, and software codes. The vendor code, product name, and software code for Control Data usage priced products are supplied with the installation information; see specific application product description in this manual. Threshold protection values are supplied by the site.

(For further description of the input card types, see the Usage Accounting Utility Reference Manual.)

INSTALLATION PROCEDURE

AP001N contains the usage accounting utility. The RANDR and ACOUNTX modules are SYSEDIT'ed into the running system.

The following job is used to retrieve the installation job UAUINS from AP001N. This deck is not available on REL2A.

```
UAUID(CM60000,T100)
USER(usernumber,password)
COMMENT.  THIS JOB SUBMITS A JOB TO INSTALL
COMMENT.  THE USAGE ACCOUNTING UTILITY
COMMENT.  INTO THE RUNNING SYSTEM
COMMENT.  **XX=MT/7-TRACK OR NT/9-TRACK
COMMENT.  **YY=800/800 bpi OR 1600/1600 cpi
LABEL(UAUTIL,D=YY,XX,F=I,VSN=AP001N,
   FI=USAGEACCTNG1P0,R)
UPDATE(P=UAUTIL,*==)
SUBMIT(COMPILE)
7/8/9 - EOR -
UPDATE DIRECTIVES
              THE UPDATE DIRECTIVES VARY
              DEPENDING UPON THE DENSITY
              AND TRACK TYPE BEING USED.
              =DEFINE MT800-7 TRACK, 800 bpi
              =DEFINE NT800-9 TRACK, 800 cpi
              =DEFINE NT1600-9 TRACK, 1600 cpi
6/7/8/9 - EOI -
```

The usage accounting utility is installed on the running system from AP001N by running job UAUINS which is executed by the above job UAUID. (Refer to the section on "Installation Job" for a listing of UAUINS.)

INSTALLATION JOB

Prior to running the UAUINS job, the user number ACXLIB must be established in the VALIDUx file. RANDR establishes the product file and the usage detail file under ACXLIB. If ACXLIB is not established or if usage accounting files do not exist under ACXLIB, the applications interface ACOUNTX will not run.

(The method of establishing ACXLIB on VALIDUx file is presented in the NOS Installation Handbook, part IV, section 1.1.2.)

UAUINS is the job which installs the usage accounting utility binaries on the running system. This job must be run under user number - ACXLIB.

```
UAUINS,CM60000,T100.
USER,ACXLIB.
COMMENT.   (This job must be run under user
           number ACXLIB.)
COMMENT.   (This job will SYSEDIT the usage ac-
COMMENT.   counting utility binaries into the
COMMENT.   running system.)
COMMENT.   **XX=MT/7-TRACK or NT/9-TRACK
COMMENT.   **YY=800/800 bpi or 1600/1600 cpi
LABEL(UAUTIL,D=YY,XX,F=I,VSN=AP001N,
   FI=USAGEACCTNG1P0,R)
REWIND(UAUTIL)
SKIPF(UAUTIL,1)
COPYBF(UAUTIL,RANDR,1)
COPYBF(UAUTIL,ACOUNTX,1)
REWIND(RANDR)
REWIND(ACOUNTX)
COMMON(SYSTEM)
GTR(SYSTEM,S)ULIB/SYSLIB
LIBEDIT(P=S,B=ACOUNTX,N=NEW,V)
REWIND(NEW)
LIBGEN(F=NEW,P=SYSLIB)
SYSEDIT(L)
CATALOG(SYSTEM,N,R)
7/8/9 - EOR -
*TYPE REL
*B*,ACOUNTX-*
7/8/9 - EOR -
*FILE SYSLIB
*FILE RANDR
6/7/8/9 - EOI -
```

Once the RANDR and ACOUNTX modules are installed on the running system, a new deadstart tape of the running system should be created.

Prior to execution of any usage priced applications, the following job UAFINS must be run. This job establishes the product file, initializes the usage detail file, and creates the product file entries for leased CDC usage priced application(s).

```
UAFINS(CM74000,T100)
USER(ACXLIB)
COMMENT.   (This job must be run under user
           number ACXLIB.)
COMMENT.   (This job establishes the product file and
COMMENT.   initializes the usage detail file.)
RANDR.
7/8/9 - EOR -
           (The data deck contents vary depending
           upon the application(s) being installed.
           For this initial run there must be 1
UPDATE     (ID=1) parameter card 3 (ID=2) CDC
DATA       address cards, 3 (ID=3) customer ad-
  CARDS    dress cards, and at least 1 (ID=4)
           product card.  [See Usage Accounting
           Utility Reference Manual.])
6/7/8/9 - EOI -
```

PRODUCT AND USAGE DETAIL FILES

The product file is maintained as a direct access permanent file and is defined as follows:

```
pfn       - AUAPTF
password  - JJFWGR
user no   - ACXLIB
```

The usage detail file is also maintained as a direct access permanent file and is defined as follows:

```
pfn       - AUADTF
password  - JJFWGR
user no   - ACXLIB
```

Because of the importance of these files, every attempt has been made to ensure that they are not lost during a RANDR run due to some external job interruption.

During a RANDR run, two other permanent file names become associated with AUAPTF (PFNEW, PFTMP) and AUADTF (DFNEW, DFTMP). The passwords, etc., are the same. Completion of RANDR purges these working files; noncompletion of RANDR (e.g., parity error, system abort, unexpected operator intervention) may cause them to be left as part of the permanent file base.

As a RANDR modifies product file entries, the file (AUAPTF) is being rewritten as PFNEW. DFNEW has multiple uses. PFTMP is the AUAPTF file renamed and represents the product file prior to the RANDR run. Likewise, DFTMP represents the AUADTF file prior to the RANDR run.

In the event of an external job interruption, the recommended recovery procedure is to purge all files except the PFTMP and DFTMP files, change them to AUAPTF and AUADTF, respectively, and rerun RANDR. In the event that one or both of the xxTMP file(s) are not found, but the xxNEW file(s) are present no action is necessary other than to purge the xxNEW file(s) and rerun RANDR.

It is further recommended that the site maintain adequate backup for these files (AUAPTF, AUADTF) through their DUMPF utility.

RELEASE DESCRIPTION

XEDIT Version 3 (XEDIT 3) is an extended interactive text editor developed by the University of Minnesota.

HARDWARE REQUIREMENTS

XEDIT 3 can be maintained on the same minimum hardware configuration as NOS 1. The minimum field length to run XEDIT 3 is 12,000$_8$.

CORRECTIONS

None.

DEFICIENCIES

None.

RELEASE MATERIALS

REL61 contains the XEDIT 3 release materials. It has the following characteristics: 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, VER3P0 as file ID in HDR1 label, six files:

File 1 - installation/verification decks in UPDATE format,

File 2 - absolute binary code and XEDIT Help file,

File 3 - empty,

File 4 - program library of XEDIT in MODIFY format,

File 5 - verification test procedure, and

File 6 - output produced by XEDIT test.

INSTALLATION PROCEDURE

XEDIT 3 is installed under user number LIBRARY by entering the following commands at the system console after mounting the XEDIT 3 release tape.

```
X.DIS
SUI(377776)
               { MT }      { 800  }
LABEL(TAPE,D=  { NT } ,D=  { 1600 } ,VSN=REL61)

UPDATE(Q,P=TAPE,C=INSJOB,L=1,D,8)
*COMPILE INSJOB
CALL,INSJOB(IOPT=1)
      -or-
CALL,INSJOB(IOPT=2)
```

If IOPT=1 is used, XEDIT 3 will be installed from the ABS binary code on file 2 of the release tape.

If IOPT=2 is used, XEDIT 3 will be installed by assembling the XEDIT source on file 4 of the XEDIT 3 release tape.

The XEDIT 3 installation procedure results in the following files added to the user number LIBRARY.

File 1 - XEDIT - direct access file (XEDIT binary), and
File 2 - XEDI64 - indirect access file (XEDIT help file).

INSTALLATION PARAMETERS

None.

VERIFICATION PROCEDURE

The XEDIT 3 verification procedure must be executed as a time-sharing origin (TXOT) type job. The following commands must be entered from a time-sharing terminal to execute the verification procedure.

```
BAT.
ATTACH,XEDIT/UN=LIBRARY.
               { MT }      { 800  }
LABEL(TAPE,D=  { NT } ,D=  { 1600 } ,VSN=REL61)

UPDATE(Q,P=TAPE,C=VFYJOB,L=1,D,8)
*COMPILE VFYJOB
CALL,VFYJOB.
```

The verification procedure will execute the XEDIT test and compare the test results with the XEDIT output on file 6 of the release tape. The results of the verification will be output to the terminal.

PDS/MAGEN 16

RELEASE DESCRIPTION

PDS/MaGen runs under NOS to provide the user with matrix generating and report writing capabilities for use in conjunction with APEX-III. PDS/MaGen is designed to execute from permanent files.

HARDWARE REQUIREMENTS

PDS/MaGen requires the same minimum hardware configuration as NOS. A minimum field length of 70K is required for installation.

DEFICIENCIES

None.

RELEASE MATERIALS

PDS/MaGen resides on the tape known as REL65. It has the following characteristics: 7-track (800 bits per inch [bpi]) or 9-track (800 or 1600 characters per inch [cpi]), binary recording mode, PDSMAGENV1P3B as file ID in HDR1 label, and six files:

File 1 - installation deck in UPDATE PL form,

File 2 - absolute binaries of PDS/MaGen,

File 3 - relocatable binaries of PDS/MaGen,

File 4 - empty,

File 5 - verification deck in UPDATE PL form, and

File 6 - sample output from verification run.

Listings of the installation deck and the verification deck may be obtained by executing the following job:

```
LIST, T50.
USER. (Supply valid USER card.)           ( HY )
LABEL(OLDPL,R,L=PDSMAGENV1P3B,D=         { HD },
  VSN=REL65)                              ( PE )
UPDATE(F,*= =,L=7)
SKIPF(OLDPL,4,17)
UPDATE(F,*= =,L=7,R)
7/8/9
6/7/8/9
```

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURE

PDS/MaGen is installed by executing deck PDSMC1, contained on the first file of REL65. This job deck reads the PDS/MaGen system from REL65, installs an executable file of the PDS/MaGen system on a permanent file name PDS with the password PDS.

The deck PDSMC1 contains UPDATE IF, DEF directives to simplify installation. The names used by these directives and their meaning when DEFINED are as follows:

Name	Effect on Installation Job
MT	Uses 7-track, 800 bpi, magnetic tape
HD	Uses 9-track, 800 cpi, magnetic tape
PE	Uses 9-track, 1600 cpi, magnetic tape

PDSMC1 may be extracted from REL65 and submitted for execution via the following job. Note that PDSMC1 requires a change for execution. A valid user card must be supplied.

```
INST, T50.
USER. (Supply valid USER card.)           ( HY )
LABEL(OLDPL,R,L=PDSMAGENV1P3B,D=         { HD },
  VSN=REL65)                              ( PE )
UPDATE(Q,*= =,D,8)
SUBMIT(COMPILE,B)
7/8/9        ( MT )
=DEFINE     { HD }
             ( PE )
=D,PDSMC1.3
USER. (Supply valid USER card.)
=C,PDSMC1
7/8/9
6/7/8/9
```

INSTALLATION VERIFICATION

The following job may be used to extract job PDSMCV from REL65 and submit for execution to verify installation of PDS/MaGen.

```
VER,T50.
USER. (Supply valid USER card.)           ( HY )
LABEL(OLDPL,R,L=PDSMAGENV1P3B,D=         { HD },
  VSN=REL65)                              ( PE )
SKIPF(OLDPL,4,17)
UPDATE(Q,*= =,D,8,R)
SUBMIT(COMPILE,B)
7/8/9
=D,PDSMCV.3
USER. (Supply valid USER card.)
=C,PDSMCV
7/8/9
6/7/8/9
```

File 6 of REL65 can be copied to OUTPUT and used to validate correct execution of the verification run.

INTRODUCTION 1

INSTALLATION AND MODIFICATION

Part II presents information pertaining to the installation of applications software under NOS/BE. While the information in this section describes the application installation, users should reference the "Introduction" to NOS/BE Installation Handbook for additional information.

The application products presented are intended for use only as described in this part. Control Data cannot be responsible for the proper functioning of undescribed features or parameters.

RELEASE TAPES

The following application products are available under NOS/BE:

Application	Tape	Section
APEX III Out-of-Core System 1	PL45	2
APEX III Mixed Integer Programming Option	PL51	2
APEX III Matrix Reduction Option	PL52	2
APEX III Parametrics Option	PL53	2
APEX III Usage Package 1	PL84	2
APT IV 2	PL59A	3
GPSS V/6000 1.2	PL35	4
IMSL 6	PL87	5
LCGT/IGS 2	PL67	6
PERT/TIME 2	PL85	7
777/IGS V2 Host	PL40	8
777/IGS V2 Remote	PL41	8
777/IGS V2 with 3D (Host)	PL46	8
777/IGS V2 with 3D (Remote)	PL47	8
SIMSCRIPT I.5 3	PL27	9
TIGS 1	PL86A	10
Tektronix 401X Post Processor under TIGS	PL86B	10
Sanders Graphic 7 Post Processor under TIGS	PL86C	10
TOTAL Universal 1	PL49	11
TOTAL Universal 2	PL75	11
TOTAL/ATHENA	PL62	12
UNIPLOT 2	PL71	13
Usage Accounting Utility	AP001B	14

RELATED DOCUMENTATION

The following manuals contain relevant information on NOS/BE and/or the application products described in part II.

Title	Publication No.
NOS/BE Installation Handbook	60494300
APEX III Reference Manual	76070000
Application Executive Reference Manual	17322200
APT IV V2 Reference Manual	17326900
APT IV Internal Maintenance Specification	†
Beginning Graphics Users Guide	76077300
Data Handler Reference Manual	17322100
GPSS V/6000 General Information Manual	84003900
IMSL Reference Manual (Volumes 1-2)	SMD150043†
LCGT/IGS Reference Manual	76079100
LCGT/IGS User's Guide	76077400
PERT/TIME General Information Manual	60133300
PERT/TIME Reference Manual	60133600
777/IGS Reference Manual	17321800
777 IGS Remote Job Entry User Guide	76077200
777 IGS User Guide	17322500
777/3D IGS Reference Manual	17326500
SIMSCRIPT I.5 Reference Manual	60358500
TIGS V1 Reference Manual	60455940
TOTAL Universal Reference Manual	76070300
UNIPLOT User Guide	76079600
Usage Accounting Utility	84000440
1743-1 Reference Manual	89638400

†These manuals may be ordered by title from Control Data Corporation, Software Manufacturing and Distribution, 4201 North Lexington Avenue, St. Paul, Minnesota 55112.

APEX-III PRODUCT OPTIONS 2

APEX-III OUT-OF-CORE SYSTEM 1

RELEASE DESCRIPTIONS

APEX-III Out-of-Core System 1 runs under the NOS/BE operating system. The FORTRAN Extended compiler is required for compiling APEX-III. APEX-III Out-of-Core System 1 is designed to execute from permanent files.

Hardware Requirements

APEX-III Out-of-Core System 1 requires the same minimum hardware configuration as NOS/BE except that a minimum field length of 100K octal is required to compile the system and a minimum of 64K octal is required to execute APEX-III.

Deficiencies

None.

Release Materials

APEX-III Out-of-Core System 1 resides on the tape known as PL45, in either 7-track (800 bpi) or 9-track (800 or 1600 cpi) format, has label OOCAPEXIIIV1P1 and contains six binary files.

File 1 - installation deck in UPDATE PL form,
File 2 - empty,
File 3 - empty,
File 4 - source code in UPDATE PL form,
File 5 - verfication deck in UPDATE PL form, and
File 6 - sample output from verification run.

Listings of the installation deck (APEXC1) and the verification deck (APEXCV) may be obtained from PL45 by executing the following job:

```
        (MT1)
LIST,   {HD1}  .
        (PE1)
ACCOUNT.          (Supply valid ACCOUNT card.)
                                              (HY)
LABEL(OLDPL,R,L=OOCAPEXIIIV1P1,D=             {HD}  ,
                                              (PE)
   VSN=APXOOC)
UPDATE(F,*==,L=7)
SKIPF(OLDPL,4,17)
UPDATE(F,*==,L=7,R)
7/8/9
6/7/8/9
```

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURE

APEX-III Out-of-Core System 1 is installed by executing deck APEXC1, contained on the first file of PL45. This job deck reads the APEX-III system from PL45, compiles the system, and installs an executable file of the APEX-III system on a permanent file named APEXIII with the ID APEX.

This job deck also creates three permanent files which are needed when one uses the APEX-III USER feature. (Reference the APEX-III Reference Manual for detailed information.) For those installations where the USER feature is used infrequently, it may be desirable to maintain these three files on magnetic tape rather than permanent files. These three files are described below:

Name	ID	Contents
APEXPL	APEX	Partial APEX-III source code. Contains APEX-III COMMON regions and subroutines QBUFFER, QGET, QUSER, and QUSERDA.
APEXOV	APEX	Relocatable binary code of APEX-III overlay drivers.
APEXRL	APEX	Relocatable binary code of remaining APEX-III subroutines.

For those installations which have purchased any of the APEX-III options, deck APEXC1 may also be used to install the APEX-III Out-of-Core System 1 with any combination of the APEX-III Mixed Integer Programming Option (MIP), APEX-III Matrix Reduction Option, and APEX-III Parametrics Option.

The deck APEXC1 contains UPDATE IF,DEF directives to simplify installation. The names used by these directives and their meaning when DEFINEd are as follows:

Name	Effect on Installation Job
MT	Uses 7-track magnetic tapes.
HD	Uses 9-track, 800 cpi, magnetic tapes.
PE	Uses 9-track; 1600 cpi, magnetic tapes.
MIP	APEX-III Mixed Integer Programming Option is installed with APEX-III Out-of-Core System 1.
RED	APEX-III Matrix Reduction Option is installed with APEX-III Out-of-Core System 1.

Name	Effect on Installation Job
PAR	APEX-III Parametrics Option is installed with APEX-III Out-of-Core System 1.
CYB176	Required when installing on a CYBER 176.

APEXC1 may be extracted from PL45 and submitted for execution via the following job. Note that APEXC1 requires a change for execution. A valid ACCOUNT card must be supplied.

```
       (MT1)
INST,  {HD1}  .
       (PE1)
ACCOUNT.          (Supply valid ACCOUNT card.)
                                         (HY)
LABEL(OLDPL, R, L=OOCAPEXIIIV1P1, D=    {HD}  ,
                                         (PE)
   VSN=APXOOC)
REQUEST, COMPILE, *Q.
UPDATE(Q, *=-, D, 8)
ROUTE(COMPILE, DC=IN)
7/8/9
          (MT)
=DEFINE   {HD}
          (PE)
=DEFINE MIP      (If Mixed Integer Programming
                 Option.)
=DEFINE RED      (If Matrix Reduction Option.)
=DEFINE PAR      (If Parametrics Option.)
=DEFINE CYB176   (If installing on a CYBER 176.)
=D, APEXC1. 8
ACCOUNT.         (Supply valid ACCOUNT card.)
=C, APEXC1
7/8/9
6/7/8/9
```

Error Messages

A nonfatal loader error will be encountered in loading the APEX-III system. The error message is of the form:

```
 NE4102///DUPLICATE ENTRY POINT NAME - EXIT
PROGRAM NAME ------ FORSYS
LAST FILE ACCESSED - FORTRAN
```

Errors of any other form should be investigated.

Installation Verification

The following job may be used to extract job deck APEXCV from PL45 and submit for execution to verify installation of APEX-III Out-of-Core System 1.

```
      (MT1)
VER,  {HD1}  .
      (PE1)
ACCOUNT.          (Supply valid ACCOUNT card.)
                                        (HY)
LABEL(OLDPL, R, L=OOCAPEXIIIV1P1, D=   {HD}  ,
                                        (PE)
   VSN=APXOOC)
SKIPF(OLDPL, 4, 17)
REQUEST, COMPILE, *Q.
UPDATE(Q, *==, D, 8, R)
ROUTE(COMPILE, DC=IN)
7/8/9
=D, APEXCV. 3
ACCOUNT.          (Supply valid ACCOUNT card.)
=C, APEXCV
7/8/9
6/7/8/9
```

File 6 of PL45 can be copied to OUTPUT and used to validate correct execution of the verification run.

APEX-III MIXED INTEGER PROGRAMMING OPTION (MIP)

RELEASE DESCRIPTIONS

The APEX-III Mixed Integer Programming Option runs under NOS/BE when installed with APEX-III Out-of-Core System 1. The FORTRAN Extended compiler is required for compiling and updating APEX-III MIP. APEX-III MIP is designed to execute from permanent files.

Hardware Requirements

APEX-III MIP requires the same minimum hardware configuration as NOS/BE except that a minimum field length of 100K octal is required to compile the system and a minimum of 64K octal is required for execution.

Deficiencies

None.

Release Materials

The APEX-III MIP resides on the tape known as PL51. PL51, in either 7-track (800 bpi) or 9-track (800 or 1600 cpi) format, has label MIPAPEXIIIV1P1 and contains six binary files.

File 1 - not used,
File 2 - empty,
File 3 - empty,
File 4 - source code in UPDATE PL form,
File 5 - verification deck in UPDATE PL form, and
File 6 - Sample output from verification run.

A listing of the verification deck (APEXMV) may be obtained from PL51 by executing the following job:

```
       (MT1)
LIST,  {HD1}  .
       (PE1)
ACCOUNT.          (Supply valid ACCOUNT card.)
                                         (HY)
LABEL(OLDPL, R, L=MIPAPEXIIIV1P1, D=    {HD}  ,
                                         (PE)
   VSN=APXMIP)
SKIPF(OLDPL, 4, 17)
UPDATE(F, *==, L=7, R)
7/8/9
6/7/8/9
```

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURE

The APEX-III MIP must be installed with APEX-III Out-of-Core System 1. The installation deck, APEXC1, on the APEX-III Out-of-Core System 1 tape, known as PL45, is used to install the MIP Option. The name MIP is defined for the UPDATE IF,DEF directives in deck APEXC1 to merge the MIP Option with the out-of-core system.

(Reference "Installation Procedure" for APEX-III Out-of-Core System 1 for details of installation and expected error messages.)

Installation Verification

The following job may be used to extract job deck APEXMV from PL51 and submit for execution to verify installation of the MIP Option.

```
        ⎧MT1⎫
VER,    ⎨HD1⎬  .
        ⎩PE1⎭
ACCOUNT.          (Supply valid ACCOUNT card.)
                                       ⎧HY⎫
LABEL(OLDPL,R,L=MIPAPEXIIIV1P1,D=      ⎨HD⎬  ,
                                       ⎩PE⎭
   VSN=APXMIP)
SKIPF(OLDPL,4,17)
REQUEST,COMPILE,*Q.
UPDATE(Q,*==,D,8,R)
ROUTE(COMPILE,DC=IN)
7/8/9
=D, APEXMV.3
ACCOUNT.          (Supply valid ACCOUNT card.)
=C, APEXMV
7/8/9
6/7/8/9
```

File 6 of PL51 can be copied to OUTPUT and used to validate correct execution of the verification run.

APEX-III MATRIX REDUCTION OPTION (REDUCE)

RELEASE DESCRIPTIONS

The APEX-III Matrix Reduction Option runs under the NOS/BE operating system when installed with APEX-III Out-of-Core System 1. The FORTRAN Extended compiler is required for compiling and updating APEX-III REDUCE. APEX-III REDUCE is designed to execute from permanent files.

Hardware Requirements

APEX-III REDUCE requires the same minimum hardware configuration as NOS/BE except that a minimum field length of 100K octal is required to compile the system and a minimum of 64K octal is required for execution.

Deficiencies

None.

Release Materials

The APEX-III Matrix Reduction Option resides on the tape known as PL52. PL52, in either 7-track (800 bpi) or 9-track (800 or 1600 cpi) format, has label REDAPEXIIIV1P1 and contains six binary files.

File 1 - not used,
File 2 - empty,
File 3 - empty,
File 4 - source code in UPDATE PL form,
File 5 - verification deck in UPDATE PL form, and
File 6 - sample output from verification run.

A listing of the verification deck (APEXRV) may be obtained from PL52 by executing the following job:

```
        ⎧MT1⎫
LIST,   ⎨HD1⎬  .
        ⎩PE1⎭
ACCOUNT.          (Supply valid ACCOUNT card.)
                                       ⎧HY⎫
LABEL(OLDPL,R,L=REDAPEXIIIV1P1,D=      ⎨HD⎬  ,
                                       ⎩PE⎭
   VSN=APXRED)
SKIPF(OLDPL,4,17)
UPDATE(F,*==,L=7,R)
7/8/9
6/7/8/9
```

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURE

The APEX-III Matrix Reduction Option must be installed with APEX-III Out-of-Core System 1. The installation deck, APEXC1, on the APEX-III Out-of-Core System 1 tape known as PL45, is used to install the REDUCE Option. The name RED is defined for the UPDATE IF, DEF directives in deck APEXC1 to merge the REDUCE Option with the out-of-core system.

(Reference "Installation Procedure" for APEX-III Out-of-Core System 1 for details of installation and expected error messages.)

Installation Verification

The following job may be used to extract job deck APEXRV from PL52 and submit for execution to verify installation of the REDUCE Option.

```
      ⎧MT1⎫
VER,  ⎨HD1⎬  .
      ⎩PE1⎭
ACCOUNT.        (Supply valid ACCOUNT card.)
                                      ⎧HY⎫
LABEL(OLDPL,R,L=REDAPEXIIIV1P1,D=     ⎨HD⎬  ,
                                      ⎩PE⎭
   VSN=APXRED)
SKIPF(OLDPL,4,17)
REQUEST,COMPILE,*Q.
UPDATE(Q,*==,D,8,R)
ROUTE(COMPILE,DC=IN)
7/8/9
=D,APEXRV.3
ACCOUNT.        (Supply valid ACCOUNT card.)
=C,APEXRV
7/8/9
6/7/8/9
```

File 6 of PL52 can be copied to OUTPUT and used to validate correct execution of the verification run.

APEX-III PARAMETRICS OPTION

RELEASE DESCRIPTIONS

The APEX-III Parametrics Option runs under the NOS/BE operating system when installed with APEX-III Out-of-Core System 1. The FORTRAN Extended compiler is required for compiling and updating APEX-III Parametrics. APEX-III Parametrics is designed to execute from permanent files.

Hardware Requirements

APEX-III Parametrics requires the same minimum hardware configuration as NOS/BE except that a minimum field length of 100K octal is required to compile the system and a minimum of 64K octal is required for execution.

Deficiencies

None.

Release Materials

The APEX-III Parametrics Option resides on the tape known as PL53. PL53, in either 7-track (800 bpi) or 9-track (800 or 1600 cpi) format, has label PARAPEXIIIV1P1 and contains six binary files.

File 1 - not used,
File 2 - empty,
File 3 - empty,
File 4 - source code in UPDATE PL form,
File 5 - verification deck in UPDATE PL form, and
File 6 - sample output from verification run.

A listing of the verification deck (APEXPV) may be obtained from PL53 by executing the following job:

```
       ⎧MT1⎫
LIST,  ⎨HD1⎬  .
       ⎩PE1⎭
ACCOUNT.        (Supply valid ACCOUNT card.)
                                      ⎧HY⎫
LABEL(OLDPL,R,L=PARAPEXIIIV1P1,D=     ⎨HD⎬  ,
                                      ⎩PE⎭
   VSN=APXPAR)
SKIPF(OLDPL,4,17)
UPDATE(F,*==,L=7,R)
7/8/9
6/7/8/9
```

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURE

The APEX-III Parametrics Option must be installed with APEX-III Out-of-Core System 1. The installation deck, APEXC1, on the APEX-III Out-of-Core System 1 tape known as PL45, is used to install the Parametrics Option. The name PAR is defined for the UPDATE IF,DEF directives in deck APEXC1 to merge the Parametrics Option with the out-of-core system.

(Reference "Installation Procedure" for APEX-III Out-of-Core System 1 for details of installation and expected error messages.)

Installation Verification

The following job may be used to extract job deck APEXPV from PL53 and submit for execution to verify installation of the Parametrics Option.

```
      ⎧MT1⎫
VER,  ⎨HD1⎬  .
      ⎩PE1⎭
ACCOUNT.        (Supply valid ACCOUNT card.)
                                      ⎧HY⎫
LABEL(OLDPL,R,L=PARAPEXIIIV1P1,D=     ⎨HD⎬  ,
                                      ⎩PE⎭
   VSN=APXPAR)
SKIPF(OLDPL,4,17)
REQUEST,COMPILE,*Q.
UPDATE(Q,*==,D,8,R)
```

```
ROUTE(COMPILE,DC=IN)
7/8/9
=D,APEXPV.3
ACCOUNT.          (Supply valid ACCOUNT card.)
=C,APEXPV
7/8/9
6/7/8/9
```

File 6 of PL53 can be copied to OUTPUT and used to validate correct execution of the verification run.

APEX-III USAGE PACKAGE 1

RELEASE DESCRIPTIONS

APEX-III Usage Package 1 runs under the NOS/BE operating system. The FORTRAN Extended compiler is required for compiling APEX-III when using the USER feature. APEX-III is designed to execute from permanent files.

Installation of APEX-III Usage Package 1 requires that the Usage Accounting Utility be installed and that a product update run be made to add the APEX-III product to the accounting files. (Consult the Usage Accounting Utility Reference Manual for further details.)

The product activity card required for APEX-III has the following form:

```
 CDAPEX III   A A 4      APEX
```

Columns	Contents
1 to 2	CD is the vendor code.
3 to 12	APEX III is the product name.
14	A signifies add product information to product name file.
16	A signifies the product is active.
18	4 is the ID value.
20 to 23 25 to 28	Represent threshhold values-consult Usage Accounting Utility Reference Manual if usage of APEX-III is to be limited.
30 to 80	APEX is the software code.

Hardware Requirements

APEX-III Usage Package 1 requires the same minimum hardware configuration as NOS/BE except that a minimum field length of 100K octal is required for compilation for the USER feature and 70K octal is required for execution of APEX-III.

Deficiencies

None.

Release Materials

APEX-III Usage Package 1 resides on the tape known as PL84. PL84, in either 7-track (800 bpi) or 9-track (800 or 1600 cpi) format, has label AUUAPEXIIIV1P1 and contains seven binary files.

File 1 - installation deck in UPDATE PL format,
File 2 - relocatable binary code of APEX-III overlay drivers,
File 3 - relocatable binary code of remaining APEX-III system routines,
File 4 - partial source code in UPDATE PL form for USER feature,
File 5 - verification deck in UPDATE PL form,
File 6 - sample output from verification deck, and
File 7 - list file of documentation of selected APEX-III routines for USER feature.

Listings of the installation deck (APEXU1) and the verification deck (APEXUV) may be obtained from PL84 by executing the following job:

```
       (MT1)
LIST,  {HD1} .
       (PE1)
ACCOUNT.         (Supply valid ACCOUNT card.)
                                        (HY)
LABEL(OLDPL,R,L=AUUAPEXIIIV1P1,D=      {HD} ,
                                        (PE)
   VSN=APXAUU)
UPDATE(F,*==,L 7)
SKIPF(OLDPL,4,17)
UPDATE(F,*==,L 7,R)
7/8/9
6/7/8/9
```

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURE

APEX-III Usage Package 1 is installed by executing deck APEXU1, contained on the first file of PL84. This job deck reads relocatable binary code of the APEX-III system from PL84 and installs an executable file of the APEX-III system on a permanent file named UAPEX3 with ID=APEX.

This job deck will optionally create three permanent files which are needed when one uses the APEX-III USER feature. (See the APEX-III Reference Manual for detailed information.) For those installations where the USER feature is used infrequently, it may be desirable to reference these files directly from the installation tape when needed rather than maintaining them on permanent files. These three files are described as follows:

Name	ID	Contents
UAPEXPL	APEX	Partial APEX-III source code. Contains APEX-III COMMON regions and subroutines QBUFFER, QGET, QUSER and QUSERDA. Same as file 4 of installation tape.
UAPEXOV	APEX	Relocatable binary code of APEX-III overlay drivers. Same as file 2 of installation tape.
UAPEXRL	APEX do not copy record 2 to disk	Relocatable binary code of remaining APEX-III subroutines. Same as file 3 of installation tape except that record 2 is ignored for a CYBER 176; record 1 is ignored for other mainframes.

The deck APEXU1 contains UPDATE IF,DEF directives to simplify installation. The names used by these directives and their meaning when DEFINED are as follows:

Name	Effect on Installation Job
MT	Uses 7-track magnetic tapes.
HD	Uses 9-track, 800 cpi magnetic tapes.
PE	Uses 9-track, 1600 cpi, magnetic tapes.
PF	Files 2, 3, and 4 of the installation tape are installed as permanent files with names UAPEXOV, UAPEXRL, and UAPEXPL respectively for the USER feature.
CYB176	Required when installing on CYBER 176.

APEXU1 may be extracted from PL84 and submitted for execution via the following job. Note that APEXU1 requires a change for execution. A valid ACCOUNT card must be supplied.

```
      (MT1)
INST, {HD1} .
      (PE1)
ACCOUNT.          (Supply valid ACCOUNT card.)
                                       (HY)
LABEL(OLDPL,R,L=AUUAPEXIIIV1P1,D=      {HD} ,
                                       (PE)
   VSN=APXAUU)
REQUEST,COMPILE,*Q.
UPDATE(Q,*==,D,8)
ROUTE(COMPILE,DC=IN)
7/8/9
         (MT)
=DEFINE  {HD}
         (PE)
=DEFINE PF        (If files for USER feature are to be
                  on permanent files.)
=DEFINE CYB176 (If installing on a CYBER 176.)
=D,APEXU1.8
ACCOUNT.          (Supply valid ACCOUNT card.)
=C,APEXU1
7/8/9
6/7/8/9
```

Error Messages

A nonfatal loader error will be encountered in loading the APEX-III system. The error message is of the form:

```
   NE4102///DUPLICATE ENTRY POINT NAME - EXIT
  PROGRAM NAME ------ FORSYS=
  LAST FILE ACCESSED- FORTRAN
```

Errors of any other form should be investigated.

Installation Verification

The following job may be used to extract job deck APEXUV from PL84 and submit for execution to verify installation of APEX-III Usage Package 1:

```
     (MT1)
VER, {HD1} .
     (PE1)
ACCOUNT.          (Supply valid ACCOUNT card.)
                                       (HY)
LABEL(OLDPL,R,L=AUUAPEXIIIV1P1,D=      {HD} ,
                                       (PE)
   VSN=APXAUU)
SKIPF(OLDPL,4,17)
REQUEST,COMPILE,*Q.
UPDATE(Q,*==,D,8,R)
ROUTE(COMPILE,DC=IN)
7/8/9
=D,APEXUV.3
ACCOUNT.          (Supply valid ACCOUNT card.)
=C,APEXUV
7/8/9
6/7/8/9
```

File 6 of PL84 can be copied to OUTPUT and used to verify correct execution of the verification run.

RELEASE DESCRIPTION

APT IV Version 2 (APT IV 2) is a numerical control language processor designed to generate cutter location coordinates. The user has the capability to fully describe the part to be machined and then drive the tool along the chosen path. The output from APT IV 2 is a verification listing and a file for input to a post processor. APT IV 2 has all the capabilities of APT IV in addition to new language features and extensions.

APT IV 2 runs under the NOS/BE operating system on a CDC host computer having at least 65,000 words of central memory.

RELEASE MATERIALS

APT IV 2 resides on the tape known as PL59A. This release tape contains six files:

File 1 - installation/maintenance program library,

File 2 - absolute modules of the APT IV 2 system. The file should be placed on the system library if the user does not wish to modify the APT IV processor. File 3 of PL59A was used to make file 2.

File 3 - relocatable binary records of all routines needed for the standard mode; included are the binary records, COD4, PRD4, PTR4, POO4 and COO4 which are produced by the LOAD COMPLEX. This file is ready to be loaded by the host computer system and will produce the absolute modules contained on file 2.

File 4 - program library in UPDATE format. Following the YANK$$$ deck, the first 67 decks, TAPEOP through ZSSPICT, are common decks called by one or more of the succeeding decks. The next 16 decks, LLDM through LBCDBIN, are the LOAD COMPLEX. The next deck, LLCDATA, is data used by the LOAD COMPLEX routines. The next 93 decks, TAPT4T through TLCANON, are the TRANSLATOR. The next 378 decks, XAPT4E through XTRANSF57 are the EXECUTION COMPLEX. The next 45 decks, CAPT4C through CTAPEWT are the CLEDITOR. The last 18 decks, PAPT4P through PTAPEWT are the POST EXEC.

File 5 - Sample part program. Job control cards are provided for correct operation with system overlays resident in the system library, permanent files, or user library.

File 6 - Listing of the sample job, not including the dayfile.

INSTALLATION REQUIREMENTS

The system has four general modes of operation:

1. Standard mode - offers all capabilities except parametric and sculptured surfaces. This mode requires approximately $110,000_8$ words of field length to build the system and for execution.

2. Parametric surface mode - offers all capabilities of the standard system plus the added features of parametric surface processing. This mode does not include sculptured surface processing. Approximately $110,000_8$ words are required to build the system and ($110,000_8$ plus the length of the FORTRAN parametric surface routines) words for execution.

3. Sculptured surface mode - offers all capabilities of the standard system plus the added features of sculptured surfaces processing. This mode does not include parametric surface processing. Approximately $125,000_8$ words are required to build and execute the system.

4. Parametric and sculptured mode - both parametric and sculptured surface processing may be used. Approximately $125,000_8$ words are required to build the system and ($125,000_8$ plus the length of the FORTRAN Parametric Surface routine) words for execution.

INSTALLATION OPTIONS

The user may choose one of four modes:

1. Parametric surface,

2. Sculptured surface,

3. Parametric and sculptured surface, and

4. Neither parametric nor sculptured surface.

The release system operates with neither parametric nor sculptured surfaces.

In addition to the above mode, the user may choose one of three residences for the APT system:

1. system library,

2. permanent files, or

3. user library.

The released system operates from the system library.

The user may wish to have PSRs installed and/or local modifications incorporated.

PSR MODIFICATION

The PSR deck on the corrective code tape which is applicable to APT IV 2 is APT. The idents within this deck are of the form APTnnnn. The initial rerelease of PL59A is at LV420.

The APT IV V2 installation job assumes that the PSRs are on a permanent file called MINIT, ID=CCT. MINIT is assumed to be a program library with a master character of / (slash).

INSTALLATION PROCEDURES

The following control cards may be used to obtain a listing of the installation decks from PL59A:

```
LIST, { MT1. }
      { HD1. }
ACCOUNT.
LABEL(OLDPL,R,L=APT4V2,D= { HY } ,
   VSN=PL59A)             { HD }
UPDATE(F,*=$$$$,L=A1247)
7/8/9
6/7/8/9
```

The following DEFINE options are specifiable in the installation decks:

```
( STANDARD                                          )
( COMPLEX                                           )
( (SYSTEM,  )                                       )  (MT)
( {PERMFILE,} {PSR,} {SSSURF,} {PARSURF}            )  {NT}
( (USERLIB, )                                       )
```

STANDARD APT SYSTEM INSTALLATION

The standard APT IV 2 system is on file 2 of the release tape, PL59A. The standard APT system is installed into the host computer operating system via an EDITLIB (SYSTEM). The standard APT system does not include parametric or sculptured surfaces.

The selection of the standard APT system precludes the selection of any and all other parameters except MT/NT. The UPDATE deck to perform the installation of the standard system is as follows:

```
STAN. { MT1. }
      { HD1. }
ACCOUNT.
LABEL(OLDPL,R,L=APT4V2,D= { HY } ,
   VSN=PL59A)             { HD }
UPDATE(Q,*=$$$$,D,8)
ROUTE(COMPILE,DC=IN)
7/8/9
$/                          (If using a 9-track tape
                            insert the following:)
$IDENT     NOSBE4
$DELETE    STANDECK.10
MAPT03,HD1.
$/END IF
$IDENT,DECKSELC
$DELETE,STANDECK.11         (To change ACCOUNT
ACCOUNT.                    cards as needed.)
$DEFINE, { MT }
         { NT }
$DEFINE,STANDARD
$COMPILE,STANDECK
6/7/8/9
```

If the user selects to install the standard system, the user must be prepared to do system EDITLIBs.

The user defines MT for 7-track and NT for 9-track. If neither MT nor NT is defined, the APT installation assumes NT.

MODIFICATION OF THE LOAD COMPLEX ROUTINES

The deck COMPLEX may be used to modify the five relocatable decks produced by the LOAD COMPLEX. The names of the decks are COD4, PRD4, PTR4, POO4, and COO4. Once the user modifications are determined and inserted, the deck COMPLEX compiles the LOAD COMPLEX routines LLDM through LBCDBIN. The deck LCCDATA, which is to be modified by the user, is the input data for the LOAD COMPLEX execution. The output, five FORTRAN BLOCK DATA routines, is then compiled and the relocatable decks replace their counterparts on file 3 of the new release tape.

Next, the user should complete the installation of the APT system by running a second job to specify any other installation option.

NOTE

> Use of the COMPLEX option precludes the use of the standard option in this next job.

The deck SOURCDK will be used to complete the installation using the output tape of the previous job as the OLDPL for this next job.

A sample procedure to install modifications to the LOAD COMPLEX routines is as follows:

```
COMPLX, { MT1. }
        { HD1. }
ACCOUNT.
LABEL(OLDPL,R,L=APT4V2,D= { HY },
   VSN=PL59A)             { HD }
UPDATE(Q,*=$$$$,D,8)
ROUTE(COMPILE,DC=IN)
7/8/9
$/                      (If using 9-track tape
                        insert the following:)
$IDENT      NOSBE4
$DELETE     COMPLEX.7
MAPT01,HD2,T200.
$/END IF
$IDENT,COMPMOD
$DELETE,COMPLEX.8       (To change ACCOUNT
ACCOUNT.                cards as needed.)
$DEFINE, { MT }
         { NT }
$DEFINE,COMPLEX
$INSERT,COMPLEX.65
*IDENT,COMPLX
*INSERT,LITL.43
C  I  MODIFIED THIS WITH A COMMENT CARD
                        (Insert mods here as
                        needed.)
*COMPILE,LITL
$INSERT,COMPLEX,67
*IDENT,LCMOD
*DELETE,LLCDATA.2       (Insert mods to LLCDATA
, (15 spaces)   1000    here as needed.)
*COMPILE,LLCDATA
$COMPILE,COMPLEX
6/7/8/9
```

Before proceeding to the next job, review the results of this job to be sure that the intended modifications have been made correctly.

NONSTANDARD APT SYSTEM INSTALLATION

To perform nonstandard installation, the user:

1. must not select the option standard and

2. must not select the option COMPLEX during this run.

The user may, however, have selected COMPLEX for the previous run to generate a new input tape for this job.

The DEFINE options available for a nonstandard installation are:

```
{ SYSTEM,   }
{ PERMFILE, }  {PSR,}  {SSSURF,}  {PARSURF,}  { MT }
{ USERLIB,  }                                 { NT }
```

The user may select only one of SYSTEM, PERMFILE or USERLIB to specify whether the APT system is installed in the system library, as permanent files or as a user library.

The deck SOURCDK is used to install the nonstandard APT system.

In addition to selecting the type of residency, the user may DEFINE:

1. PSR - PSR code is to be installed and resides on the permanent file MINIT,ID=CCT as described earlier.

2. SSSURF - the sculptured surface features is installed.

3. PARSURF - the parametric surface feature is installed.

Local modifications may be installed (no DEFINE needed) by inserting the mods into SOURCDK at the obvious place (look at the listing of SOURCDK at line 170).

Any combination of the above may be installed.

SAMPLE INSTALLATION RUNS

To install APT with parametric surfaces into the system library:

```
INSTAL, { MT1. }
        { HD1. }
ACCOUNT.
LABEL(OLDPL,R,L=APT4V2,D= { HY },
   VSN=PL59A).            { HD }
UPDATE(Q,*=$$$$,D,8)
ROUTE(COMPILE,DC=IN)
7/8/9
$/                       (If using a 9-track tape,
                         insert the following:)
$IDENT      NOSBE4
$DELETE     SOURCDK.7
MAPT02,HD2,T2500.
$/END IF
$IDENT,SELECDK
$DELETE,SOURCDK.8        (To change ACCOUNT cards
ACCOUNT.                 as needed.)
$DEFINE, { MT }
         { NT }
$DEFINE,SYSTEM
$DEFINE,PARSURF
$COMPILE,SOURCDK
6/7/8/9
```

To install APT with sculptured surfaces, parametric surfaces, PSRs and local modifications onto permanent files:

```
INSTAL, { MT1. }
        { HD1. }
ACCOUNT.
LABEL(OLDPL,R,L=APT4V2,D= { HY } ,
   VSN=PL59A)             { HD }
UPDATE(Q,*=$$$$,D,8)
ROUTE(COMPILE,DC=IN)
7/8/9
$/                        (If using 9-track tape,
                          insert the following:)
$IDENT      NOSBE4
$DELETE     SOURCDK.7
MAPT02,HD2,T2500.
$/END IF
$IDENT,SELECDK
$DELETE,SOURCDK.8         (To change ACCOUNT
ACCOUNT.                  cards as needed.)
$DEFINE, { MT }
         { NT }
$DEFINE,SSSURF
$DEFINE,PARSURF
$DEFINE,PSR
$DEFINE,PERMFILE
$INSERT,SOURCDK.170
*INSERT,TCILOUT.4
C MODIFY TCILOUT WITH A COMMENT CARD;
  A SAMPLE LOCAL MOD
*COMPILE,TCILOUT
$COMPILE,SOURCDK
6/7/8/9
```

VERIFICATION PROCEDURE

To obtain a listing of the verification job, run the following procedure:

```
LIST, { MT1. }
      { HD1. }
ACCOUNT.
LABEL(PL59A,R,L=APT4V2,D= { HY } ,
   VSN=PL59A)             { HD }
SKIPF(PL59A,4,17)
COPYBF(PL59A,OLDPL)
UPDATE(F,*=$$$$,L=A1247)
6/7/8/9
```

The DEFINE options in the verification job coincide with the method under which APT was installed (i.e., STANDARD, SYSTEM, USERLIB, PERMFILE). Choose only one of these options when running the verification job.

To run the verification job, assuming APT was installed on permanent files, execute the following procedure:

```
PERMAMP, { MT1. }
         { HD1. }
ACCOUNT.
LABEL(PL59A,R,L=APT4V2,D= { HY } ,
   VSN=PL59A)             { HD }
SKIPF(PL59A,4,17)
COPYBF(PL59A,OLDPL)
UPDATE(Q,*=$$$$,D,8)
ROUTE(COMPILE,DC=IN)
7/8/9
$IDENT,SELECDK
$DELETE,VERIFY.8          (To change ACCOUNT
ACCOUNT.                  cards as needed.)
$DEFINE,PERMFILE
$COMPILE,VERIFY
6/7/8/9
```

The verification job is a sample APT IV part program with neither parametric nor sculptured surfaces. The output from this job should be compared to file 6 of PL59A. File 6 can be printed with a COPYSBF.

RELEASE DESCRIPTION

HARDWARE REQUIREMENTS

A minimum field length of 100K octal is required to load GPSS V 1.2. Once loaded, it requires a minimum field length of 45K octal to execute. Other requirements are the same as the minimum hardware configuration for NOS/BE.

RELEASE MATERIALS

The release materials for GPSS V Version 1.2 are included on the release tape PL35. The release tape contains six files:

File 1 - GPSS V installation job decks GPS1 and GPS2, also the verification job deck VGPS in UPDATE format. The master control character for this UPDATE file is an equal sign (=).

File 2 - GPSS V preloaded absolute binaries.

File 3 - GPSS V relocatable binaries.

File 4 - GPSS V program library in UPDATE format.

File 5 - sample GPSS V model which is used as input by the verification deck VGPS.

File 6 - output from verification job VGPS.

INSTALLATION PROCEDURES

File 1 of PL35 contains the installation decks GPS1, GPS2, and the verification deck VGPS.

GPS1 updates the GPSS V program library (file 4), reassembles GPSS V, and creates a new PL35 in the same format as the original release PL35. GPS1 may be used to configure the original release PL35 for a particular installation and also to install PSR corrective code.

GPS2 installs the preloaded absolute binaries on a permanent file named GPSS,ID=CCT. GPSS V may be executed from this permanent file. The local file name for GPSS V must be GPSS.

VGPS attaches the permanent file GPSS,ID=CCT and executes the model on file 5 of PL35. Output from VGPS should be similar to the output on file 6 of PL35.

GPS1, GPS2, or VGPS may be obtained from PL35 by executing a job of the following form:

```
                       { MT1 }
EXTRACT,CM45000,T100,  { HD1 } .
                       { PE1 }
ACCOUNT.                       (Supply valid ACCOUNT card.)
                                            { HY }
LABEL(PL35,R,L=GPSSV1P2,VSN=PL35,D=         { HD } )
                                            { PE }
UPDATE(Q,P=PL35,*==,C=PUNCH)
7/8/9
=DEFINE X
=/ UPDATE CARDS TO MODIFY INSTALL DECKS GO
   HERE.
=C   Y
6/7/8/9
```

where:

X = HY or HD or PE
Y = the deck name

HY selects 800 bpi 7-track tapes
HD selects 800 cpi 9-track tapes
PE selects 1600 cpi 9-track tapes

If a card punch is not available or if the user desires to execute either GPS1, GPS2, or VGPS directly from PL35, the following three cards should be substituted for the UPDATE card in the previous deck:

```
REQUEST(COMPILE,*Q)
UPDATE(Q,P=PL35,*==)
ROUTE(COMPILE,DC=IN)
```

In order to list GPS1, GPS2, or VGPS as they appear on PL35, a job of the following form should be executed:

```
                    { MT1 }
LIST,CM45000,T100,  { HD1 } .
                    { PE1 }
ACCOUNT.                   (Supply valid ACCOUNT card.)
VSN(TAPE=PL35)                 { HY }
LABEL(TAPE,R,L=GPSSV1P2,D=     { HD } )
                               { PE }
UPDATE(Q,L=A12347,*==,P=TAPE)
7/8/9
=C GPS1
=C GPS2
=C VGPS
6/7/8/9
```

INSTALLATION PARAMETERS

For maximum efficiency, several installation options are provided to configure GPSS V to individual installations. To select an option, an update *DEFINE option name card should be included in the configuration run. Options are:

Option — Definition

CRM
All I/O requests will be processed by the CYBER record manager. If not selected, standard CIO requests will be used. CYBER record manager will be used for I/O on the following files: INPUT, OUTPUT, JOBTA1, JOBTA2, and JOBTA3. All internal file operations on files not available to the HELP block user still use CIO directly for speed purposes.

CP64
Code will be optimized for a 6400 type processor. If not selected, a 6600 type processor is assumed.

XJOP
Monitor requests will be executed via the XJ instruction. If not selected, the XJ instruction is not used.

IMUL
Integer multiply hardware is installed.

FILELOAD
If defined, the preloader will create a random file containing the overlays. This version of GPSS cannot be placed on the system, but must be run from a permanent file. However, the field length required to start a GPSS run is reduced to what is needed to load the (0,0) overlay plus the longest primary overlay, which is about 30K instead of approximately 75 to 105K when FILELOAD is not defined. No ECS load options are available when FILELOAD is defined.

REPRIEVE
If defined, the error termination printout section is executed when GPSS execution is halted due to operator drop, exceeding CP or I/O time limit, or exceeding mass storage limit. Thus, the results of the simulation up to the point at which GPSS execution is halted for one of the above reasons is printed. If REPRIEVE is not defined, GPSS execution is halted without an error termination printout.

FTRACE
If defined, then in addition to the standard one-line trace of GPSS V/6000, the current transaction is also listed. This option slows down a simulation which uses the TRACE feature by a factor of 2 to 10 depending upon the amount of tracing done. It also increases the minimum field length for all runs by 2 to 3 K.

Option — Definition

LISALL
If defined, LIST and UNLIST cards are listed on the output file when encountered. Otherwise, they are never listed.

CMR, XJOP, IMUL, and REPRIEVE are included by default in the install deck GPS1.

Other parameters that may be changed for an installation are the operating system name and installation name. Cards of the following type may be used as input to update to change these parameters:

```
*D,GPSSTXT.419.GPSSTXT.420
 OPSYSTEM MICRO 1,,$NOS/BE 1.X$
 INSTALL  MICRO  1,,$installation name$
```

CONFIGURATION RUN

A job of the following form may be run to generate a new PL35 tape which is configured for a particular installation.

```
                            ⎧ MT1 ⎫
CONFIG,CM45000,T100,        ⎨ HD1 ⎬ .
                            ⎩ PE1 ⎭
ACCOUNT.                (Supply valid ACCOUNT card.)
REQUEST(COMPILE,*Q)
VSN(TAPE=PL35)                                ⎧HY⎫
LABEL(TAPE,R,L=GPSSV1P2,D=                    ⎨HD⎬  )
                                              ⎩PE⎭
UPDATE(Q,*= =,P=TAPE)
ROUTE(COMPILE,DC=IN)
7/8/9
=DEFINE X
=D,GPS1.8
=/ USER                  (Supply valid USER card.)
=I,GPS1.54
*D,GPSSTXT.419,GPSSTXT.420
 OPSYSTEM MICRO 1,,$NOS/BE 1.X$
 INSTALL  MICRO  1,,$installation name$
=C,GPS1
6/7/8/9
```

where:

X = HY or HD or PE

PSR MODIFICATION RUN

A job of the following form may be run to update GPSS V with PSR corrective code and generate a new PL35. The PSRs should be placed on a permanent file called USER, ID=CCT previous to this run.

```
                  ( MT1 )
PSR,CM45000,T100, { HD1 } .
                  ( PE1 )
ACCOUNT.          (Supply valid ACCOUNT card.)
REQUEST(COMPILE,*Q)
VSN(TAPE=PL35)                     (HY)
LABEL(TAPE,R,L=GPSSV1P2,D=         {HD}  )
                                   (PE)

 UPDATE(Q,*= =,P=TAPE)
 ROUTE(COMPILE,DC=IN)
 7/8/9
 =DEFINE X
 =D GPS1.8
 =/ ACCOUNT CARD GOES HERE.
 =I,GPS1.15
 ATTACH(PSRS,USER,ID=CCT)
 =D,GPS1.48,GPS1.55
 *READ PSRS
 =C,GPS1
 6/7/8/9
```

where:

X = HY or HD or PE

NOTES AND CAUTIONS

The deck GPSSTXT on the source OLDPL (file 4) of PL35 contains documentation which is an overview of the system. It is not required but may be helpful for the user to obtain a listing of GPSSTXT by assembling it using COMPASS.

When loading GPSS V, five nonfatal errors are noted. These errors are of the form:

DUPLICATE ENTRY POINT NAME - INIT

Errors of any other form should be investigated.

RELEASE DESCRIPTION

IMSL 6 (International Mathematical and Statistical Library, Version 6) consists of more than 400 FORTRAN functions and subprograms. IMSL 6 is on PL87.

HARDWARE REQUIREMENTS

IMSL 6 requires the minimum hardware configuration for NOS/BE.

NOTES AND CAUTIONS

The installation process does not provide program listings. If a program listing is needed (it is very long), the FTN card in the install deck may be appropriately modified (remove L=0).

RELEASE MATERIALS

IMSL 6 resides on the tape known as PL87. PL87 is a program library in UPDATE format. It has the following characteristics: 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, seven files:

File 1 - install deck in UPDATE OLDPL format,

File 2 - empty file,

File 3 - empty file,

File 4 - program deck in UPDATE OLDPL format,

File 5 - verify job control language deck in UPDATE OLDPL format,

File 6 - expected verify output ((COPYBF,OUTPUT) format), and

File 7 - verify program deck in UPDATE OLDPL format.

INSTALLATION PROCEDURES

The procedures for the installation of IMSL 6 are outlined in the following four steps.

Step 1. Generate Essential Listings

This step includes:

- Installation deck,
- Verification deck, and
- Expected verify output.

NOTE

The user must supply appropriate job and accounting cards.

```
     { MT1 }
JOB, { HD1 } .
     { PE1 }
ACCOUNT.                 (Supply appropriate accounting
                         data.)
                  { MT } { HY }
REQUEST(TAPE,     { NT },{ HD } ,VSN=IMSLIB, E)
                         { PE }
REWIND(TAPE)
UPDATE(P=TAPE, F, D, 8, I=DUMMY, *=/, L=7)
REWIND(TAPE)
SKIPF(TAPE,4,17)
UPDATE(P=TAPE, F, D, 8, *=/, L=7, R, I=DUMMY)
REWIND(TAPE)
SKIPF(TAPE,5,17)
COPYBF(TAPE,OUTPUT,1)
6/7/8/9
```

Step 2. Install IMSL

Installation characteristics are:

- Permanent file name = IMSLIB, and
- Public access rights (ID=PUBLIC).

```
     { MT1 }
JOB, { HD1 } .
     { PE1 }
ACCOUNT.                 (Supply appropriate accounting
                         data.)
                  { MT } { HY }
REQUEST(TAPE,     { NT },{ HD } ,VSN=IMSLIB, E)
                         { PE }
REQUEST(COMPILE,*Q)
UPDATE(P=TAPE, F, D, 8, *=/)
ROUTE(COMPILE,DC=IN)
7/8/9
/IDENT  IMSL             (Optional.)
/DEFINE NOSBE            (Required.)
```

Note that one of the following three tape parameters is required.

```
/DEFINE MT           (7-track, 800 bpi.)
/DEFINE NT           (9-track, 800 cpi.)
/DEFINE NTPE         (9-track, 1600 cpi.)

/DELETE IMSLIL.XX    (Required.)
     ( MT1 )
JOB, { HD1 } ,T1200.  (Supply appropriate job
     ( PE1 )          and account information.)
ACCOUNT.
6/7/8/9
```

Step 3. Run Verification Tests

Refer to the following:

```
     ( MT1 )
JOB, { HD1 } .
     ( PE1 )
ACCOUNT.          (Supply appropriate accounting
                  data.)
                           ( MT1 )
REQUEST(TAPE, { MT } ,     { HD1 } ,VSN=IMSLIB,E)
              { NT }       ( PE1 )
SKIPF(TAPE,4,17)
REQUEST(COMPILE,*Q)
UPDATE(P=TAPE,F,D,8,*=/,R=C)
ROUTE(COMPILE,DC=IN)
7/8/9

/IDENT IMSL          (Optional.)
/DEFINE NOSBE        (Required.)
```

One of the following three tape parameters (required).

```
/DEFINE MT           (7-track, 800 bpi.)
/DEFINE NT           (9-track, 800 bpi.)
/DEFINE NTPE         (9-track, 1600 bpi.)

/DELETE IMSLVL.XX    (Required.)
     ( MT1 )         (*,*,*,*) accounting data.
JOB, { HD1 } ,T300   (Supply appropriate job and
     ( PE1 )         account information.)
ACCOUNT.
6/7/8/9
```

Step 4. Verification

Do a minimal visual comparison of the expected verify output and the actual verify output.

RELEASE DESCRIPTION

LCGT/IGS V2 (Low Cost Graphics Terminal Interactive Graphics System Version 2) runs under NOS/BE and INTERCOM to provide both a remote interactive graphic capability mode and timesharing capability in command mode.

The LCGT/IGS subroutines are callable from user application programs only under FORTRAN Extended, and provide the user with the capability of interacting with his program from the console using the keyboard and cross-hair.

The operation of LCGT/IGS V2 requires the installation of INTERCOM and all other associated products with it under NOS/BE.

HARDWARE CONFIGURATION

The minimum hardware configuration required by NOS/BE and INTERCOM are required to support the LCGT terminal.

Under INTERCOM, the station address for the LCGT terminal operating in synchronous mode must have an odd number address, such as 161. The station number is irrelevant for the terminal operating in asynchronous mode (TTY compatible).

HARDWARE OPTIONS

A CDC synchronous interface (Tektronix product number 021-0135-01) is required on each terminal to communicate with the 6671 multiplexer or the 255X host communications processors in synchronous mode.

RELEASE MATERIALS

LCGT/IGS V2 release consists of a single magnetic tape, PL67, containing six files of data as follows:

File 1 - installation job program library in UPDATE format,

File 2 - empty,

File 3 - empty,

File 4 - program library of LCGT/IGS in UPDATE format,

File 5 - verification job program library in UPDATE format, and

File 6 - empty.

NOTES AND CAUTIONS

All limitations applicable to NOS/BE and INTERCOM also apply to LCGT/IGS V2.

All subroutines in LCGT/IGS V2 are callable from user application programs in FORTRAN Extended only.

The following terminal communications between LCGT and the central site are supported according to the interface modem.

Mode	Modem	Baud Rate
Synchronous	201A 201B	2000 2400
Synchronous	208B	4800
Asynchronous	103A	110 or 300

INSTALLATION PROCEDURES

The installation of LCGT/IGS V2 requires creation of a user library, LCGTVR2, and a graphics class PP routine, GCC. Graphics activity should be stopped during the installation. The job required to install LCGT/IGS is released as an installation deck. INTERCOM is required before attempting to utilize LCGT/IGS V2 but is not necessary for its installation.

The installation job is file 1 on the release tape. This job is a program library in UPDATE format with = (equals) as the master character. A dummy ACCOUNT card is a part of this job. The user should first perform an UPDATE run and list the COMPILE file. As a part of the EXTRACT procedure, the user should modify the JOB/ACCOUNT cards appropriately for his installation.

To obtain the installation job, execute a procedure of the form shown below:

```
                          ( MT1 )
EXTRACT,T10,CM60000,     {  HD1  }.
                          ( PE1 )
ACCOUNT.
LABEL(LCGT,R,L=LCGTIGSV2,
      ( HY )
   D={  HD  } ,VSN=LCGTV2)
      ( PE )
COPYBF(LCGT,OLDPL)
REQUEST(COMPILE,*Q)
UPDATE(F,*==,C,D,8)
UNLOAD(LCGT)
ROUTE(COMPILE,DC=IN)
7/8/9
```

```
=IDENT,XXX
=DEFINE,NOSBE
         ⎧ MT   ⎫   (7-track, 800 bpi.)
=DEFINE, ⎨ NT   ⎬   (9-track, 800 bpi.)
         ⎩ NTPE ⎭   (9-track, 1600 bpi.)
=D,LCGT1,yyy        (To change job and ACCOUNT
ACCOUNT.            cards as needed.)
=COMPILE,LCGT1
6/7/8/9
```

File 5 is the verification job. This job is also in UPDATE format and may require changes in the JOB/ACCOUNT cards of the deck. A preliminary UPDATE run should be made and this deck listed for inspection.

To catalog the binary of the verification job, the following XTRACT procedure can be used:

```
           ⎧ MT1 ⎫
XTRACT,T10,⎨ HD1 ⎬.
           ⎩ PE1 ⎭
ACCOUNT.            (Supply valid ACCOUNT card.)
LABEL(LCGT,R,L=LCGTIGSV2,
     ⎧ HY ⎫
   D=⎨ HD ⎬,VSN=LCGTV2)
     ⎩ PE ⎭
SKIPF(LCGT,4,17)
COPYBF(LCGT,OLDPL)
REQUEST(COMPILE,*Q)
UPDATE(F,*==,C,D,8)
ROUTE(COMPILE,DC=IN)
7/8/9
=IDENT,XX
=DEFINE,NOSBE
=D,LCGT2.yyy        (To change JOB and ACCOUNT
ACCOUNT.            cards as needed.)
=COMPILE,LCGT2
6/7/8/9
```

VERIFICATION PROCEDURE

The verification program, VRFY, is created as a permanent file as described earlier. This program can be run to verify the proper installation of LCGT/IGS as follows:

1. Follow the "Terminal Operating Procedures" to properly set up the terminal.

2. Login with valid username and password.

3. ATTACH (LCGTVR2,ID=LCGT,MR=1)

4. ATTACH(VRFY,ID=LCGT,MR=1)

5. LIBRARY (LCGTVR2)

6. VRFY.

7. You will be asked to enter terminal type (see figure II-6-1):

   1 = 4010 synchronous,
   2 = 4010 asynchronous,
   3 = 4014 synchronous,
   4 = 4014 asynchronous,
   5 = 4014 with EGM, synchronous, and
   6 = 4014 with EGM, asynchronous.

8. The screen will be cleared and figure II-6-2 will be displayed on the screen.

9. Type in any number in the range 5 through 25 followed by a (CR) .

10. Figure II-6-3 resulted by typing in 13.

11. Log out when finished.

TERMINAL OPERATING PROCEDURES

SYNCHRONOUS INTERFACE

With CDC synchronous interface (see "Hardware Option"), LCGT can properly communicate with the following ports under INTERCOM:

- ASCII, synchronous mode at 4800 baud rate.
- ASCII, synchronous mode at 2000-baud rate (6671 only).
- ASCII, asynchronous mode at 110-baud rate.
- ASCII, asynchronous mode at 300-baud rate.

To operate the terminal:

1. Turn the terminal power on by pressing POWER switch to ON position (POWER lamp comes on red).

2. Set ASCII/BCD switch to the ASCII position.

3. Set the baud rate switch to one of the three following positions:

   - EXT position if synchronous 2000-baud rate.
   - 300 position if asynchronous at 300-baud rate.
   - PROG position if asynchronous at 110-baud rate.

4. Set CODE EXPANDER switch as follows:

   - Synchronous mode: turn the switch ON.
   - Asynchronous mode: turn the switch OFF.

5. Following keyboard operation establishes proper operating mode:

   - Synchronous mode: press RESET key. ALPHA cursor should appear at upper left-hand corner of the screen. The terminal is ready for dialing up.

- Asynchronous mode: first press the RESET key, and then press the SHIFT, CTRL and P keys simultaneously. The terminal is ready for asynchronous mode.

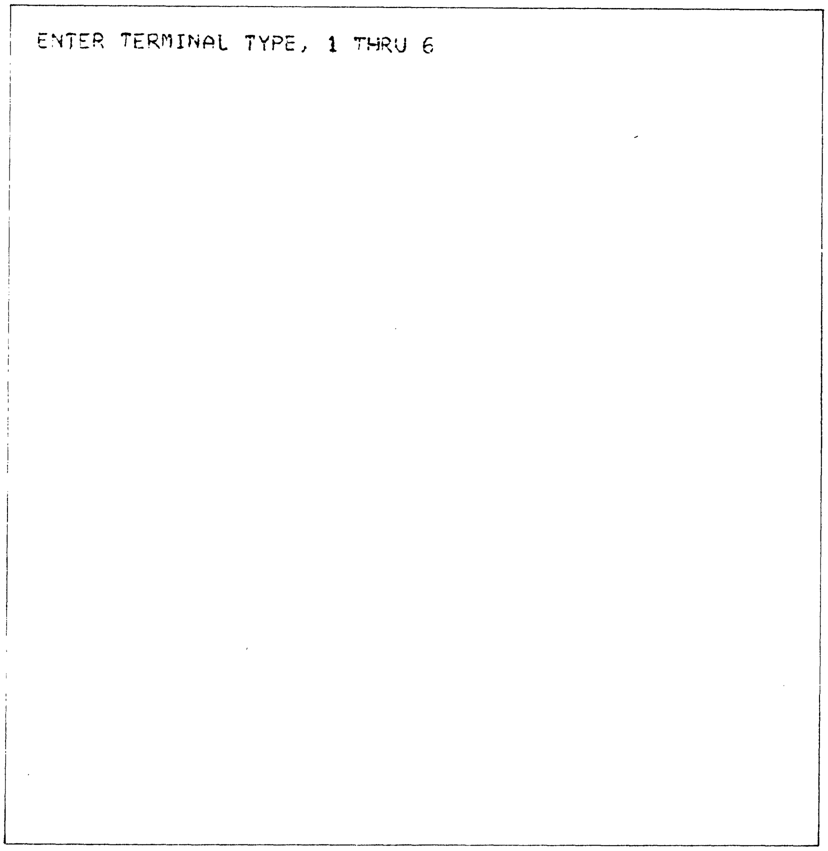
6. If the terminal is in synchronous mode, ON-LINE indicator lamp starts blinking when the proper communication is established. The indicator will not operate in asynchronous mode.

INTERCOM will send the banner page and LOGIN message when it is ready for interactive command processing.

STANDARD INTERFACE

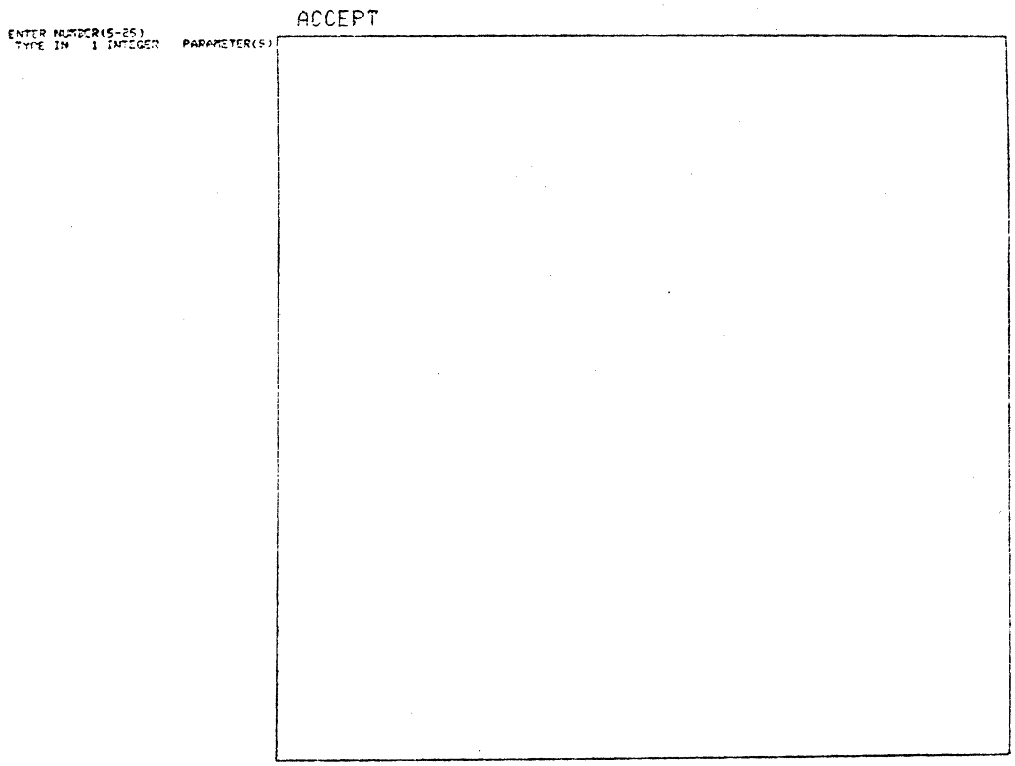
With standard interface, and without the special CDC Synchronous interface, the terminal can only be operated in asynchronous mode.

The Asynchronous mode operating procedures are the same as the one described above, (except for step 5). The terminal is always in asynchronous mode, and, therefore, no special keyboard operations are required to establish the mode switching.



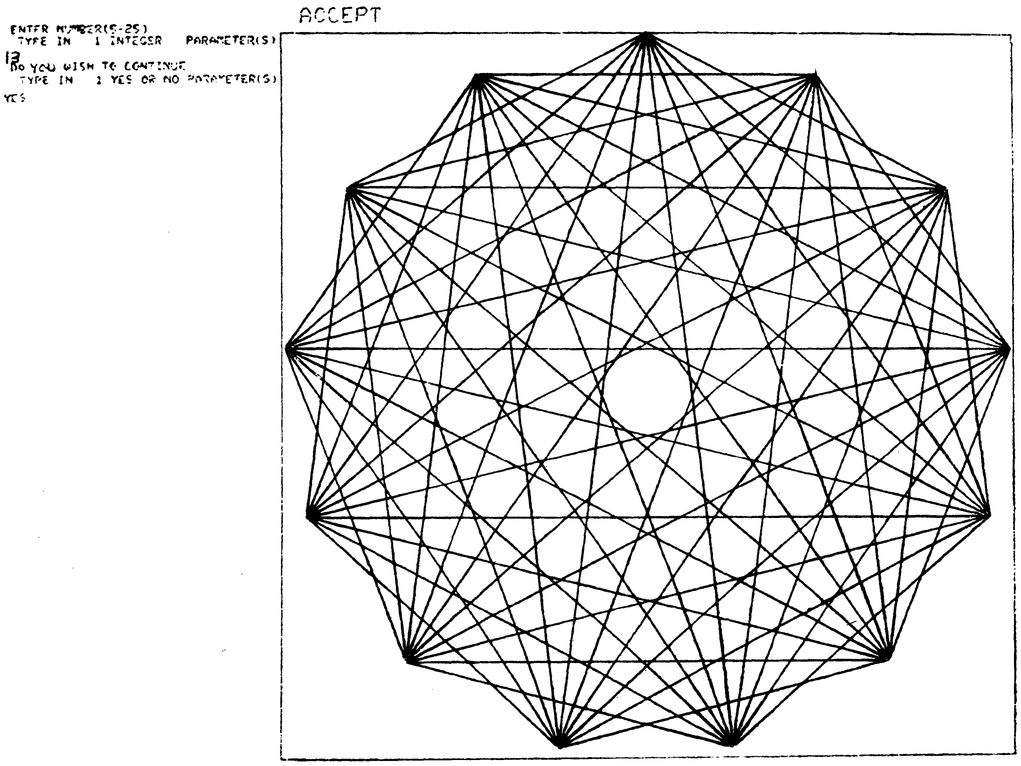
```
ENTER TERMINAL TYPE, 1 THRU 6
```

Figure II-6-1. The User Is Asked to Enter Terminal Type



```
ENTER NUMBER(5-25)
 TYPE IN    1 INTEGER    PARAMETER(S)
```

Figure II-6-2. Terminal User Is Asked to Select Number



```
ENTER NUMBER(5-25)
 TYPE IN    1 INTEGER    PARAMETER(S)
13
DO YOU WISH TO CONTINUE
 TYPE IN    1 YES OR NO PARAMETER(S)
YES
```

Figure II-6-3. Result of Typing in a 13

RELEASE DESCRIPTION

HARDWARE CONFIGURATION

PERT/TIME requires a minimum hardware configuration with at least 65K (octal) central memory, three tape units, and the other minimum equipment required by NOS/BE.

RELEASE MATERIALS

PERT/TIME is on release tape PL85. PL85 contains PERT2P0 as file ID in HDR1 and six files:

File 1 - installation deck program library in UPDATE format,

File 2 - PERT/TIME in absolute binary format,

File 3 - PERT/TIME in relocatable binary format,

File 4 - source PL in UPDATE format,

File 5 - empty, and

File 6 - empty.

LIMITATIONS

Because the file names TAPE1 through TAPE7 are used internally, no user file name may be TAPE 1 through 7.

INSTALLATION PROCEDURES

Release tape PL85 contains PERT/TIME. To get a listing and a punched-card copy of the installation decks from the first file of PL85, run a job similar to the following (if a card punch is not on-line, the decks must be keypunched by hand):

```
                 ( MT1 )
JOB statement,  {  HD1  }
                 ( PE1 )
ACCOUNT information
VSN(TAPE=PL85)                      ( HY )
LABEL(TAPE,R,L=PERT2P0,D=  {  HD  }   )
                                    ( PE )
UPDATE(F,L=A12347,P=TAPE)
REWIND(COMPILE)
COPYBF(COMPILE,PUNCH)
7/8/9
6/7/8/9
```

The following job installs PERT/TIME executable binary on a permanent file:

```
                 ( MT1 )
JOB statement,  {  HD1  }
                 ( PE1 )
ACCOUNT information
VSN(TAPE=PL85)                ( HY )
LABEL(TAPE,R,L=PERT2P0,  {  HD  }   )
                              ( PE )
REQUEST(PERT78,*PF)
SKIPF(TAPE,1,17)
COPYBF(TAPE,PERT78)
CATALOG(PERT78,ID=PERT78)
6/7/8/9
```

To execute from the release tape PL85, run a job similar to the following:

```
              ( MT1 )
PERT,T1000,  {  HD1  }.
              ( PE1 )
ACCOUNT information
VSN(TAPE=PL85)                ( HY )
LABEL(TAPE,R,L=PERT2P0,  {  HD  }.
                              ( PE )
SKIPF(TAPE,1,17)
COPYBF(TAPE,PERT78)
PERT78.
7/8/9    )
          }  PERT networks
6/7/8/9  )
```

If a master data tape is made, insert the following after the LABEL statement:

```
REQUEST TAPE6,MT.
```

If an old master data tape is input and a new master data tape saved, insert the following after the LABEL statement.

```
REQUEST TAPE4,MT.     OLD MASTER
REQUEST TAPE6,MT.     NEW MASTER
```

If either option is used the MT1 job card parameter must be increased. An entry must be placed on the Y-card for each additional tape used.

Successive PERT/TIME networks may be processed by batching as follows: (A higher FL will be required.)

```
K
L
M
X
Y
W
A      PERT BATCHED NETWORKS
●
●
●
A
Z
K
L
●
●
●
Z
```

(Input for each network starts with the K control card and ends with the Z control card.)

The following job compiles and installs PERT/TIME from the source PL to a permanent file.

```
                ⎧ MT1 ⎫
JOB statement,  ⎨ HD1 ⎬ .
                ⎩ PE1 ⎭
ACCOUNT information
VSN(TAPE=PL85)              ⎧ HY ⎫
LABEL(TAPE,L=PERT2P0,       ⎨ HD ⎬ )
                            ⎩ PE ⎭
SKIPF(TAPE,3,17)
UPDATE(F,P=TAPE,R)
UNLOAD(TAPE)
REWIND(COMPILE)
REQUEST(PERT78,*PF)
FTN(I=COMPILE,B=PERT,L=0)
LOAD(PERT)
NOGO(PERT78)
CATALOG(PERT78,ID=PERT78)
7/8/9
6/7/8/9
```

PRODUCT DESCRIPTION 777/IGS (HOST AND REMOTE), VERSION 2.2

777/IGS (HOST and REMOTE), Version 2.2, in conjunction with INTERCOM 4 and the NOS/BE operating system, provides a remote interactive graphics capability with time-shared access to Control Data CYBER 170L, CDC CYBER 70L, and 6000 series computers from the 777 CYBER graphics terminal. Remote batch jobs may be submitted from 777 terminals equipped with a remote card reader and line printer.

The 777/IGS host library provides the interface between the user's application program written in FORTRAN Extended and the operator of the 777 console. The console operator interacts with the application through the keyboard, function keys, and light pen.

The 777/IGS controller software allows the console operator all INTERCOM commands and capabilities available to nongraphic CRT terminals and the use of remote batch capabilities as provided by 1700 IMPORT and controlled by the TTY. If the controller software is defined as voice grade communication, remote batch capabilities are compatible to a 200 UT via the 777 console.

PRODUCT DESCRIPTION - 777/IGS WITH 3D (HOST AND REMOTE), VERSION 2.2

777/IGS with 3D (HOST and REMOTE), Version 2.2, is a logical extension of 777/IGS (HOST and REMOTE), Version 2.2. 777/IGS with 3D provides the capabilities of 777/IGS and a three-dimensional (3D) graphics capability.

Prior to Version 2.2, the 777/IGS 3D products consisted of 3D routines added to the 777/IGS routines during installation to form a graphics system with 3D capability. Beginning with Version 2.2, 777/IGS with 3D contains all the routines of a complete graphics system with 3D capability. This change in release format allows a simpler installation procedure using two release tapes rather than four.

777/IGS (HOST)

RELEASE DESCRIPTION

777/IGS (HOST) runs under the NOS/BE operating system. Accordingly, it requires the same minimum hardware configuration as NOS/BE. 777/IGS (HOST) is designed to be used from permanent files.

Release Materials

777/IGS (HOST) resides on the tape known as PL40. This release tape contains six files:

File 1 - program library of installation job.

File 2 - program library of demo programs,

File 3 - program library of the CDC CYBER graphics terminal performance program (CGTPP),

File 4 - program library of 777/IGS (HOST),

File 5 - empty, and

File 6 - empty.

The CGTPP is provided by the Systems Division of Vought Corporation and is included with these release materials at the request of VIM, Inc.

Notes and Cautions

The installation of 777/IGS requires installing from PL40 (HOST) and PL41 (REMOTE). The installation jobs are on the first file of the tapes. The installation of INTERCOM 4 is required before attempting to use 777/IGS but is not necessary for proper installation. 777/IGS runs on a Control Data CYBER graphics terminal, model 777-1, 777-2, or 777-3. Graphics activity should be stopped during installation of new 777/IGS host software.

INSTALLATION PROCEDURE

File 1 on PL40 contains the deck INSTALL which is used to install the host software system (777/IGS library, debug library, and PP program GCC), the binaries for the demo programs, and the CGTPP.

The user performs an UPDATE run to list the install job deck and locate the ACCOUNT and JOB cards in the deck. As part of the EXTRACT procedure, the ACCOUNT and/or JOB cards should be modified appropriately for the user's installation. To list the contents of the install job deck, run the following job:

```
      ⎧ MT1 ⎫
LIST, ⎨ HD1 ⎬.
      ⎩ PE1 ⎭
ACCOUNT.
                          ⎧ HY ⎫
LABEL,OLDPL,R,L=777IGSH,D=⎨ HD ⎬,VSN=PL40.
                          ⎩ PE ⎭
UPDATE,F,*=+,L=A7.
UNLOAD,OLDPL.
7/8/9
6/7/8/9
```

Braces,{ }, in this section, indicate alternate parameters for installation from 7-track release tapes (top alternate) or from 9-track release tapes (bottom two alternates).

Execution of the following procedure will install the host software:

```
          ( MT1 )
EXTRACT,  { HD1 }.
          ( PE1 )
ACCOUNT.
                                  ( HY )
LABEL,OLDPL,R,L=777IGSH,D=        { HD } ,VSN=PL40.
                                  ( PE )
REQUEST,COMPILE,*Q.
UPDATE,F,*=+,D,8.
ROUTE,COMPILE,DC=IN.
7/8/9
+IDENT BUILD
         ( 7-TRACK )  ( 7 TRACK,  800 bpi )
+DEFINE  { 9-TRACK }  { 9 TRACK,  800 cpi }
         ( 9-TRPE  )  ( 9 TRACK, 1600 cpi )
+DEFINE parameters as described below
+DELETE INSTALL.6
ACCOUNT.              }  To change ACCOUNT and/
other modifications   }  or JOB cards as necessary.
6/7/8/9
```

The second +DEFINE card defines parameters by which optional functions of the install job are requested. These parameters are as follows:

Parameter	Definition
IGSDBUG	The 777/IGS debug library is cataloged (IGSDBUG,ID=777).
CATDEMOS	Demo program relocatable binary files are cataloged (DEMO1,ID=777 through DEMO4,ID=777).
VOUGHT	Vought Corporation's Control Data CYBER graphics terminal performance program absolute binary file is cataloged (CGTPP,ID=777).
EDITLIBPP	The 777/IGS PP program GCC is installed via a system EDITLIB. EDITLIBPP must be defined for correct and complete installation of 777/IGS (HOST) unless there have been no modifications to GCC since the last time a version of 777/IGS was installed.

Regardless of the defined option parameters, the 777/IGS library will be cataloged (IGS777,ID=777).

777/IGS (REMOTE)

RELEASE DESCRIPTION

The 777 controller software allows a console operator the use of all INTERCOM commands and capabilities to non-graphic CRT terminals. Use of remote batch capabilities as provided by 1700 IMPORT and controlled by the TTY are also available. If the controller software is defined as voice grade, remote batch capabilities are compatible to a 200 UT via the 777 console.

Release Materials

777/IGS (REMOTE) resides on the tape known as PL41. This release tape contains six files:

File 1 - program library of installation job,

File 2 - absolute binary of 8-bit intermediate bootstrap (405/430 card reader bootstrap),

File 3 - absolute binaries of installation utility programs
  record 1: assembler, and
  record 2: binary generation routine;

File 4 - program library of 777/IGS (REMOTE),

File 5 - empty, and

File 6 - empty.

Notes and Cautions

The installation of 777/IGS requires installing from PL40 (HOST) and PL41 (REMOTE). The installation jobs are on the first file of the tapes. The installation of INTERCOM 4 is required before attempting to use 777/IGS but is not necessary for proper installation.

777/IGS runs on a Control Data CYBER graphics terminal, model 777-1, 777-2, or 777-3.

A 415 card punch is needed to install 777/IGS REMOTE software.

At sites with a 1729-3 card reader, do not mark the backs of any cards. This type of reflective reader may interpret the marks as punched holes.

There is a configuration change (for MSOS 4 compatibility) with Version 2.1 of 777/IGS and an addition of several new peripheral options with Version 2.2. Users currently on 777/IGS Version 2.0 will have to change the controller interrupts before going to a later system. The changes and additions are shown in Tables II-8-1 and II-8-3. See also the <u>1743-1 Reference Manual</u>.

TABLE II-8-1. INTERRUPT/EQUIPMENT HISTORY

	V2.0	V2.1	V2.2
HW Error Processor	0/-	0/-	0/-
TTY	1/1	1/1	1/1
Zeta Plotter	-	-	2/2
Line Printer	5/15	4/4	4/4
Voice Grade Communications	-	5/5	5/5
405 CR	6/6	6/6	6/6
1732-3 Magnetic Tape	-	-	7/7
777 DCI Console 1	10/4	8/8	8/8
Wideband Communications	3/5	9/9	9/9
1743-2 (Hardcopy)	-	-	10/10
430 CR	9/9	11/11	11/11
777 DCI Console 2	11/12	12/12	12/12
777 DCI Console 3	12/2	14/14	14/14

TABLE II-8-2. FJ606 (1743) STRAPPING OPTIONS

Sync Bits	→ Bits 4, 2, and 1 (16_F)
Parity Mode	→ No Parity
No Data Bits	→ 8 Bits
Equip. Code	→ (^{5}16) Bits 7 & 9 & 6
Protect	→ No Protect
Int. Cycle Time	→ No Plugs In
Operating Mode	→ Full Duplex, Both Data Sets

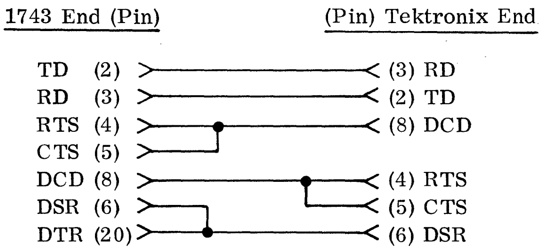
LOCAL HARDCOPY

Local hardcopy hardware requirements are:

1743-2:
- Set on equipment 10 interrupt line 10.
- Baud rate set at 300-9600 (same as Tektronix).
- Half duplex.
- 8 bits, ASCII data.
- No parity.
- Port 0.

Tektronix 4010 (or equivalent):
- Full duplex.
- Baud rate set at 300-9600 (same as 1743-2).

Required cable modifications to run without modems on the 1743-2 and 4010 are as follows:



HOST HARDCOPY

A Houston Instrument plotter and associated controller can be directly attached to the 1742-X line printer. This requires the use of the Houston Instrument UNIPLOT post-processor.

INSTALLATION PROCEDURE

File 1 on PL41 contains the deck INSTALL. It builds and punches out the controller software deck (with appropriate intermediate bootstrap).

First, the user performs an UPDATE run to list the install job deck and locate the ACCOUNT and JOB cards in the deck. As part of the EXTRACT procedure, the ACCOUNT and/or JOB cards should be modified appropriately for the user's installation. To list the contents of the install job deck, run the following job:

```
      ⎧ MT1 ⎫
LIST, ⎨ HD1 ⎬ .
      ⎩ PE1 ⎭
ACCOUNT.
                                ⎧ HY ⎫
LABEL,OLDPL,R,L=777IGSR,D=      ⎨ HD ⎬ ,VSN=PL41.
                                ⎩ PE ⎭
UPDATE,F,*=+,L=A7.
UNLOAD,OLDPL.
7/8/9
6/7/8/9
```

Braces, { }, used in this section, indicate alternate parameters for installation from 7-track release tapes (top alternate) or from 9-track release tapes (bottom two alternates).

Execution of the following procedure will provide a card deck on the 415 card punch unit. This deck consists of an 8-bit special loader (intermediate bootstrap) followed by the controller software. There is one and only one 8-bit format loader. It will execute correctly for the 430, 1729-3 or 1726/405 card readers. This deck is used as input for the controller initialization procedure as described in the 777/IGS Remote Job Entry User Guide.

```
         ⎧ MT1 ⎫
EXTRACT, ⎨ HD1 ⎬ .
         ⎩ PE1 ⎭
ACCOUNT.
                                ⎧ HY ⎫
LABEL,OLDPL,R,L=777IGSR,D=      ⎨ HD ⎬ ,VSN=PL41.
                                ⎩ PE ⎭
REQUEST,COMPILE,*Q.
UPDATE,F,*=+,D,8.
ROUTE,COMPILE,DC=IN.
7/8/9
```

```
+IDENT BUILD
         ⎧7-TRACK⎫   ⎧7 TRACK,  800 bpi ⎫
+DEFINE  ⎨9-TRACK⎬   ⎨9 TRACK,  800 cpi ⎬
         ⎩9-TRPE ⎭   ⎩9 TRACK, 1600 cpi ⎭
+DEFINE parameters as described below
+DELETE INSTALL.6  ⎫  To change ACCOUNT
ACCOUNT.           ⎬  and/or job cards
other modifications⎭  as necessary
6/7/8/9
```

The second +DEFINE card defines parameters by which optional functions of the install job are requested and certain characteristics of the controller software configuration are specified. These parameters are as shown in figure II-8-1.

CONTROLLER INITIALIZATION PROCEDURE

The operator loads the card deck produced by the 777/IGS (REMOTE) installation procedure into the 777 CYBER graphics terminal controller from the remote card reader to initialize the controller. For a detailed description of this process, please refer to the 777 IGS Remote Job Entry User Guide.

VERIFICATION PROCEDURES— 777/IGS (HOST AND REMOTE)

In order to verify that the 777/IGS system has been properly installed, the four demo programs may be executed. It is assumed that CATDEMOS was defined in the host EXTRACT job so that the demo program binaries have been cataloged as DEMO1,ID=777 through DEMO4,ID=777. For general information on terminal operations, please refer to the 777 IGS Reference Manual.

	Parameter		Definition
1.	May define 1 of:	CON1, CON2, or CON3	Configure 777 system with one, two, or three consoles, respectively. Default is CON1.
2.	May define 1 of:	24K, 28K, 32K, 36K, 40K, 44K, 48K, 52K, 56K, 60K, or 65K	Configure 777 system with controller memory size indicated. Default is 28K.
3.	May define 1 of:	CR405 or CR430	Configure system with 1726/405 card reader or 1728/430 card read-punch, respectively. (1729-2,3 card readers need CR430). Default is no card reader.
4.	May define 1 of:	026 or 029	Set keypunch code for all remote batch to 026 or 029, respectively. Default is 026.
5.	May define:	LP501	Configure 777 system with 1740/501 or 1742 line printer.
6.	May define:	LP501 and LP1742	Configure 777 system with 1742-30 line printer.
7.	May define 1 of:	DMP421 or DMP42X	Controller software will include a dump routine. It will output to a line printer configured into the system via defining LP501 or LP501 and LP1742, respectively. Default is no dump routine.
8.	May define 1 of:	IMPORT or CM1743	Configure 777 system to use wideband or voice-grade communications, respectively, and to have batch processing capabilities. If neither IMPORT nor CM1743 is defined, the 777 system uses wideband communications and has no batch processing capabilities. (Wideband communication with the host computer uses high-speed INTERCOM.)
9.	May define:	TTY	Configure 777 system with a TTY.
10.	May define:	UT	Configure 777 system with a utility processor as described in the 777 IGS Remote Job Entry User Guide.
11.	May define:	MT	Configure 777 system with a 1732-3/616-92,95 9-track magnetic tape unit.
12.	May define:	PL	Configure 777 system with a Zeta 1200-series plotter.
13.	May define:	HC	Configure 777 system for Tektronix 4631 hardcopy capability via a Tektronix display and a 1743-2 communications controller.

Figure II-8-1. Second +DEFINE Card Parameters

The operator should log in and enter the following commands to INTERCOM:

```
ATTACH,IGS777,ID=777,MR=1
ATTACH,DEMO1,ID=777,MR=1
ATTACH,DEMO2,ID=777,MR=1
ATTACH,DEMO3,ID=777,MR=1
ATTACH,DEMO4,ID=777,MR=1
LIBRARY,IGS777
```

At this point, entering DEMO1, DEMO2, DEMO3, DEMO4 will cause a demo program to be executed. Brief descriptions of the demo programs are as follows:

DEMO1 After the operator enters GO in response to a PAUSE, the so-called quick-look pattern graphics sampler appears on the console screen along with another PAUSE. When the operator again enters GO, the program terminates.

DEMO2 A display of numbers appears on the console screen along with a prompting message which requests that the operator choose a number with the lightpen. Then, a doily pattern appears. The complexity of the pattern depends directly on the magnitude of the selected number. At this point, the operator may select the ENCORE lightbutton returning the initial display for another choice or select the FIN lightbutton terminating the program.

DEMO3 A lightbutton menu of interactive functions appears on the screen. The operator uses this menu, together with the lightpen and keyboard, to interact with the graphics display file. The interactive functions are tracking, windowing, zooming, copying, moving, deletion, freehand, rubber-band drawing, and hardcopy. When the operator enters GO, the program terminates.

DEMO4 180 simple pictures appear and disappear in quick succession followed by an eight-leafed rose pattern and a PAUSE. Press the INT key twice so that the pattern is on the screen, but the input rectangle is not. Then, space-bar presses start and stop an intriguing simulated film loop. To terminate, press the INT key to regain the input rectangle and then enter a GO.

777/IGS WITH 3D (HOST)

RELEASE DESCRIPTION

777/IGS with 3D (HOST) runs under the NOS/BE operating system. Accordingly, it requires the same minimum hardware configuration as NOS/BE. 777/IGS with 3D (HOST) is designed to be used from permanent files.

Release Materials

777/IGS with 3D (HOST) resides on the tape known as PL46. This release tape contains six files:

File 1 - program library of installation job,

File 2 - program library of demo programs,

File 3 - program library of the Control Data CYBER graphics terminal performance program (CGTPP),

File 4 - program library of 777/IGS with 3D (HOST),

File 5 - empty, and

File 6 - empty.

The CGTPP is provided by the Systems Division of Vought Corporation and is included with these release materials at the request of VIM, Inc.

Notes and Cautions

The installation of 777/IGS with 3D requires installing from PL46 (HOST) and PL47 (REMOTE). The installation jobs are on the first file of the tapes. The installation of INTERCOM 4 is required before attempting to use 777/IGS with 3D but is not necessary for proper installation.

777/IGS with 3D runs on a Control Data CYBER graphics terminal, model 777-1, 777-2, or 777-3.

Graphics activity should be stopped during installation of new 777/IGS HOST software.

INSTALLATION PROCEDURE

Installation of 777/IGS with 3D (HOST) from PL46 is exactly the same as the installation of 777/IGS (HOST) from PL40 with three exceptions.

1. The LABEL control card in the LIST and EXTRACT jobs is as follows:

```
                                 ⎧HY⎫
LABEL,OLDPL,R,L=777IGSH3D,D=     ⎨HD⎬ ,
    VSN=PL46.                    ⎩PE⎭
```

2. An extra card should be entered into the EXTRACT job deck immediately after the IDENT card. The new extra card is as follows:

```
+DEFINE (2D+3D)
```

3. The CATDEMOS option in the EXTRACT job will catalog a fifth demo program (DEMO5,ID=777) which may be used to verify correct installation of the 3D system. During execution of the program, selection of the word ROTATE with the lightpen causes a box-shaped 3D figure to rotate. Pressing any function key causes program termination to occur.

777/IGS WITH 3D (REMOTE)

RELEASE DESCRIPTION

The 777 controller software allows a console operator the use of all INTERCOM commands and capabilities available to nongraphic CRT terminals. The use of remote batch capabilities as provided by 1700 IMPORT and controlled by the TTY are also available. If the controller software is defined as voice grade, remote batch capabilities are compatible to a 200 UT via the 777 console.

Release Materials

777/IGS with 3D (REMOTE) resides on the tape known as PL47. This release tape contains six files:

File 1 - program library of installation job,

File 2 - absolute binary of 8-bit intermediate bootstrap (405/430 card reader bootstrap),

File 3 - absolute binaries of installation utility programs
  record 1: assembler, and
  record 2: binary generation program;

File 4 - program library of 777/IGS with 3D (REMOTE),

File 5 - empty, and

File 6 - empty.

Notes and Cautions

The installation of 777/IGS with 3D requires installing from PL46 (HOST) and PL47 (REMOTE). The installation jobs are on the first file of the tapes. The installation of INTERCOM 4 is required before attempting to use 777/IGS with 3D but is not necessary for proper installation.

777/IGS with 3D runs on a Control Data CYBER graphics terminal, model 777-1, 777-2, or 777-3.

A 415 card punch is needed to install 777/IGS REMOTE software.

At sites with a 1729-3 card reader, do not mark the backs of any cards. This type of reflective reader may interpret the marks as punched holes.

The configuration change described in the notes and cautions under 777/IGS (REMOTE) also applies to 777/IGS with 3D (REMOTE).

INSTALLATION PROCEDURE

Installation of 777/IGS with 3D (REMOTE) from PL47 is exactly the same as the installation of 777/IGS (REMOTE) from PL41 with two exceptions.

1. The LABEL control card in the LIST and EXTRACT jobs should be as follows:

```
                               ⎧HY⎫
LABEL,OLDPL,R,L=777IGSR3D,D=   ⎨HD⎬  ,
   VSN=PL47.                   ⎩PE⎭
```

2. An extra card should be entered into the EXTRACT job deck immediately after the IDENT card. The new extra card should be as follows:

```
+DEFINE (2D+3D)
```

RELEASE DESCRIPTION

SIMSCRIPT I.5 Version 3 operates under NOS/BE 1 on the same minimum configuration as NOS/BE.

RELEASE MATERIALS

The release materials for SIMSCRIPT I.5 Version 3 are included on program library tape PL27. The release tape contains six files:

File 1 - SIMSCRIPT installation job decks SIMI51 and SIMI52 plus verification job deck VSIMI5,

File 2 - SIMI5 compiler (in EDITLIB binary format),

File 3 - SIMI5 execution library,

File 4 - program library containing compiler and execution library in UPDATE 1.2 format (a detailed description of the file appears below),

File 5 - blank file, and

File 6 - output from verification job VSIMI5.

LIMITATIONS

SIMSCRIPT I.5 Version 3 operates under NOS/BE 1 in conjunction with COMPASS 3 and FORTRAN Extended 4.

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURES

SIMSCRIPT Version 3 is made up of a compiler and an execution library. They are contained in a single UPDATE record to facilitate generation of an overlay tape.

The first 73 decks (ADEF through ASML) belong to the SIMSCRIPT Version 3 compiler and are written in SIMSCRIPT.

ADEF contains the definition deck necessary to compile the compiler. It always must be written on the COMPILE file when any compiler routine written in SIMSCRIPT is to be recompiled.

MAINAD through WASM (15 decks) represent the definition processor part of the compiler (overlay 1.0).

DEFINE through REFL (52 decks) represent the translator part of the compiler (overlay 2.0).

ASMBL through ASML (5 decks) represent the assembler phase of the compiler (overlay 3.0).

The TREND$ deck marks the end of the SIMSCRIPT language routines; it contains a *WEOR.

The following 38 decks (SIMI5 through ALLOUT) belong to the SIMI5 compiler and are written in COMPASS.

SIMI5 through XXERROR represent the root segment of the compiler (overlay 0.0).

XX$DEFS through CONUM are part of the definition processor overlay (overlay 1.0).

XX$TRAN through PACKIT are part of the translator overlay (overlay 2.0).

XX$FAKE represents overlay 2.1.

XX$ASMB through ALLOUT are part of the assembler overlay (overlay 3.0).

The UTEND$ deck marks the end of the compiler routines written in COMPASS; it contains a *WEOR.

The 26 decks from XX$MAIN through XX$FMT represent the SIMI5 execution library and are written in COMPASS.

Loader control cards are included in the following decks: SIMI5, XX$DEFS, XX$TRAN, XX$FAKE, and XX$ASMB.

The SIMI5 compiler automatically generates the following LDSET table to reference the necessary libraries:

SIMLIB	which should contain the SIMI5 execution library
FORTRAN	
BAMLIB	

SYSTEM GENERATION

SIMSCRIPT installation job decks SIMI51 and SIMI52, plus verification job deck VSIMI5, may be obtained from the first file of PL27 by executing a job of the following form:

```
                           ⎧MT1.⎫
    EXTRACT,CM45000,T100,  ⎨HD1.⎬
                           ⎩PE1.⎭
    ACCOUNT CARD.                   ⎧HY⎫
    LABEL(PL27,R,L=SIMI53P0,D=      ⎨HD⎬  ,VSN=PL27)
                                    ⎩PE⎭
    UPDATE(Q,P=PL27,*==,C=PUNCH)
    7/8/9
    =DEFINE X
    =C  Y
    6/7/8/9
```

where X=HY, HD or PE and Y is the deck name.

The SIMI51 deck includes the control cards necessary to generate the SIMSCRIPT Version 3 compiler and execution library from the release tape. It also generates a new program library tape. If PSRs are to be applied against SIMSCRIPT they should be placed on a permanent file called MINIT, ID=CCT previous to execution of SIMI51.

Job SIMI52 can be used to enter SIMSCRIPT Version 3 into the running system through EDITLIB either from the release PL27 or from the tape created by SIMI51. Job DST3 may then be run to secure a deadstart tape including SIMSCRIPT Version 3.

VERIFICATION PROGRAM

The verification deck provided with the release validates SIMSCRIPT. The validation consists of a SIMSCRIPT job with a report. The time required to run the validation job is about one minute. The output includes the actual validation (magic squares) and dayfile as follows:

```
THIS IS A 3 BY 3 MAGIC SQUARE.  ALL OF THE ROWS,
COLUMNS AND DIAGONALS SUM TO 15:

4    3    8
9    5    1
2    7    6

THIS IS A 5 BY 5 MAGIC SQUARE.  ALL OF THE ROWS,
COLUMNS AND DIAGONALS SUM TO 65:

11   10    4   23   17
18   12    6    5   24
25   19   13    7    1
 2   21   20   14    8
 9    3   22   16   15
```

```
04.13.36.VSIMS1H
04.13.36.VSIMS,CM65000,T1000.
04.13.36. THIS SIMPLE PROGRAM VERIFIES CORRECT
04.13.36. INSTALLATION OF
04.13.36.  SIMSCRIPT
04.13.36.SIMI5.
04.13.42.LGO.
04.13.47.STOP
04.13.47.CP      004.403 SEC.
04.13.47.PP      002.815 SEC.
```

HY selects 800 bpi 7-track tapes.
HD selects 800 bpi 9-track tapes.
PE selects 1600 bpi 9-track tapes.

If a card punch is not available, or if the user desires to execute either SIMI51, SIMI52, or VSIMI5 directly from PL27, a job of the following form should be executed.

```
                         ⎧MT1⎫
    EXTRACT,CM45000,     ⎨HD1⎬ .
                         ⎩PE1⎭
    ACCOUNT CARD.                   ⎧HY⎫
    LABEL(PL27,R,L=SIMI53P0,D=      ⎨HD⎬  ,VSN=PL27)
                                    ⎩PE⎭
    REQUEST(COMPILE,*Q)
    UPDATE(Q,P=PL27,*==)
    REWIND(COMPILE)
    ROUTE(COMPILE,DC=IN)
    7/8/9
    =DEFINE  X
    =C  Y
    6/7/8/9
```

where X=HY, HD or PE and Y=deck name.

In order to list SIMI51, SIMI52, or VSIMI5 as they appear on PL27 a job of the following form should be executed.

```
                     ⎧MT1⎫
    LIST,CM45000,    ⎨HD1⎬
                     ⎩PE1⎭
    ACCOUNT CARD.                   ⎧HY⎫
    LABEL(PL27,R,L=SIMI53P0,D=      ⎨HD⎬  ,VSN=PL27)
                                    ⎩PE⎭
    UPDATE(Q,L=A12347)
    7/8/9
    =C SIMI51
    =C SIMI52
    =C VSIMI5
    6/7/8/9
```

TIGS 1 OVERVIEW

RELEASE DESCRIPTION

TIGS 1 (Terminal Independent Graphics System Version 1) runs under NOS/BE and INTERCOM (Version 4 or Version 5) to provide the user with both a remote interactive graphic capability in graphic mode and a timesharing capability in interactive mode.

TIGS 1 consists of a preprocessor and a post processor. The preprocessor writes a neutral display file that is then traversed by the post processor to produce graphics output to a specific terminal.

The TIGS subroutines are callable from user application programs that are written in FORTRAN Extended.

The operation of TIGS 1 under NOS/BE requires the installation of INTERCOM and all associated products.

HARDWARE CONFIGURATION

The minimum hardware configuration required by NOS/BE and INTERCOM is required to support TIGS 1.

Under INTERCOM, the site address for a terminal operating in synchronous mode must be 160_8 for BCD and 161_8 for ASCII. The site address is irrelevant for the terminal operating in asynchronous mode.

RELEASE MATERIALS

The TIGS 1 release requires both the preprocessor magnetic tape, PL86A, and an appropriate post processor tape. Refer to the individual post processor sections for installation, verification, and terminal operation of TIGS 1.

TIGS 1 PREPROCESSOR

The TIGS 1 preprocessor resides on the tape known as PL86A. PL86A has the following characteristics: 7-track (800-bits per inch (bpi)) or 9-track (800 or 1600-characters per inch (cpi)), binary recording mode, TIGSPREV1 as the file ID in the HDR1 label, and six files:

File 1 - empty,
File 2 - empty,
File 3 - empty,
File 4 - TIGS 1 preprocessor source code program library in UPDATE format,
File 5 - empty, and
File 6 - empty.

TEKTRONIX 401X POST PROCESSOR OPTION

RELEASE MATERIALS

The Tektronix 401X post processor runs under NOS when installed with the TIGS 1 preprocessor.

The Tektronix 401X post processor resides on the tape known as PL86B. PL86B is a program library in UPDATE format and has the following characteristics: 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, TIGSTEKV1 as file ID in HDR1 label, six files:

File 1 - installation job program library in UPDATE format,
File 2 - empty,
File 3 - empty,
File 4 - TIGS 1 Tektronix post processor source code program library in UPDATE format,
File 5 - vertification job in UPDATE PL format, and
File 6 - empty.

NOTES AND CAUTIONS

All limitations applicable to NOS/BE and INTERCOM also apply to TIGS 1 with the Tektronix 401X post processor.

All TIGS 1 subroutines are callable from application programs that have been written in FORTRAN Extended.

A CDC Synchronous interface (Tektronix product number 021-0135-01) is required on each Tektronix terminal to communicate in synchronous mode with the 6671 multiplexer or the 255X host communications processors.

Communications between the terminal and the central site are supported for TIGS 1 with the Tektronix 401X post processor according to table II-10-1.

TABLE II-10-1. TIGS/TEKTRONIX COMMUNICATIONS CAPABILITIES

Communications Mode	Baud Rate	Compatible Modem Type
Synchronous	4800	208B
	2400	201B
	2000	201A
Asynchronous	1200	212A
	300	103A

INSTALLATION DESCRIPTION

The installation of TIGS 1 with the Tektronix 401X post processor allows the user the option of generating either a segment/overlay loader library, TIGSLIB, or both a segment/overlay loader library, TIGSLIB, and a basic (nonsegment/nonoverlay) loader library, TIGSSFL.

The segment/overlay loader library, TIGSLIB, can be used in those situations where memory requirements of graphic applications exceed the available NOS/BE field length. The user can employ the segment/overlay loader library, TIGSLIB, in conjunction with the CYBER loader to segment or to overlay a graphic application and thereby reduce memory requirements. Note that the use of the segment/overlay loader library, TIGSLIB, without segmentation or overlays greatly increases an application program field length requirement. For example, the installation verification program VRFY requires 50350_8 field length using TIGSSFL and 54267_8 field length using TIGSLIB without segmentation or overlays. The segment/overlay loader library, TIGSLIB, must be used if an old neutral display file is to be accessed by the application program.

The basic (nonsegment/nonoverlay) loader library, TIGSSFL, can be used for graphic applications with memory requirements that do not exceed the NOS/BE available field length.

TIGSLIB is installed by default by the installation job. Both TIGSLIB and TIGSSFL are installed by the installation job by specifying =DEFINE,BASICLDR as an UPDATE directive.

The installation of TIGS 1 with the Tektronix 401X post-processor requires the installation of a PP routine into the running system. The PP routine, GCC-Change Program to Graphics Classification, gives the calling program graphics classification for the duration of program execution and returns the page width and length to the calling routine. GCC is installed into the running system by the TIGS installation job. All graphics activity should be stopped during the installation of TIGS.

INSTALLATION PROCEDURE

The procedure for installation of TIGS 1 with the Tektronix 401X post processor is outlined in the following two steps.

Step 1.

The installation job is file 1 on the Tektronix 401X post processor release tape (PL86B). File 1 is a program library in UPDATE format with = (equals) as the master control character.

A dummy ACCOUNT card has been included as part of the installation job. Perform an UPDATE run against file 1 of PL86B and list the COMPILE file to obtain the sequence number which corresponds to the ACCOUNT card (and any other control cards) which must be modified for installation. A listing of the COMPILE file may be obtained by executing the following procedure:

```
                           (MT1)    (7-track,  800 bpi.)
LISTI,T10,CM65000,         {HD1} .  (9-track,  800 cpi.)
                           (PE1)    (9-track, 1600 cpi.)
ACCOUNT(*)                (Supply appropriate accounting
                           information.)
LABEL(TIGSPL,R,L=TIGSTEKV1,
     ( HY )                        (7-track,  800 bpi.)
   D={ HD } ,VSN=PL86B)            (9-track,  800 cpi.)
     ( PE )                        (9-track, 1600 cpi.)
COPYBF(TIGSPL,OLDPL)
UNLOAD(TIGSPL)
UPDATE(F,U,*==,L=17)
COPYSBF(COMPILE,OUTPUT)
7/8/9
=IDENT,INSTALL
=DEFINE,NOSBE
          ( HY )  (7-track,  800 bpi.)
=DEFINE,  { HD }  (9-track,  800 cpi.)
          ( PE )  (9-track, 1600 cpi.)
=DEFINE,BASICLDR        (This is an optional UPDATE
                         directive.  If the user wishes
                         to install both a segment/
                         overlay loader library
                         (TIGSLIB) and a basic (non-
                         segment/nonoverlay) loader
                         library (TIGSSFL), then this
                         DEFINE statement must be
                         included.  If this DEFINE
                         statement is not included,
                         then only the control cards
                         required to install a segment/
                         overlay loader library will
                         be listed.)
6/7/8/9
```

Step 2

Perform a full UPDATE on file 1 of the Tektronix post processor release tape, (PL86B), modifying the installation job program library with the appropriate accounting information and DEFINE UPDATE directives required for installation. Route the resulting installation job to the NOS/BE input queue by executing the following procedure. NOTE: Only the Tektronix post processor release tape (PL86B) is required for the EXTRACT job, but both the Tektronix post processor release tape (PL86B) and the TIGS 1 preprocessor release tape (PL86A) are required for the installation job routed to the NOS/BE input queue.

```
EXTRACT,T10,CM65000,
    (MT1)     (7-track,  800 bpi.)
    {HD1} .   (9-track,  800 cpi.)
    (PE1)     (9-track, 1600 cpi.)
ACCOUNT(*)        (Supply appropriate accounting
                   information.)
LABEL(TIGSPL,R,L=TIGSTEKV1,
   ( HY )                        (7-track,  800 bpi.)
 D={ HD }   ,VSN=PL86B)          (9-track,  800 cpi.)
   ( PE )                        (9-track, 1600 cpi.)
COPYBF(TIGSPL,OLDPL)
UNLOAD(TIGSPL)
REQUEST(COMPILE,*Q)
UPDATE(F,*==,L=17,D,8)
ROUTE(COMPILE,DC=IN)
7/8/9
```

```
=IDENT, INSTALL
=DEFINE, NOSBE
           ( HY )       (7-track, 800 bpi.)
=DEFINE,   { HD }       (9-track, 800 cpi.)
           ( PE )       (9-track, 1600 cpi.)
=DEFINE, BASICLDR      (This is an optional UPDATE
                       directive. If the user wishes
                       to install both a segment/
                       overlay loader library
                       (TIGSLIB) and a basic
                       (nonsegment/nonoverlay)
                       loader library (TIGSSFL),
                       the =DEFINE, BASICLDR
                       directive must be included
                       as part of the UPDATE. If
                       the =DEFINE, BASICLDR
                       directive is not included as
                       part of the UPDATE, then
                       only a segment/overlay
                       loader library (TIGSLIB)
                       will be installed.)
=D, TIGS1.9
ACCOUNT(     )         (The user must modify the
                       installation job program
                       library with appropriate
                       accounting information.
                       The dummy ACCOUNT card
                       sequence number, 9, was
                       obtained from the compile
                       file listing resulting from
                       step 1 of the "Installation
                       Procedure.")
6/7/8/9
```

VERIFICATION PROCEDURE

Use the following steps for verification:

Step 1

The verification job is file 5 on the Tektronix post processor release tape, (PL86B). File 5 is a program library in UPDATE format with = (equals) as the master control character.

A dummy ACCOUNT card has been included as part of the verification job. Perform an UPDATE run against file 5 of PL86B and list the COMPILE file to obtain the sequence number which corresponds to the USER card (and any other control cards) which must be modified for installation. A listing of the compile file may be obtained by executing the following procedure:

```
                           (MT1)   (7-track, 800 bpi.)
LISTV, T10, CM65000,       {HD1} . (9-track, 800 cpi.)
                           (PE1)   (9-track, 1600 cpi.)
ACCOUNT(*)                 (Supply appropriate accounting
                           information.)
LABEL(TIGSPL, R, L=TIGSTEKV1,
      ( HY )                       (7-track, 800 bpi.)
D=    { HD }    VSN=PL86B)         (9-track, 800 cpi.)
      ( PE )                       (9-track, 1600 cpi.)
SKIPF(TIGSPL, 4, 17)
COPYBF(TIGSPL, OLDPL)
UNLOAD(TIGSPL)
UPDATE(F, U, *==, L=17)
COPYSBF(COMPILE, OUTPUT)
7/8/9
=IDENT, VERIFY
=DEFINE, NOSBE
6/7/8/9
```

Step 2

Perform a full UPDATE on file 5 of PL86B modifying the verification job program library with the appropriate accounting information and DEFINE UPDATE directives required for installation. Route the resulting verification job to the NOS/BE input queue by executing a procedure of the following form:

```
XTRACT, T10, CM65000,
 (MT1)                           (7-track, 800 bpi.)
 {HD1} .                         (9-track, 800 cpi.)
 (PE1)                           (9-track, 1600 cpi.)
ACCOUNT(*)              (Supply appropriate accounting
                        information.)
LABEL(TIGSPL, R, L=TIGSTEKV1,
       (HY)                      (7-track, 800 bpi.)
   D=  {HD} , VSN=PL86B)         (9-track, 800 cpi.)
       (PE)                      (9-track, 1600 cpi.)
SKIPF(TIGSPL, 4, 17)
COPYBF(TIGSPL, OLDPL)
UNLOAD(TIGSPL)
REQUEST(COMPILE, *Q)
UPDATE(F, *==, L=17, D, 8)
ROUTE(COMPILE, DC=IN)
7/8/9
=IDENT, VERIFY
=DEFINE, NOSBE
=D, TIGS2.4          (The user must modify the veri-
ACCOUNT(     )       fication job program library with
                     appropriate accounting informa-
                     tion. The dummy ACCOUNT
                     card sequence number, 4, was
                     obtained from the compile file
                     listing resulting from step 1 of
                     the "Verfication Procedure".)
6/7/8/9
```

Step 3.

Run the verification program, catalogued as permanent file VRFY by the verification job, to verify the proper installation of TIGS 1. The verification program may be run as follows:

1. Refer to the "Terminal Operating Procedures" to properly set up the terminal.

2. Log in to NOS/BE with a valid user name and password.

3. Attach the segment/overlay loader library permanent file and declare TIGSLIB a library:

```
ATTACH(TIGSLIB,ID=TEK)
LIBRARY(TIGSLIB)
```

4. Attach the verification program permanent file and execute the verification program:

```
ATTACH(VRFY,ID=TIGS)
VRFY.
```

5. The terminal operator will be asked to enter the terminal type (figure II-10-1). Enter the number which corresponds to the terminal type in use.

6. The terminal operator will be asked to enter the baud rate (figure II-10-1). Enter the baud rate in use.

7. The terminal operator will be asked if the tablet is to be used (see figure II-10-1). Enter N. The tablet is not required for the verification program.

8. The terminal operator will be asked if a hardcopy unit is available (see figure II-10-1). Enter N. The verification program does not require the availability of a hardcopy unit.

9. The terminal screen will be cleared and figure II-10-2 will be displayed if terminal type 4, 5, 6, or 7 was selected in step 5. Figure II-10-3 will be displayed if terminal type 2 or 3 was selected. Figure II-10-4 will be displayed if terminal type 1 was selected.

10. If terminal type 1 was selected in step 5, operations 11 through 14 will be skipped.

11. The operator is allowed to pick one of three buttons: WINDOW, RESTORE, or QUIT. The button is picked by moving the crosshairs over the text of the button and depressing an alpha key on the keyboard.

12. If WINDOW is picked, the terminal operator will be asked to locate the lower-left and then the upper-right corner of a new window to display. This is done by moving the crosshairs to the appropriate corner and depressing the T key on the keyboard. When both corners have been located, the new window will be displayed.

13. If RESTORE is picked, the original display will be displayed.

14. If QUIT is picked, the verification program will terminate.

15. If a basic (nonsegment/nonoverlay) loader library, TIGSSFL, was not installed, then go to operation 19.

16. Attach the basic (nonsegment/nonoverlay) loader library permanent file and declare TIGSSFL a library:

```
ATTACH(TIGSSFL,ID=TEK)
LIBRARY(TIGSSFL)
```

17. Execute the verification program:

```
VRFY.
```

18. Repeat operation 5 through 14 to verify the basic (nonsegment/nonoverlay) loader library, TIGSSFL.

19. Purge any unnecessary files generated during the installation or verification of TIGS 1.

20. Log off when finished.

TERMINAL OPERATING PROCEDURES

TEKTRONIX 4006 TERMINAL

Tektronix 4006 terminal operating procedures are as follows:

1. Turn the terminal power on by turning the POWER switch, located at the rear of the display unit, to the ON position.

2. Allow the terminal to warm up.

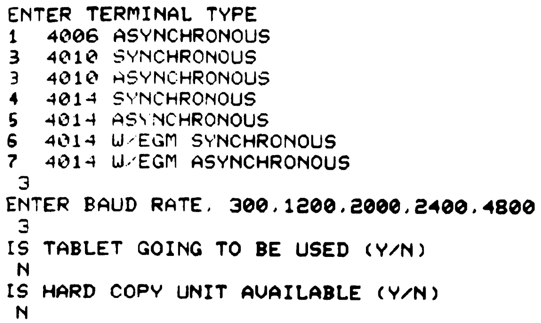
3. Depress the PAGE key to erase the screen and to position the cursor to home position (the upper left-hand corner of the screen).

4. Select the appropriate transmit and receive baud rates using the baud rate switches located at the rear of the display unit.

5. The setting of the FULL/HALF DUPLEX switch located at the rear of the display unit is NOS/BE installation dependent.

6. Turn the modem power on and verify that the modem is connected to the terminal.

7. Dial the appropriate telephone number.



```
ENTER TERMINAL TYPE
1  4006 ASYNCHRONOUS
3  4010 SYNCHRONOUS
3  4010 ASYNCHRONOUS
4  4014 SYNCHRONOUS
5  4014 ASYNCHRONOUS
6  4014 W/EGM SYNCHRONOUS
7  4014 W/EGM ASYNCHRONOUS
 3
ENTER BAUD RATE, 300,1200,2000,2400,4800
 3
IS TABLET GOING TO BE USED (Y/N)
 N
IS HARD COPY UNIT AVAILABLE (Y/N)
 N
```

Figure II-10-1. Terminal User is Asked to Enter Terminal Type, Baud Rate, if Tablet is to be Used, and if Hardcopy is Available

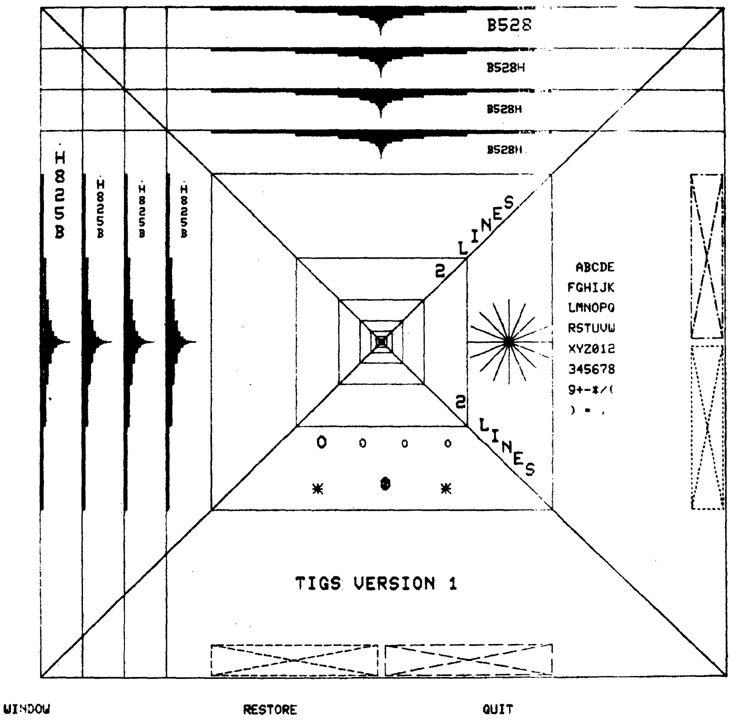


Figure II-10-2. Figure Displayed on Terminal Types 4, 5, 6, or 7

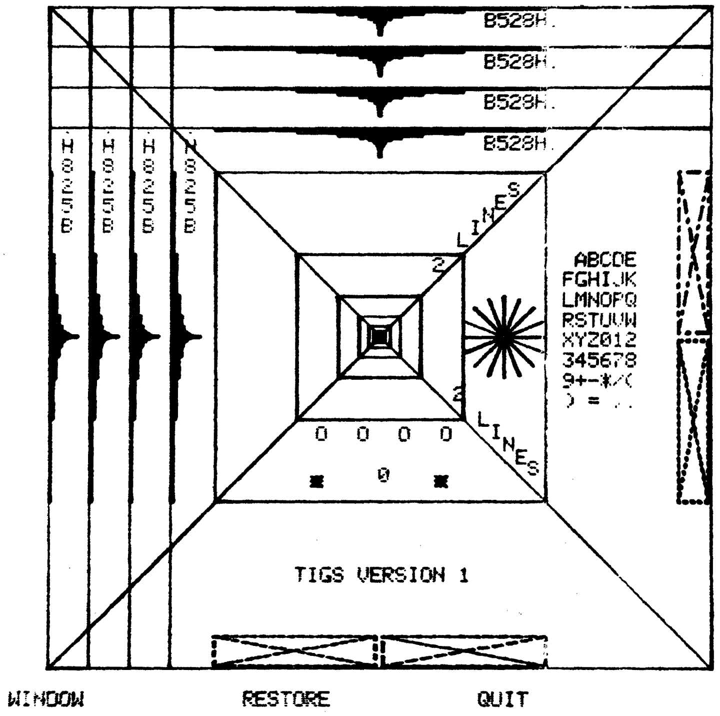


Figure II-10-3. Figure Displayed on Terminal Types 2 or 3

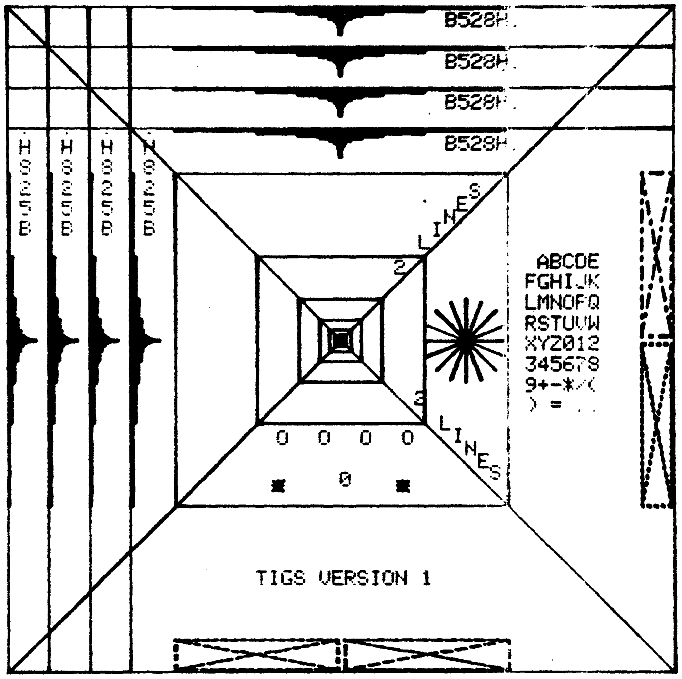


Figure II-10-4. Figure Displayed on Terminal Type 1

8. INTERCOM Version 4 will respond with a LOGIN message when ready for interactive command processing. INTERCOM Version 5, however, requires the user to depress the RETURN key after the terminal is polled before it will respond with a LOGIN message.

TEKTRONIX 4010/4014 TERMINALS WITH SYNCHRONOUS INTERFACE

Tektronix 4010 and 4014 terminals (equipped with Control Data synchronous interface) synchronous/asynchronous operating procedures are as follows:

1. Turn the terminal power on.

   - Tektronix 4010 terminal: The POWER switch is located beneath the keyboard at the top of the pedestal stand. The red POWER indicator, located at the top of the keyboard, will illuminate when power has been applied to the terminal.

   - Tektronix 4014 terminal: The POWER switch is located at the front lower right-hand corner of the pedestal stand. The green POWER indicator, located on the upper left-hand corner of the keyboard, will illuminate when power has been applied to the terminal.

2. Allow the terminal to warm up.

3. Press the PAGE key to erase the screen and to position the cursor to home position (the upper left-hand corner of the screen).

4. Set the ASCII/ALT switch to the ASCII position.

5. Select the appropriate baud rate switch settings. The switch should select the EXT position if the terminal is to be used for synchronous communications at 2000 and 4800 baud. Otherwise, the terminal baud rate switch setting will correspond to the baud rate in use.

   - Tektronix 4010 terminal: Select the appropriate transmit and receive baud rates using the baud rate switches located at the rear of the pedestal stand.

   - Tektronix 4014 terminal: Select the appropriate baud rate using the baud rate switch located at the rear of the pedestal stand.

6. The setting of the FULL/HALF DUPLEX switch is NOS/BE installation dependent.

7. Set the CODE EXPANDER switch to one of the following positions:

   - Synchronous mode communications: Turn the switch ON.

   - Asynchronous mode communications: Turn the switch OFF.

8. The following keyboard operations establish proper operating mode:

   - Synchronous mode communications: Place the terminal in local mode. Press the SHIFT key. While the SHIFT key is pressed, press the RESET PAGE key. Place the terminal in line mode and repeat the keyboard procedure.

   - Asynchronous mode communications: Place the terminal in local mode. Press and keep depressed the SHIFT key. Press and keep depressed the CNTL key. While both the SHIFT and CNTL keys are pressed, press the P key. Place the terminal in line mode.

9. Dial the appropriate telephone number.

10. INTERCOM Version 4 will respond with a LOGIN message when ready for interactive command processing. INTERCOM Version 5, however, requires the user to press the RETURN key after the terminal is polled before it will respond with a LOGIN message.

TEKTRONIX 4010/4014 TERMINALS WITHOUT SYNCHRONOUS INTERFACE

If the Tektronix 4010 and 4014 terminals are not equipped with the Control Data synchronous interface option, then the terminal can be operated in asynchronous mode only.

The Tektronix 4010 and 4014 terminals asynchronous mode operating procedures are essentially the same as for the synchronous mode operating procedures. However, step 8 of the synchronous mode operating procedures can be omitted. No special keyboard operations are required to establish mode switching since the terminals will always be in asynchronous mode when not equipped with the Control Data synchronous interface option.

SANDERS GRAPHIC 7 POST PROCESSOR OPTION

RELEASE MATERIALS

The Sanders Graphic 7 post processor runs under NOS/BE when installed with the TIGS 1 preprocessor.

The Sanders Graphic 7 post processor resides on the tape known as PL86C. PL86C is a program library in UPDATE format and has the following characteristics: 7-track (800 bpi.) or 9-track (800 or 1600 cpi.), binary recording mode, TIGSANDV1 as file ID in HDR1 label, six files:

File 1 - installation job program library in UPDATE format,
File 2 - empty,
File 3 - empty,
File 4 - TIGS 1 Sanders Graphic 7 post processor source code program library in UPDATE format,

File 5 - verification job in UPDATE PL format, and
File 6 - empty.

NOTES AND CAUTIONS

All limitations applicable to NOS/BE and INTERCOM 5 also apply to TIGS 1 with the Sanders Graphic 7 post processor.

All TIGS 1 subroutines are callable from application programs that have been written in FORTRAN Extended.

Communications between the terminal and the central site are supported for TIGS 1 with the Sanders Graphic 7 post processor according to table II-10-2.

TABLE II-10-2. TIGS/SANDERS COMMUNICATIONS CAPABILITIES

Communication Mode	Baud Rate	Compatible Modem Type
Asynchronous	1200	212A
	300	103A

INSTALLATION DESCRIPTION

The installation of TIGS 1 with the Sanders Graphic 7 post processor allows the user the option of generating either a segment/overlay loader library, TGR7LIB or both a segment/overlay loader library, TGR7LIB, and a basic (nonsegment/nonoverlay) loader library, TGR7SFL.

The segment/overlay loader library, TGR7LIB, can be used in those situations where memory requirements of graphic applications exceed the available NOS/BE field length. The user can employ the segment/overlay loader library, TGR7LIB, in conjunction with CYBER loader to segment or to overlay a graphic application and thereby reduce memory requirements. Note that use of the segment/overlay loader library, TGR7LIB, without segmentation or overlays increases an application program field length requirement. For example, the installation verification program VRFY requires 56374 (octal) field length using TGR7SFL and 62432 (octal) field length using TGR7LIB without segmentation or overlays.

The basic (nonsegment/nonoverlay) loader library, TGR7SFL, can be used for graphic applications with memory requirements that do not exceed the NOS/BE available field length.

TGR7LIB is installed by default by the installation job. Both TGR7LIB and TGR7SFL are installed by the installation job by specifying =DEFINE, BASICLDR as an UPDATE directive.

The installation of TIGS 1 with the Sanders Graphic 7 post processor requires the installation of a PP routine into the running system. The PP routine, GCC-Change Program to Graphics Classification, gives the calling program graphics classification for the duration of program execution and returns the page width and length to the calling routine. GCC is installed into the running system by the TIGS installation job. All graphics activity should be stopped during the installation of TIGS.

The amount of refresh memory available in the Sanders Graphic 7 terminal must be specified during installation of the post processor. The allowable sizes of refresh memory are 8K, 16K, and 24K words. Table II-10-3 lists these options and the corresponding installation job directives.

TABLE II-10-3. REFRESH MEMORY SIZES

Refresh Memory Size	Required Installation Job UPDATE Directive
8K words	=DEFINE, REF8K
16K words	=DEFINE, REF16K
24K words	=DEFINE, REF24K

INSTALLATION PROCEDURE

The procedure for installation of TIGS 1 with the Sanders Graphic 7 post processor is outlined in the following two steps.

Step 1.

The installation job is file 1 on the Sanders Graphic 7 post processor release tape (PL86C). File 1 is a program library in UPDATE format with = (equals) as the master control character.

A dummy ACCOUNT card has been included as part of the installation job. Perform an UPDATE run against file 1 of PL86C and list the COMPILE file to obtain the sequence number which corresponds to the ACCOUNT card (and any other control cards) which must be modified for installation. A listing of the COMPILE file may be obtained by executing the following procedure:

```
                        (MT1)   (7-track, 800 bpi.)
LISTI, T10, CM65000,    {HD1} . (9-track, 800 cpi.)
                        (PE1)   (9-track, 1600 cpi.)
ACCOUNT(*)              (Supply appropriate accounting
                        information.)
LABEL(TIGSPL, R, L=TIGSANDV1,
      (HY)                      (7-track, 800 bpi.)
  D=  {HD}  , VSN=PL86C)        (9-track, 800 cpi.)
      (PE)                      (9-track, 1600 cpi.)
COPYBF(TIGSPL, OLDPL)
UNLOAD(TIGSPL)
UPDATE(F, U, *==, L=17)
COPYSBF(COMPILE, OUTPUT)
7/8/9

=IDENT, INSTALL
=DEFINE, NOSBE
              (HY)   (7-track, 800 bpi.)
=DEFINE,      {HD}   (9-track, 800 cpi.)
              (PE)   (9-track, 1600 cpi.)
```

```
=DEFINE,BASICLDR        (This is an optional UPDATE
                        directive. If the user wishes
                        to install both a segment/
                        overlay loader library
                        (TGR7LIB) and a basic
                        (nonsegment/nonoverlay)
                        loader library (TGR7SFL),
                        then this DEFINE statement
                        must be included. If this
                        DEFINE statement is not
                        included, then only the con-
                        trol cards required to
                        install a segment/overlay
                        loader library will be listed.)
          ⎧REF8K ⎫
=DEFINE,  ⎨REF16K⎬
          ⎩REF24K⎭
                        (Specify the amount of re-
                        fresh memory available in
                        the terminal.)
6/7/8/9
```

Step 2

Perform a full UPDATE on file 1 of the Sanders Graphic 7 post processor release tape, (PL86C), modifying the installation job program library with the appropriate accounting information and DEFINE UPDATE directives required for installation. Route the resulting installation job to the NOS/BE input queue by executing the following procedure. NOTE: Only the Sanders Graphic 7 post processor release tape (PL86C) is required for the EXTRACT job, but both the Sanders Graphic 7 post processor release tape (PL86C) and the TIGS 1 preprocessor release tape (PL86A) are required for the installation job routed to the NOS/BE input queue.

```
EXTRACT,T10,CM65000,
⎧MT1⎫          (7-track, 800 bpi.)
⎨HD1⎬  .       (9-track, 800 cpi.)
⎩PE1⎭          (9-track, 1600 cpi.)
ACCOUNT(*)     (Supply appropriate accounting
               information.)
LABEL(TIGSPL,R,L=TIGSANDV1
     ⎧HY⎫                     (7-track, 800 bpi.)
  D= ⎨HD⎬  ,VSN=PL86C)        (9-track, 800 cpi.)
     ⎩PE⎭                     (9-track, 1600 cpi.)
COPYBF(TIGSPL,OLDPL)
UNLOAD(TIGSPL)
REQUEST(COMPILE,*Q)
UPDATE(F,*=,L=17,D,8)
ROUTE(COMPILE,DC=IN)
7/8/9

=IDENT,INSTALL
=DEFINE,NOSBE
          ⎧HY⎫  (7-track, 800 bpi.)
=DEFINE,  ⎨HD⎬  (9-track, 800 cpi.)
          ⎩PE⎭  (9-track, 1600 cpi.)
=DEFINE,BASICLDR        (This is an optional UPDATE
                        directive. If the user wishes
                        to install both a segment/
                        overlay loader library
                        (TGR7LIB) and a basic
                        (nonsegment/nonoverlay)
                        loader library (TGR7SFL),
                        the =DEFINE,BASICLDR
                        directive must be included
                        as part of the UPDATE. If
                        the =DEFINE,BASICLDR
                        directive is not included as
                        part of the UPDATE, then
                        only a segment/overlay
                        loader library (TGR7LIB)
                        will be installed.)
          ⎧REF8K ⎫
=DEFINE,  ⎨REF16K⎬
          ⎩REF24K⎭
                        (Specify the amount of re-
                        fresh memory available in
                        the terminal.)
=D,TIGS1.9
ACCOUNT(   )            (The user must modify the
                        installation job program
                        library with appropriate
                        accounting information. The
                        dummy ACCOUNT card
                        sequence number, 9, was
                        obtained from the compile
                        file listing resulting from
                        step 1 of the "Installation
                        Procedure.")
6/7/8/9
```

VERIFICATION PROCEDURE

Use the following steps for verification:

Step 1

The verification job is file 5 on the Sanders Graphic 7 post processor release tape, (PL86C). File 5 is a program library in UPDATE format with = (equals) as the master control character.

A dummy ACCOUNT card has been included as part of the verification job. Perform an UPDATE run against file 5 of PL86C and list the COMPILE file to obtain the sequence number which corresponds to the ACCOUNT card (and any other control cards) which must be modified for installation. A listing of the COMPILE file may be obtained by executing the following procedure:

```
                          ⎧MT1⎫  (7-track,  800 bpi.)
LISTV,T10,CM65000,        ⎨HD1⎬ . (9-track,  800 bpi.)
                          ⎩PE1⎭  (9-track, 1600 cpi.)
ACCOUNT(*)                (Supply appropriate account-
                          ing information.)
LABEL(TIGSPL,R,L=TIGSANDV1,
    ⎧HY⎫                       (7-track,  800 bpi.)
  D=⎨HD⎬ , VSN=PL86C)          (9-track,  800 cpi.)
    ⎩PE⎭                       (9-track, 1600 cpi.)
SKIPF(TIGSPL,4,17)
COPYBF(TIGSPL,OLDPL)
UNLOAD(TIGSPL)
UPDATE(F,U,*==,L=17)
COPYSBF(COMPILE,OUTPUT)
7/8/9

=IDENT,VERIFY
=DEFINE,NOSBE
6/7/8/9
```

Step 2.

Perform a full UPDATE on file 5 of PL86C modifying the verification job program library with the appropriate accounting information and DEFINE UPDATE directives required for installation. Route the resulting verification job to the NOS/BE input queue by executing a procedure of the following form:

```
XTRACT,T10,CM65000,
 ⎧MT1⎫         (7-track,  800 bpi.)
 ⎨HD1⎬ .       (9-track,  800 cpi.)
 ⎩PE1⎭         (9-track, 1600 cpi.)
ACCOUNT(*)     (Supply appropriate accounting
               information.)
LABEL(TIGSPL,R,L=TIGSANDV1,
    ⎧HY⎫                       (7-track,  800 bpi.)
  D=⎨HD⎬ , VSN=PL86C)          (9-track,  800 cpi.)
    ⎩PE⎭                       (9-track, 1600 cpi.)
SKIPF(TIGSPL,4,17)
COPYBF(TIGSPL,OLDPL)
UNLOAD(TIGSPL)
REQUEST(COMPILE,*Q)
UPDATE(F,*==,L=17,D,8)
ROUTE(COMPILE,DC=IN)
7/8/9

=IDENT,VERIFY
=DEFINE,NOSBE
=D,TIGS2.4
ACCOUNT(    )            (The user must modify the
                         verification job program
                         library with appropriate
                         accounting information. The
                         dummy ACCOUNT card
                         sequence number, 4, was
                         obtained from the compile
                         file listing resulting from
                         step 1 of the "Verification
                         Procedure.")
6/7/8/9
```

Step 3.

Run the verification program, catalogued as permanent file VRFY by the verification job, to verify the proper installation of TIGS 1. The verification program may be run as follows:

1. Refer to the "Terminal Operating Procedures" to properly set up the terminal.

2. Log in to NOS/BE - INTERCOM 5 with a valid user name and password.

3. Attach the segment/overlay loader library permanent file and declare TGR7LIB a library:

   ```
   ATTACH(TGR7LIB,ID=SAND)
   LIBRARY(TGR7LIB)
   ```

4. Attach the verification program permanent file and execute the verification program:

   ```
   ATTACH(VRFY,ID=TIGS)
   VRFY.
   ```

5. The terminal operator will be asked if a lightpen is available (see figure II-10-5). Enter Y if it is available; otherwise, enter N.

6. The terminal operator will be asked if a trackball, or joystick, is available (see figure II-10-5). Enter Y if it is available; otherwise enter N.

7. The terminal operator will be asked if a hardcopy unit is available (see figure II-10-5). Enter N. The verification program does not require the availability of a hardcopy unit.

8. The terminal operator will be asked if the error display should be enabled (see figure II-10-5). Enter Y.

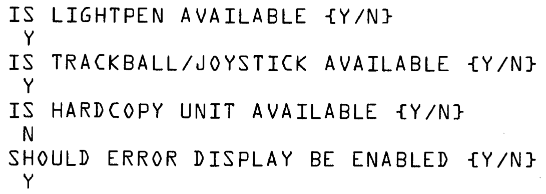
9. Figure II-10-6 will be displayed.

10. If neither a lightpen nor a trackball (joystick) are available, operations 11 through 14 are skipped.

11. The operator is allowed to pick one of three buttons: WINDOW, RESTORE, or QUIT. The button is picked in one of the following ways:

    - If the lightpen is available, the operator positions the pen tip over the text of the button and depresses the tip.

    - If only the trackball (joystick) is available, the operator positions the center of the cursor over the text of the button and depresses the function key labelled FØ.



```
IS LIGHTPEN AVAILABLE {Y/N}
 Y
IS TRACKBALL/JOYSTICK AVAILABLE {Y/N}
 Y
IS HARDCOPY UNIT AVAILABLE {Y/N}
 N
SHOULD ERROR DISPLAY BE ENABLED {Y/N}
 Y
```

Figure II-10-5. Terminal User is Asked to Enter Availability of Lightpen, Trackball/Joystick, and Hardcopy, and Whether Error Display is to be Enabled

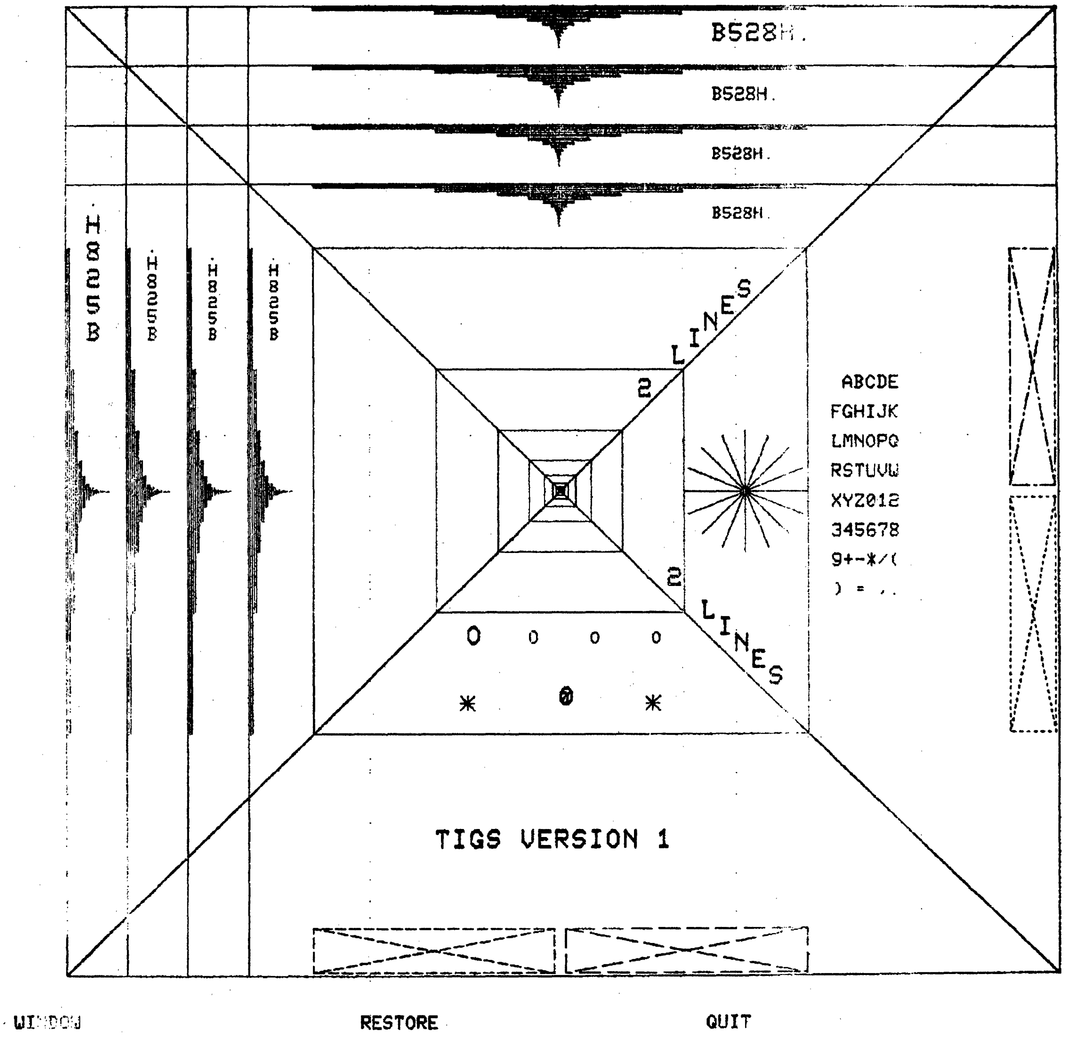


Figure II-10-6. Figure Displayed on the Sanders Graphic 7 Terminal

12. If WINDOW is picked, the terminal operator will be asked to locate the lower-left and then the upper-right corner of a new window to display. This is done in one of the following ways:

    - If the lightpen is available, the operator positions the pen tip at the lower-left corner of the window selected, and depresses the tip, and then does the same for the upper-right corner. Finally the operator depresses the function key labelled F15, after waiting for prompting at each step.
    - If only the trackball (joystick) is available, the operator positions the center of the cursor at the lower-left corner of the window selected and depresses the function key labelled FØ, and then does the same for the upper-right corner. Finally the operator depresses the function key labelled F15, after waiting for prompting at each step.

13. If RESTORE is picked, the original display is displayed.

14. If QUIT is picked, the verification program terminates.

15. If a basic (nonsegment/nonoverlay) loader library, TGR7SFL, was not installed, then go to operation 19.

16. Attach the basic (nonsegment/nonoverlay) loader library permanent file and declare TIGSSFL a library:

    ```
    ATTACH(TGR7SFL,ID=SAND)
    LIBRARY(TGR7SFL)
    ```

17. Execute the verification program:

    ```
    VRFY.
    ```

18. Repeat operation 5 through 14 to verify the basic (nonsegment/nonoverlay) loader library, TGR7SFL.

19. Purge any unnecessary files generated during the installation or verification of TIGS 1.

20. Log off when finished.

TERMINAL OPERATING PROCEDURES

The Sanders Graphic 7 operating procedures are as follows:

1. Turn the terminal power on by pushing in the ON/OFF button on the front of the display and flipping the ON/OFF switch, located behind the front panel of the controller unit, to ON.
2. Push the LOCAL button on the front panel of the controller unit to bring up a test pattern on the display screen.
3. Depress the carriage return key on the keyboard to bring down the test pattern. An M will appear in the center of the screen. Then depress Y followed by carriage return. A G7 F will appear at the top of the screen indicating that the terminal is in teletype emulation mode.
4. Switch the terminal to half duplex by depressing the function key labelled F1.
5. Turn the modem power on and verify that the modem is connected to the terminal.
6. Dial the appropriate phone number.
7. INTERCOM Version 5 requires the user to press the RETURN key after the terminal is polled before it will respond with a LOGIN message.

TOTAL UNIVERSAL 1

RELEASE DESCRIPTION

TOTAL Universal 1 consists of the following relocatable binaries:

DBGEN (Data base generation program) - this program reads user-prepared DBDL statements and generates COMPASS source statements which in turn produce the data base descriptor module.

DBFMT (Data set format program) - this program reads format parameter statements, and, utilizing a data base descriptor module, preformats the data sets.

DATBAS (Data base interface module) - this module serves as an interface between the user application program and the TOTAL and data base descriptor modules.

TOTAL (Data base management module) - this module provides the data management capability of the system, interpreting and executing the various DML commands from the user application program.

DBRCV (Data base recovery module) - this module, optional in use, provides the ability to recover record images from the TOTAL logging file.

These files are to be installed on the user's permanent file library.

Hardware Requirements

TOTAL Universal 1 can be maintained and run on the same minimum hardware requirements as NOS/BE.

Deficiencies

None.

INSTALLATION PROCEDURES

TOTAL Universal 1 resides on the tape PL49. PL49 is a program library in binary format. It has the following characteristics: labeled, 7- or 9-track, 800 bpi, binary recording mode, TOTALU1P0 as file ID in HDR1 label, six files:

File 1 - installation and verification deck in UPDATE format,
File 2 - empty,
File 3 - five TOTAL relocatable binaries (described earlier),
File 4 - empty,
File 5 - empty, and
File 6 - sample output of installation and validation job.

Installation consists of the following steps:

1. Determine update directives to modify the install deck (as described in the next section).

2. Create the following job to call the install/validate job from the install tape and submit it as a batch job:

```
JOB/ACCOUNT.           (Supply valid JOB/
                        ACCOUNT card.)
LABEL,TOT,R,L=TOTALU1P0,D=HY,
   VSN=PL49.
REQUEST,COMPILE,*Q.
UPDATE,Q,*=$$$$,P=TOT,L=F.
ROUTE,COMPILE,DC=IN.
7/8/9
UPDATE directives
6/7/8/9
```

3. Validate the install/validate job output against the sample output copied from the install tape.

INSTALLATION PARAMETERS

Following is a sample UPDATE directive deck to be used as described in the previous section. All options are shown and explained.

```
$INST
$COMPILE INSTALL
{$DEFINE MT}           (7-track install tape.)
{$DEFINE NT}           (9-track install tape.)
$DELETE INSTALL.2,INSTALL.3
 JOB/ACCOUNT           (Site/user dependent.)
```

INSTALLATION JOB

The installation/validation job performs the following:

1. Issue LABEL to have install tapes assigned to job.

2. REQUEST's permanent file space for the five TOTAL files, copies them from tape to the permanent files, and catalogues the permanent files as in the following example (no passwords are assigned):

```
CATALOG,DBGEN,ID=TOTAL
```

3. Validation is begun by defining a data base using DBGEN.

4. The test data base is formatted with DBFMT.

5. Sample data is placed in the data base with a COBOL program, and retrievals are performed. DATBAS and TOTAL are verified in this phase.

6. DBRCV is next run against the data base for its validation.

7. Finally, the sample output of the install/validation job is copied from the install tape to output.

TOTAL UNIVERSAL 2

RELEASE DESCRIPTION

TOTAL Universal 2 consists of the following relocatable binaries:

DBGEN (Data base generation program) - this program reads user-prepared DBDL statements and generates COMPASS source statements which in turn produce the data base descriptor module.

DBFMT (Data set format program) - this program reads format parameter statements, and, utilizing a data base descriptor module, preformats the data sets.

DATBAS (Data base interface module) - this module serves as an interface between the user application program and the TOTAL and data base descriptor modules.

TOTAL (Data base management module) - this module provides the data management capability of the system, interpreting and executing the various DML commands from the user application program.

DBRCV (Data base recovery module) - this module, optional in use, provides the ability to recover record images from the TOTAL logging file.

TOTUTIL (TOTAL utilities) - this program provides TOTAL users with a generalized utility package capable of performing file loading/unloading, statistics, and other data base functions.

These files are to be installed on the user's permanent file library.

Hardware Requirements

TOTAL Universal 2 can be maintained and run on the same minimum hardware requirements as NOS/BE.

Deficiencies

None.

RELEASE MATERIALS

TOTAL UNIVERSAL 2 release materials are contained on program library tape PL75. PL75 has the following characteristics: labeled, 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, TOTALU2P0 as file ID in HDR1 label. PL75 contains seven files:

File 1 - installation and verification deck in UPDATE format,

File 2 - empty,

File 3 - five TOTAL relocatable binaries: DBGEN, DBFMT, DATBAS, TOTAL, DBRCV,

File 4 - source code for TOTAL utilities in UPDATE format,

File 5 - empty,

File 6 - COBOL compilation output from the validation job, and

File 7 - execution output from the validation job.

INSTALLATION PROCEDURES

Installation consists of the following steps:

1. Determine UPDATE directives to modify the install deck (as described in the next section).

2. Create the following job to call the install/validate job from the install tape and submit it as a batch job:

```
JOB/ACCOUNT.              (Supply valid JOB/
                           ACCOUNT card.)
LABEL,TOT,R,L=TOTALU2P0,D= ( HY ) ,
   VSN=PL75.               { HD }
REQUEST,COMPILE,*Q.        ( PE )
UPDATE,Q,*=$$$$,P=TOT,L=F.
ROUTE,COMPILE,DC=IN.
7/8/9
UPDATE directives
6/7/8/9
```

3. Validate the install/validate job output against the sample output copied from the install tape.

INSTALLATION PARAMETERS

Following are the UPDATE directives to be used as described in the previous section. All options are shown and explained.

```
 $IDENT INST
 $DEFINE CBL4†    (Compiles utilities with COBOL 4 -
                  Default is COBOL 5)
{$DEFINE HY }     (7-track 800 bpi)  } Select one to
{$DEFINE HD }     (9-track 800 cpi)  } correspond to
{$DEFINE PE }     (9-track 1600 cpi) } the installation
                                       tape density.
 $DELETE INSTALL.2,INSTALL.3
 JOB/ACCOUNT CARD.  (Supply valid JOB/
                     ACCOUNT card.)
 $COMPILE INSTALL
```

INSTALLATION JOB

The installation/validation job performs the following:

1. Issues LABEL card to have install tape assigned to job.

2. REQUEST's permanent file space for the five TOTAL files, copies them from tape to the permanent files and catalogues the permanent files as in the following example:

   CATALOG,DBGEN,ID=TOTAL.

3. Compiles TOTAL utilities and creates the relocatable binary on the permanent file TOTUTIL.

4. Validation is begun by defining a data base using DBGEN.

5. The test data base is formatted with DBFMT.

6. Sample data is placed in the data base with a COBOL program, and retrievals are performed. DATBAS and TOTAL are verified in this phase.

7. DBRCV is run against the data base for its validation.

8. The TOTAL utilities are validated by executing the statistics option.

9. Finally, the sample output of the install/validation job is copied from the install tape to output for verification.

FILES CREATED

The installation/validation job creates the following permanent files with ID=TOTAL and no passwords assigned:

DBGEN DBFMT DATBAS TOTAL DBRCV	Five TOTAL binaries
TOTUTIL -	TOTAL utility binary
CUSTDB -	Validation DBMOD
CUCUST CUINVF	Validation data base files

The user should insure that these files will not conflict with already existing files prior to installation.

[†] The ability to compile the utilities using COBOL 4 is provided as a user convenience at this time, but may not be supported in future releases.

RELEASE DESCRIPTION

TOTAL/ATHENA consists of four relocatable binaries:

```
ATB1 ⎫   Modules that interface with the TOTAL
ATB2 ⎭   data base management system.
```

ATRP = the Report Writer module.

ATPLT = the Plotter module.

HARDWARE REQUIREMENTS

TOTAL/ATHENA can be maintained and run on the same minimum hardware requirements as NOS/BE.

SOFTWARE REQUIREMENTS

TOTAL Universal 2 must be installed before TOTAL/ATHENA installation is attempted.

INSTALLATION PROCEDURES

PL62 contains six files:

File 1 - install deck in UPDATE format,

File 2 - empty,

File 3 - four relocatable binaries:

```
-ATB1 ⎫   TOTAL/ATHENA query update
-ATB2 ⎭   binaries,
-ATRP     TOTAL/ATHENA Report Writer, and
-ATPLT    TOTAL/ATHENA Plotter;
```

File 4 - TOTAL/ATHENA, Report Writer, and Plotter source code deck in UPDATE format,

File 5 - demonstration decks in UPDATE format:

-TOTAL/ATHENA,
-REPORT WRITER,
-PLOTTER, and
-TEXT INFORMATION;

File 6 - sample output from demonstration jobs.

Installation consists of the following steps:

1. Choose the UPDATE directives that are required in step 2 to modify the install deck.

   a. Select a DEFINE,MT or DEFINE,NT or DEFINE,NTPE for install tape processing.

   b. Select a DEFINE,BINARY or DEFINE,SOURCE. For execution of TOTAL/ATHENA, Report Writer and Plotter compiled binaries, select the DEFINE,BINARY. For users to compile their own copies of TOTAL/ATHENA, Report Writer and Plotter, select the DEFINE,SOURCE control directive.

   c. Select DEFINE,DEBUG if COBOL DEBUG mode is desired.

2. Create and submit for batch processing the following job to call the install deck from the install tape:

```
     ⎧MT1⎫
JOB, ⎨HD1⎬ .
     ⎩PE1⎭
ACCOUNT.                                 ⎧HY⎫
LABEL,ATH,R,L=TOTALATHENA,D=             ⎨HD⎬ ,
    VSN=PL62.                            ⎩PE⎭
REQUEST,COMPILE,*Q.
UPDATE(Q,P=ATH,L=F,*=/)
ROUTE(COMPILE,DC=IN)
7/8/9
/ID INST
        ⎧MT (7 TRACK TAPE) ⎫
/DEFINE ⎨NT (9 TRACK TAPE) ⎬       (Select MT,NT
        ⎩NTPE              ⎭       or NTPE)
    (9 TRACK TAPE - 1600 cpi)
/DEFINE ⎧SOURCE⎫                   Select BINARY
        ⎩BINARY⎭                   or SOURCE
[ /DEFINE,DEBUG]
/DELETE INSTALL.2,INSTALL.3
        ⎧MT1⎫
JOBNAME ⎨HD1⎬                     (choose)
        ⎩PE1⎭
ACCOUNT.     (Insert correct ACCOUNT here.)
/D INSTALL.103,INSTALL.104
              ⎧MT1⎫
JOBNAME,T200, ⎨HD1⎬               (choose)
              ⎩PE1⎭
ACCOUNT.     (Insert correct ACCOUNT here.)
/COMPILE INSTALL
6/7/8/9
```

3. Verify that the output from the demonstration job matches the listing of the install tape sample output file. Verify that the output obtained from interactively executing the ATHENA plotter matches the output listed in the demonstration job text file.

INSTALLATION JOB

The installation jobs perform the following:

1. Issue LABEL to have PL62 assigned to job.

2. For installation of the binary files (/DEFINE,BINARY), the install job defines the two TOTAL ATHENA files, the Report Writer and Plotter files, and copies them from release PL62 to the permanent files. No passwords are assigned.

3. For the user to compile binaries from source code (/DEFINE,SOURCE) the install job defines the two TOTAL/ATHENA files, the Report Writer file and Plotter file, copies and updates the source code from tape to disk and compiles the source code generating TOTAL/ATHENA, Report Writer and Plotter binaries.

4. A TOTAL/ATHENA demonstration is begun by defining a data base using DBGEN (TOTAL module).

5. A test data base is formatted with DBFMT (TOTAL module).

6. Sample data is placed in the data base with TOTAL/ATHENA and then TOTAL/ATHENA retrievals and functions are performed.

7. A TOTAL/ATHENA Report Writer demonstration is then processed with SUBSET and COMMAND file data supplied on the install tape.

8. A TOTAL/ATHENA plotter demonstration should then be run interactively by the installer. The plotter data base and a text file of instructions to the installer is supplied on the install tape. The needed files are catalogued by the installation job, and the text file of instructions is printed out following the ATHENA Report Writer demonstration.

   When the plotter verification is complete, the installer should compare all the demonstration output with the expected output which was copied from the install tape.

This job lists the install deck as it appears on file 1 of PL62:

```
          ⎧MT1⎫
JOBNAME,  ⎨HD1⎬ .
          ⎩PE1⎭
ACCOUNT.
                                   ⎧HY⎫
LABEL(ATH,R,L=TOTALATHENA,D=       ⎨HD⎬  ,
   VSN=PL62)                       ⎩PE⎭
UPDATE(Q,L=F,P=ATH,*=/)
7/8/9
/ID ATHLIST
/COMPILE INSTALL
6/7/8/9
```

DEMONSTRATION JOB

The demonstration deck in UPDATE format includes the demonstration job and the necessary input records to the demonstration job. The demonstration job performs these operations:

1. Generates a TOTAL DBMOD and formats a TOTAL data base.

2. Creates and executes the TOTAL/ATHENA query/update executable binary.

3. Executes the TOTAL/ATHENA report generator module.

4. Defines and copies a TOTAL/ATHENA plotter SUBSET file.

5. Prints a text file that describes how to verify the TOTAL/ATHENA plotter.

This job lists the demonstration deck as it appears on file 5 of the install tape:

```
          ⎧MT1⎫
JOBNAME,  ⎨HD1⎬ .
          ⎩PE1⎭
ACCOUNT.                              ⎧HY⎫
LABEL(ATH,R,L=TOTALATHENA,D=          ⎨HD⎬  ,
   VSN=PL62)                          ⎩PE⎭
SKIPF,ATH,4,17.
UPDATE(Q,L=F,P=ATH,*=$$$$,R)
7/8/9
$ID ATHLIST
$COMPILE TADEMO
6/7/8/9
```

NOTE

The binary files on this release were generated on a non-CMU CYBER system.

UNIPLOT 2 13

RELEASE DESCRIPTION

UNIPLOT 2 (Universal Plotting Software Version 2) runs under NOS/BE to provide a standard interface between an application program and various plotting or display devices.

The UNIPLOT subroutines are callable from user application programs only under FORTRAN Extended, and provide the user with a standard format plot data file and the capability of obtaining a plot on a variety of devices without regenerating the data.

The installation of UNIPLOT 2 requires the installation of the device dependent plot routines supplied by the plotter manufacturer.

HARDWARE CONFIGURATION

The minimum hardware configuration required by NOS/BE is required to support UNIPLOT.

HARDWARE OPTIONS

A graphics terminal or plotting device is required if plotting or displaying of plot data is desired to be done in-house. Those currently supported by UNIPLOT are:

1. CalComp's 500 and 700 series plotters and controllers,
2. Houston Instruments BRC-7 and MTR-4 controllers,
3. ZETA models 230, 1240 and 3640,
4. Tektronix 40XX series graphics terminals, and
5. Control Data CYBER graphics terminal.

RELEASE MATERIALS

UNIPLOT 2 release consists of a single magnetic tape, PL71, containing six files of data as follows:

File 1 - installation/maintenance job program library,

File 2 - absolute overlay of UNIPLOT post-processor,

File 3 - user library of UNIPLOT neutral plot routines in EDITLIB format,

File 4 - program library of UNIPLOT in UPDATE format,

File 5 - verification job program library, and

File 6 - empty.

PSR MODIFICATIONS

The PSR deck on the corrective code tape which is applicable to UNIPLOT 2 is UNP. The idents within this deck are of the form UNPxxxx. The initial release of PL71 is at LV444.

The UNIPLOT 2 installation/maintenance job assumes that the PSRs are on a permanent file called MINIT,ID=CCT. MINIT is assumed to be a program library with a master character of / (slash).

NOTES AND CAUTIONS

All limitations applicable to NOS/BE also apply to UNIPLOT 2.

All subroutines in UNIPLOT are callable from user application programs in FORTRAN Extended only.

If the ZETA or Tektronix devices are to be used, INTERCOM is required.

INSTALLATION PROCEDURE

The installation of UNIPLOT 2 requires the cataloging of the UNIPLOT 2 neutral plot routine library and the post-processor absolute (UNIPOST). The job required to install UNIPLOT 2 is released as an installation deck on file 1 of the release tape. This job is a program library in UPDATE format with = (equal) as the master character. A dummy ACCOUNT card is part of this job. The user should first perform an UPDATE run and list the compile file to discover the sequence numbers. As part of the EXTRACT procedure, the user should modify the JOB/ACCOUNT cards to conform to this installation.

To obtain and execute the installation or maintenance job, a procedure of the form shown below should be used:

```
          {MT1}
EXTRACT,  {NT1}
ACCOUNT.              (Supply valid ACCOUNT card.)
                                 {HY}
LABEL(PL71,R,L=UNIPLOTV2,D=      {HD}  ,
   VSN=PL71)
COPYBF(PL71,OLDPL)
REWIND(OLDPL)
UPDATE(Q,*==,C,D,8)
ROUTE(COMPILE,DC=IN)
7/8/9
```

```
=D,INSTAL.XX          (Use for initial installation.)
=D PSR.XX             (Use for maintenance run.)
ACCOUNT.
=COMPILE INSTAL       (Use for initial installation.)
=COMPILE PSR          (Use for maintenance run.)
6/7/8/9
```

Both the installation and maintenance jobs initiated by the EXTRACT job will catalog the following three permanent files.

File 1 - UNIPOST,ID=UNIPLOT - absolute code for UNIPLOT post-processor,

File 2 - UNIPLOT,ID=UNIPLOT - user library of UNIPLOT neutral plot routines, and

File 3 - POSTLGO,ID=UNIPLOT - relocatable binary of UNIPLOT post-processor.

The maintenance run, when the IDENT PSR is used will, in addition, generate a new PL71 tape containing the corrective code.

INSTALLING A NEW DEVICE

A labeled common block, /PPREQ/, is used by the post-processor main overlay to communicate with the device routines in the primary overlay. Each time control is returned to the primary overlay, /PPREQ/ contains information which directs the overlay's main program to call a specific device routine with parameter values also contained in /PPREQ/.

In order to support a new plotting device, the programmer must write a new overlay main program that calls initialization, vector plotting, pen select, termination and possible character plotting routines specific to the device being supported. The subroutine DEVLIST in the main overlay must be modified to include the keyname of the new overlay. Then the post-processor must be reloaded along with the new overlay.

MODIFYING ROUTINE DEVICE LIST

The modifying routine device list (DEVLIST) conforms to the following:

1. The dimension of the array DEVICE must be incremented by the number of devices added.

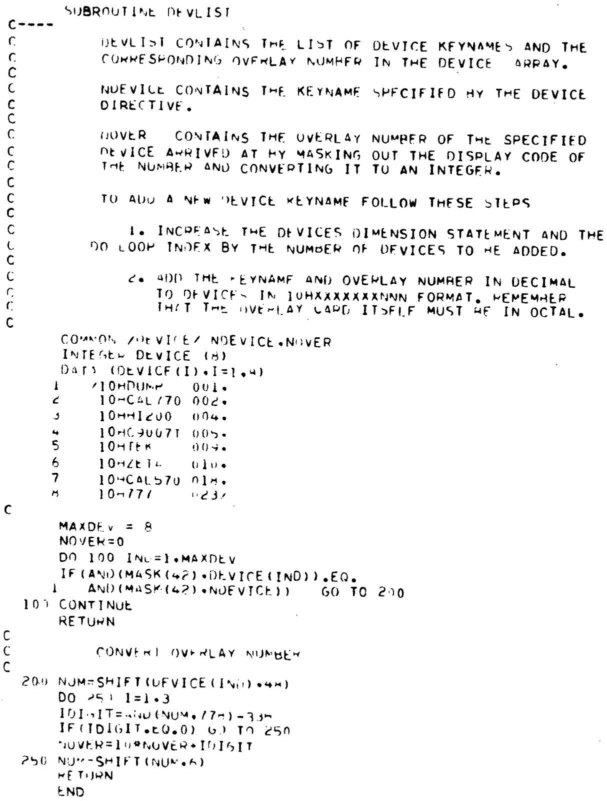
2. The name and decimal overlay number of the new plotting device must be added at the end of the list of device names in the DATA statement. This name is in the form of a Hollerith string and may be from one to seven characters long.

3. The value of variable MAXDEV is set by an assignment statement:

   MAXDEV=constant

   The constant must be incremented to conform to the number of devices supported and thus to the sizes of array DEVICE.

A listing of routine DEVLIST is given in figure II-13-1.



```
      SUBROUTINE DEVLIST
C----
C           DEVLIST CONTAINS THE LIST OF DEVICE KEYNAMES AND THE
C           CORRESPONDING OVERLAY NUMBER IN THE DEVICE  ARRAY.
C
C           NDEVICE CONTAINS THE KEYNAME SPECIFIED BY THE DEVICE
C           DIRECTIVE.
C
C           NOVER   CONTAINS THE OVERLAY NUMBER OF THE SPECIFIED
C           DEVICE ARRIVED AT BY MASKING OUT THE DISPLAY CODE OF
C           THE NUMBER AND CONVERTING IT TO AN INTEGER.
C
C           TO ADD A NEW DEVICE KEYNAME FOLLOW THESE STEPS
C
C               1. INCREASE THE DEVICES DIMENSION STATEMENT AND THE
C          DO LOOP INDEX BY THE NUMBER OF DEVICES TO BE ADDED.
C
C               2. ADD THE KEYNAME AND OVERLAY NUMBER IN DECIMAL
C                  TO DEVICES IN 10HXXXXXXXNNN FORMAT. REMEMBER
C                  THAT THE OVERLAY CARD ITSELF MUST BE IN OCTAL.
C
      COMMON /DEVICE/ NDEVICE,NOVER
      INTEGER DEVICE (8)
      DATA (DEVICE(I),I=1,8)
     1   /10HDUMP    001,
     2    10HCAL770 002,
     3    10HHI200   004,
     4    10HC9007T 005,
     5    10HTEK     009,
     6    10HZETA    010,
     7    10HCAL570 018,
     8    10H777     023/
C
      MAXDEV = 8
      NOVER=0
      DO 100 IND=1,MAXDEV
      IF(AND(MASK(42),DEVICE(IND)).EQ.
     1   AND(MASK(42),NDEVICE))    GO TO 200
  100 CONTINUE
      RETURN
C
C          CONVERT OVERLAY NUMBER
C
  200 NUM=SHIFT(DEVICE(IND),48)
      DO 250 I=1,3
      IDIGIT=AND(NUM,77B)-33B
      IF(IDIGIT.EQ.0) GO TO 250
      NOVER=10*NOVER+IDIGIT
  250 NUM=SHIFT(NUM,6)
      RETURN
      END
```

Figure II-13-1. Subroutine DEVLIST

WRITING A NEW OVERLAY MAIN PROGRAM

The design requirements on an overlay main program are as follows:

1. That it call device dependent routines in order to plot the graphic data specified in common /PPREQ/. (Details follow in 2 and 3.)

2. That upon being entered with an initialization request, it enters into /PPREQ/ a list of properties and parameters of the device that the overlay supports before the first call to NPFDATA.

3. That the overlay make calls to a main overlay subroutine NPFDATA to obtain data from the neutral picture file. The NPFDATA subroutine will set the /PPREQ/ common block and return to the overlay. Thus, the overlay will loop, calling NPFDATA until processing is complete.

The name of the overlay must be UNIPOST and the number must be in octal and correspond to the decimal number of the name of the device in table DEVICE in subroutine DEVLIST.

If the device initialization routine specifies a fixed file name on which plot commands are to be written, the file should be 5LPLOTF. If the device routines open the file specified in the PLOTS call, IPRM(2) should be used on the IREQ=0 request.

INTERFACE WITH THE DEVICE ROUTINES

The manufacturer-supplied, device-dependent subroutines for each different plotting device are loaded in different overlays. Each overlay has a custom-written main program that calls a working routine in the main (0,0) overlay to pick up request codes and parameters from the neutral picture file. Upon return to the primary overlay, these codes and respective parameters are in the PPREQ common block. The codes direct the primary overlay's main program to call a specific device dependent routine with the parameters supplied.

The PPREQ common block is defined as follows:

```
COMMON/PPREQ/IREQ,IPRM(31)
DIMENSION FPRM(31)
EQUIVALENCE(IPRM,FPRM)
```

IREQ is an integer the value of which specifies the kind of request being made. IPRM is an array that contains integer parameters to be used in carrying out the request. FPRM is an array that contains floating point parameters to be used in carrying out the request. The requests are listing in the following paragraphs.

INITIALIZATION REQUEST (IREQ = 0)

The overlay is requested to initialize itself and to return to the main program certain information about the display device serviced by the overlay. It is usually a requirement of the device specific subroutines that they be initialized only once in a run unless PLOT(X,Y,999) has been called.

Upon entry into the overlay with IREQ=0, the IPRM array is set as follows:

IPRM(1) = device keyname, and
IPRM(2) = logical unit number or name of the device dependent plot file, default of PLOTF.

The primary overlay main program then sets the values in PPREQ as follows:

Code	Definition
IPRM(1)	Device type where: 1 Drum plotter, 2 Table plotter, 3 Storage tube display, and 4 Microfilm device.
FPRM(2)	Display surface extent in X direction in inches.
FPRM(3)	Display surface extent in Y direction in inches.
IPRM(4)	Number of pens.
IPRM(5)	Is there hardware character generation? If no, code 0; If yes, code 1.
IPRM(6)	Are there hardware generated dotted lines? If no, code 0; If yes, code 1.
IPRM(7)	Are there hardware generated dashed lines? If no, code 0; If yes, code 1.
IPRM(8)	Is there hardware arc generation? If no, code 0; If yes, code 1.
IPRM(9)	Is there hardware cubic generation? If no, code 0; If yes, code 1.
IPRM(10)	Number of increments per inch (that is, 100, 200, 400,...).
IPRM(11)	Normal mode of operation where: 0 Batch processing, and 1 Interactive processing.

The primary overlay then calls the working routine, NPFDATA, in the main overlay which transfers the above information into the appropriate variables and transfers control to the main processor.

The main processing unit of the main overlay fills the PPREQ common area and returns to the working routine which returns to the primary overlay.

Upon return to the primary overlay, PPREQ contains one of the following:

VECTOR REQUEST (IREQ = 1)

The vector request provides the arguments necessary for a call to the CalComp standard subroutine PLOT or its equivalent. A value of 999 in IPRM(3) indicates that the current request is the final request in the run, and buffers should be purged, files closed, etc., as required and the program ended. A dashed or dotted vector (IPRM(3) = 4 to 5) is requested only if the presence of those features was indicated in the return from the initialization request.

Code	Definition
FPRM(1)	X - coordinate.
FPRM(2)	Y - coordinate.
IPRM(3)	Vector type where: 2 Solid line, 3 Invisible, 4 Dashed line (only if hardware dashed lines are available), 5 Dotted line (only if hardware dotted lines are available), -3 New picture, and 999 End of run.

NEWPEN REQUEST (IREQ = 2)

A call to the CalComp standard subroutine NEWPEN or the equivalent is requested.

IPRM(1) = pen number

This request may be made for a single-pen plotter if the application creating the neutral picture file calls NEWPEN with a negative pen number. If IPRM(1) is negative on a single pen plotter the following calls should be made:

```
CALL PLOT(0.0,0.0,999)
CALL PLOTS(IBUF,ISIZE,IUNIT)
```

ALPHA STRING REQUEST (IREQ = 3)

This request provides the overlay with the parameters needed for a character string call to the CalComp standard subroutine SYMBOL. Up to 100 characters may be present in each request. The X and Y coordinates usually have values of 999.0, indicating that the string is to be started at the current pen position. Centered symbols are not included in this type of request.

Code	Definition
FPRM(1)	X - coordinate of start of character string.
FPRM(2)	Y - coordinate of start of character string.
FPRM(3)	Character height.
FPRM(4)	Angle of baseline of character string in degrees.
IPRM(5)	Number of characters.
IPRM(6) through IPRM(15)	BCD character string.

DEVICE MODE REQUEST (IREQ = 4)

Arguments to device independent subroutines are passed unchanged through the neutral plot file. If the user of the post-processor selects the proper display device, a plotter mode request will be made to the overlay. The overlay should inspect IPRM(1) which will contain the name of the device specific routine in left-justified A10 format, and then call the specified routine using the subsequent words of IPRM as the arguments (use FPRM if an argument is of type REAL). The words of the array are used in ascending order from left to right in the subroutine argument list. (See UNIPLOT Reference/User's Guide for description of Device Mode Subroutine.)

Code	Definition
IPRM(1)	Name of subroutine. A10 format, left justified.
IPRM(2) through IPRM(13)	Subroutine arguments.
FPRM(2) through FPRM(13)	Subroutine arguments.

REORIGIN REQUEST (IREQ = 5)

This request is part of the overplot feature and is intended to be used by cathode-ray tube (CRT) displays only. Whenever a new drawing (reorigin) occurs in the neutral plot file, the overlay for a CRT display must decide whether or not to erase the current contents of the display. This decision is made by testing the overplot flag.

Code	Definition
IPRM(1)	Overplot flag where: 0 Do not overplot: erase, and 1 Overplot: do not erase.

INCREMENT REQUEST (IREQ = 6)

This request passes the value of the plotter increment size, as input in an INCREMENT statement to the overlay. The overlay may, on the basis of this value, call FACTOR to adjust the size of the finished plot. If no INCREMENT statement has been processed, the value of the plotter increment will be the default increment as set by the device dependent routines. If IREQ=6, the variable containing the increments per inch should be modified accordingly for any future initialization.

```
FPRM(1) = plotter increment size (i.e., 01, .005,
          or .0025)
```

PEN RESTORE REQUEST (IREQ = 7)

This request instructs the primary overlay to call the device dependent PLOT routine to move the pen to the bottom of the plotter as follows:

```
CALL PLOT(0.0,-30.0,-3)
```

After the primary overlay is built it can be added to the PL71 tape and a new absolute file cataloged by running the following EXTRACT job.

```
EXTRACT, {MT1} .
         {NT1}
ACCOUNT.              (Supply valid ACCOUNT card.)
LABEL(PL71,R,L=UNIPLOTV2,D= {HY} ,
   VSN=PL71)                {HD}
COPYBF(PL71,OLDPL)
REWIND(OLDPL)
UPDATE(Q,*==,C,D,8)
ROUTE(COMPILE,DC=IN)
7/8/9
=D ADD.XX
ACCOUNT.              (Change ACCOUNT card if
                      appropriate.)
=D ADD.XX             (Delete the comment card and
*IDENT DEV022         insert the updates to DEVLIST
*D DEVLIST.XX         and the new overlay deck. See
                      the listing of DEVLIST from
                      the installation run for sequence
                      numbers.)
         INTEGER DEVICE(15)
            .
            .
            .
*AF
*DECK OVL022          (Insert interface program and
                      device dependent routines here.)
6/7/8/9
```

If it is not desired to update PL71, a new post-processor can be created from a user library device dependent routines by running the following job:

```
ADDEV, {MT1} .
       {NT1}
ACCOUNT.
LABEL,PL71,R,L=UNIPLOTV2,D= {HY} ,
   VSN=PL71.                {HD}
SKIPF,PL71,3,17,B.
COPYBF,PL71,OLDPL.
UNLOAD(PL71)
REWIND(OLDPL)
UPDATE(Q)
ATTACH,UNI2,POSTLGO,ID=UNIPLOT.
ATTACH,LIB1,DEVICE ROUTINES,ID=xxx.
                      (Attach the device dependent
                      routines.)
FTN,I.
FTN,B=OVL026.
REWIND,LGO,OVL026.
COPYL,UNI2,LGO,UNI2A.
REQUEST,UNIPOST,*PF.
FILE(PLOTF,RT=S,BT=C)
LDSET(FILES=PLOTF)
LOAD(UNI2A)
LOAD(OVL026)
SATISFY(LIB1)
NOGO.
CATALOG,UNIPOST,ID=UNIPLOT,PW=1.
7/8/9
*IDENT DEVxxx
                      (Insert DEVLIST updates.)
*COMPILE DEVLIST
7/8/9
                      (Insert primary overlay main
                      program.)
6/7/8/9
```

VERIFICATION PROCEDURE

The program, VERIFY, can be run to verify proper installation of UNIPLOT as follows:

```
VRFY, {MT1} .
      {NT1}
ACCOUNT.
LABEL(PL71,R,L=UNIPLOTV2,D= {HY} ,
   VSN=PL71)                {HD}
SKIPF(PL74,4,17,B)
COPYBF(PL71,OLDPL)
UNLOAD(PL71)
REWIND(OLDPL)
UPDATE(F,*==,C,D,8)
ROUTE(COMPILE,DC-IN)
7/8/9
=IDENT xx
=D VERIFY.xx          (To change job and account
ACCOUNT.              card as needed.)
=D VERIFY.xx          (To specify the device.)
UNIPOST,D=device.
6/7/8/9
```

The run initiated by VRFY will create a 7-track tape ready for plotting on the specified device.

USAGE ACCOUNTING UTILITY 14

RELEASE DESCRIPTION

The Usage Accounting Utility release tape (AP001B) for NOS/BE 1 consists of the following three files:

File 1 - the installation deck PL containing the UAUINS job which installs the software necessary to provide accounting for usage priced software.

File 2 - RANDR (Update/data reduction/report generation program) - this program reads user prepared input directive cards and performs the functions specified by the directives. The functions, all relating to the product file and the usage detail file are:

1. Maintain product file data base.
2. Reduce data on the usage detail file and produce detail usage summary/billing reports as per directives.
3. Generate system status reports.

File 3 - ACOUNTX - (Usage pricing accounting module) - this set of subroutines provides the accounting interface to usage priced applications. ACOUNTX tracks application usage and logs that usage on the usage detail file.

RANDR and ACOUNTX are to be installed on the SYSTEM library NUCLEUS and SYSLIB respectively.

HARDWARE REQUIREMENTS

RANDR can be maintained on the same minimum hardware configuration as NOS/BE 1. The amount of core over and above that required by the calling application depends upon whether or not the calling application uses the CYBER record manager (CRM), the common memory manager (CMM), and/or is written in FORTRAN Extended Version 4 (FTN-4) (see Table II-14-1).

TABLE II-14-1. ADDITIONAL CORE REQUIREMENTS

Application Characteristics			Maximum Additional Core Required for ACOUNTX
CRM	FTN-4	CMM	(In Octal)
YES	YES	YES	2.5K
YES	YES	NO	3.1K
YES	NO	YES	3.6K
YES	NO	NO	4.2K
NO	YES	YES	12.5K
NO	YES	NO	13.1K
NO	NO	YES	13.6K
NO	NO	NO	14.2K

CORRECTIONS

There are no applicable PSR summaries.

DEFICIENCIES

It is imperative that at least one billing run be made in the month of January to prevent corruption of the year-to-date totals.

Notes and Cautions

When loading an application calling ACOUNTX, nonfatal load errors may occur due to duplicate system entry points. Other errors should be investigated.

INSTALLATION PARAMETERS

Upon installation (creation) of the two necessary files, the product file and the usage detail file, various protection parameters may be specified on the input card (ID=1) specifying UPDATE (see Usage Accounting Utility Reference Manual).

Column	Contents
1 to 4	CPU model code; this is a four-character code associated with the machine type and will not be changed after the files are established. (The codes are found in appendix C of the Usage Accounting Utility Reference Manual.) The customer, in the event of a CPU model upgrade (for example, a CYBER 172 with a 10316-1 upgrade is a CYBER 173), is obligated to perform a final billing run on the existing product and detail files, remove those files, and recreate them with the appropriate CPU model code.
5	Blank, reserved.
6	Threshold protection indicator; Y specifies that special usage accumulator entries will be built, updated, and maintained in the usage detail file; accumulators, thresholds and product status (active or inactive) are checked and/or updated at start and end of an application. N specifies no special threshold checking is to be done. Blank defaults to N.

Column	Contents
	This parameter may be changed at any time. However, altering it from Y to N and vice versa between billing runs corrupts the effectiveness of threshold protection. The special usage accumulator entries contain a usage accumulator. During a billing run, which covers a period of at least one full month, the accumulator is decremented by the invoiced amount of usage. The special usage accumulator entries are lost when a billing run is made with the threshold protection installed as N. They can be reestablished (initialized) via an UPDATE run with threshold protection set to Y.

NOTE

> Threshold protection is merely to assist the site in limiting their usage and is not intended in any way to limit the sites' liability for usage as reported in monthly billing runs.

Column	Contents
7 to 9	Blank,reserved.
10 to 14	Right-justified number (blanks treated as zero). It is multiplied by 1000 and represents the size (in number of words) at which the operator is warned that the detail file size protect has been reached. No other action is performed. It is the site responsibility to make a billing run (see <u>Usage Accounting Utility Reference Manual)</u> to reduce and report the usage detail file. This parameter may be changed at any time.
15	Blank, reserved.
16	User number field protection; Y specifies that the user number will be blanked on all detail usage reports. N specifies that user numbers for each accounting job step will appear on the detail usage reports. Blank defaults to N. This parameter may be changed at any time.

The following information represents directives to RANDR and is not considered installation parameters.

Column	Contents
17	Blank, reserved.
18	1; represents parameter type card.
19	1 to 9; represents number of copies of report.
20 to 25	UPDATE
26 to 80	Blank, reserved.

NOTE

> After installation creation, any of the above installation parameters may be changed except the CPU model code without removing and recreating files.

Also upon installation (creation) of the two necessary files (product and usage detail), a Control Data regional address must be installed. The Control Data address to be used is specified on the front of the software contract. In addition, the customer is obligated to supply their address and the name of an individual as a primary contact for resolution of any matters which may arise pertaining to the preparation, content and submission of the usage reports. The address information is specified on the input cards (ID 2,3) to the creation run and may be altered in part or whole by update runs.

For each specific usage priced product, there is additional installation information to be specified on the product activity card (ID 4); vendor code, product name, transaction codes, threshold protection values, and software codes. The vendor code, product name, and software code for Control Data usage priced products are supplied with the installation information; see specific application product description in this manual. Threshold protection values are supplied by the site.

(For further description of the input card types, see the <u>Usage Accounting Utility Reference Manual.</u>)

INSTALLATION PROCEDURE

AP001N contains the usage accounting utility. The RANDR and ACOUNTX modules are SYSEDIT'ed into the running system.

The following job is used to retrieve the installation job UAUINS from AP001B. This deck is not available on the INSTALLDECK,ID=IDECKS files.

```
UAUID(CM50000,T100,MT01) FOR 7-TRACK.
   HD01 OR PE01 FOR 9-TRACK.
COMMENT.  REPLACE THIS CARD WITH LOCAL
   ACCOUNTING INFORMATION
COMMENT.  THIS JOB ROUTES A JOB TO INSTALL
COMMENT.  THE USAGE ACCOUNTING UTILITY
COMMENT.  INTO THE RUNNING SYSTEM
COMMENT.  **XX=HY 7-TRACK (800 bpi),
   HD/9-TRACK (800 cpi) OR PE/9-TRACK (1600 cpi)
REQUEST(COMPILE,*Q)
LABEL(UAUTIL,D=XX,R,VSN=AP001B,
   L=USAGEACCTNG1P0)
UPDATE(P=UAUTIL,*==)
ROUTE(COMPILE,DC=IN)
7/8/9 - EOR -
UPDATE DIRECTIVES
                    THE UPDATE DIRECTIVES VARY
                    DEPENDING UPON THE DENSITY
                    AND TRACK TYPE BEING USED.
                    =DEFINE MTHY - 7 TRACK,  800 bpi
                    =DEFINE NTHD - 9 TRACK,  800 cpi
                    =DEFINE NTPE - 9 TRACK,  1600 cpi
6/7/8/9 - EOI -
```

The usage accounting utility is installed on the running system from AP001B by running job UAUINS which is executed by the above job UAUID. Refer to the section on "Installation Job" for a listing of UAUINS.

INSTALLATION JOB

RANDR establishes the product file and the usage detail file. If usage accounting files do not exist under ACXLIB, the applications interface ACOUNTX will not run.

UAUINS is the job which installs the usage accounting utility binaries on the running system.

```
UAUINS,CM50000,T100,MT1.  FOR 7-TRACK.
   HD1 OR PE1 FOR 9-TRACK.
COMMENT.        (Replace this card with local ac-
COMMENT.        counting information.  This job will
COMMENT.        EDITLIB the usage accounting
COMMENT.        utility binaries into the running
COMMENT.        system.)
COMMENT.  **XX=HY/7-TRACK (800 bpi),
   HD/9-TRACK (800 cpi) OR PE/9-TRACK (1600 cpi)
LABEL(UAUTIL,D=XX,R,VSN=AP001B,
   L=USAGEACCTNG1P0)
REWIND(UAUTIL)
SKIPF(UAUTIL,1,17)
COPYBF(UAUTIL,RANDR,1,17)
COPYBF(UAUTIL,ACOUNTX,1,17)
REWIND(RANDR,ACCUNTX)
EDITLIB(SYSTEM)
7/8/9 - EOR -
READY(SYSTEM,OLD)
LIBRARY(NUCLEUS,OLD)
REPLACE(*,RANDR,AL=0)
SETFL(RANDR,70000)
SETAL(RANDR,1)
FINISH.
LIBRARY(SYSLIB,OLD)
REPLACE(*,ACOUNTX,AL=0)
FINISH.
LISTLIB(*,SYSTEM,NUCLEUS)
LISTLIB(*,SYSTEM,SYSLIB)
COMPLETE.
ENDRUN.
6/7/8/9 - EOI -
```

Once the RANDR and ACOUNTX modules are installed on the running system, a new deadstart tape of the running system should be created.

Prior to execution of any usage priced applications, the following job UAFINS must be run. This job establishes the product file, initializes the usage detail file, and creates the product file entries for the leased Control Data usage priced application(s).

```
UAFINS(CM74000,T100)
COMMENT.         (Replace this card with local account-
                 ing information.)
COMMENT.         (This job establishes the product file
COMMENT.         and initializes the usage detail file.)
RANDR.
7/8/9 - EOR -    (The data deck's contents vary de-
                 pending upon the application(s) being
                 installed.  For this initial run there
UPDATE           must be 1 (ID=1) parameter card,
DATA CARDS       3 (ID=3) CDC address cards and at
                 least 1 (ID=4) product card.  See
                 Usage Accounting Utility Reference
                 Manual.)
6/7/8/9 - EOI -
```

PRODUCT AND USAGE DETAIL FILE

The product file is maintained as a permanent file and is catalogued as follows:

```
pfn - AUAPTF
ID  - JJFWGR
CY  - 1
RP  - 999
TK  - SURPTK
RD  - SURPRD
EX  - SURPEX
MD  - SURPMD
CN  - SURPCN
```

The usage detail file is also maintained as a permanent file and is catalogued as follows:

```
pfn - AUADTF
ID  - JJFWGR
RP  - 999
TK  - SURPTK
RD  - SURPRD
EX  - SURPEX
MD  - SURPMD
CN  - SURPCN
```

Because of the importance of these files, every attempt has been made to ensure that they are not lost during a RANDR run due to some external job interruption.

During a RANDR run, two other permanent file names become associated with AUAPTF (PFNEW, PFTMP) and AUADTF (DFNEW, DFTMP). The passwords, etc., are the same. Completion of RANDR purges these working files; noncompletion of RANDR (e.g., parity error, system abort, unexpected operator intervention) may cause them to be left as part of the permanent file base.

As RANDR modifies product file entries, the file (AUAPTF) is being rewritten as PFNEW. DFNEW has multiple uses. PFTMP is the AUAPTF file renamed and represents the product file prior to the RANDR run. Likewise, DFTMP represents the AUADTF file prior to the RANDR run.

In the event of an external job interruption, the recommended recovery procedure is to purge all files except the PFTMP and DFTMP files, RENAME then as AUAPTF and AUADTF, respectively, and rerun RANDR. In the event that one or both of the xxTMP file(s) are not found, but the xxNEW file(s) are present no action is necessary other than to purge the xxNEW file(s) and rerun RANDR.

It is further recommended that the site maintain adequate back up for these files (AUAPTF, AUADTF) through their DUMPF utility.

INTRODUCTION

INSTALLATION, MODIFICATION AND USER EXECUTION

Part III presents information pertaining to the installation of applications software under SCOPE 2.1. However, users must also reference the SCOPE 2.1 Installation Handbook for appropriate systems information that governs installation and execution dependencies, mandatory installation sequences and the inclusion of modifications (see part I, section 3 and part II, section 1 for specific details).

Control Data assumes that the installation of applications software will be performed by an analyst having at least six months experience with a Control Data 7600 or CYBER 70 computer system. These products are intended to be installed and maintained only as described in this manual and the accompanying SCOPE 2.1 Installation Handbook. As a result Control Data is only responsible for the proper functioning of features or parameters described in these two publications.

RELEASE MATERIALS

Application	Tape	Chapter
APEX III Out-Of-Core System 1	A3CREL	2
APEX III Mixed Integer Programming Option (MIP)	MIPREL	2
APEX III Matrix Reduction Option (REDUCE)	REDREL	2
APEX III Parametrics Option	PARREL	2
APT IV 2	PL59	3
IMSL 6	IMSLREL	4
SIMSCRIPT I.5 3	SIMREL	5

RELATED DOCUMENTATION

The following manuals pertain to SCOPE 2.1 and to appropriate applications products described in part III.

Title	Publication No.
SCOPE 2.1 Installation Handbook	60426100
SCOPE 2.1 Reference Manual	60342600
APEX III Reference Manual	76070000
APT IV Internal Maintenance Specification	†
APT IV Reference Manual	17326900
IMSL Reference Manual, Vols. I - II	SMD150043†
SIMSCRIPT I.5 Reference Manual Version 3	97400200

† The APT IV Version 2.0 Internal Maintenance Specifications may only be ordered by title through Control Data Corporation, Software Manufacturing and Distribution, 4201 North Lexington Avenue, St. Paul, Minnesota 55112.

APEX-III PRODUCT OPTIONS 2

APEX-III consists of a number of product options, each of which is separately priced and ordered.

APEX-III OUT-OF-CORE SYSTEM 1

RELEASE DESCRIPTION

APEX-III Out-of-Core System 1 runs under the SCOPE 2.1 operating system. The FORTRAN Extended compiler is required for compiling and updating APEX-III. APEX-III Out-of-Core System 1 is designed to execute from permanent files.

Hardware Requirements

APEX-III Out-of-Core System 1 requires the same minimum hardware configuration as SCOPE 2.1. A minimum field length of 60K octal is required to compile and execute APEX-III.

Deficiencies

None.

RELEASE MATERIALS

APEX-III Out-of-Core System 1 resides on the tape known as A3CREL. A3CREL is a 7- or 9-track tape with the following characteristics: labeled, 800 bpi, odd parity, BT=I, RT=W, MBL=5120. OOCAPEXIIIV1P1 is the file identifier in the HDR1 label.

- Partition 1 - installation deck in UPDATE PL form,
- Partition 2 - empty,
- Partition 3 - empty,
- Partition 4 - source code in UPDATE PL form,
- Partition 5 - verification deck in UPDATE PL form, and
- Partition 6 - sample output from verification run.

Listings of the installation deck (APEXC1) and the verification deck (APEXCV) may be obtained from tape A3CREL by executing the following job:

```
           {NT1}
LIST,T20,  {   }  ,STGGG.
           {MT1}
ACCOUNT.
LABEL(OLDPL,L=OOCAPEXIIIV1P1)
                   {HD}
STAGE(OLDPL,PRE,   {  }  ,E,VSN=APXOOC)
                   {HY}
COPYBF(OLDPL,OLD)
UPDATE(F,*==,L=7,P=OLD)
SKIPF(OLDPL,3,17)
UPDATE(F,*==,L=7,R)
7/8/9
6/7/8/9
```

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURES

APEX-III Out-of-Core System 1 is installed by executing deck APEXC1, contained on the first file of tape A3CREL. This job deck reads the APEX-III system from A3CREL, compiles the system, and installs an executable file of the APEX-III system on a permanent file named APEXIII with ID=APEX.

This job deck also creates three permanent files which are needed when one uses the APEX-III USER feature. (Reference the APEX-III Reference Manual for detailed information.) For those installations where the USER feature is used infrequently, it may be desirable to maintain these three files on magnetic tape rather than permanent files. These three files are described below:

Name	ID	Contents
APEXPL	APEX	Partial APEX-III source code. Contains APEX-III COMMON regions and subroutines QBUFFER, QGET, QUSER, and QUSERDA.
APEXOV	APEX	Relocatable binary code of APEX-III overlay drivers.
APEXRL	APEX	Relocatable binary code of remaining APEX-III subroutines.

For those installations which have purchased any of the APEX-III options, deck APEXC1 may also be used to install the APEX-III Out-of-Core System 1 with any combination of the APEX-III Mixed Integer Programming Option, APEX-III Matrix Reduction Option, and APEX-III Parametrics Option.

The deck APEXC1 contains UPDATE IF, DEF directives to simplify installation. The names used by these directives and their meaning when DEFINEd are as follows:

Name	Effect on Installation Job
NT	Uses 9-track magnetic tapes.
MT	Uses 7-track magnetic tapes.
MIP	APEX-III Mixed Integer Programming Option is installed with APEX-III Out-of-Core System 1.
RED	APEX-III Matrix Reduction Option is installed with APEX-III Out-of-Core System 1.
PAR	APEX-III Parametrics Option is installed with APEX-III Out-of-Core System 1.

Deck APEXC1 can be retrieved from tape A3CREL through the following procedure:

```
EXTR,T20, {NT1} ,STGGG.
          {MT1}
ACCOUNT.
LABEL(OLDPL, L=OOCAPEXIIIV1P1)
STAGE(OLDPL, PRE, {HD} , E, VSN=APXOOC)
                  {HY}
UPDATE(F,*==,C=PUNCH,D,8)
COMMENT.  **INSTALLATION DECKS ON PUNCH**
7/8/9
=DEFINE {NT}
        {MT}
=DEFINE MIP     (If Matrix Integer Programming
                Option.)
=DEFINE RED     (If Matrix Reduction Option.)
=DEFINE PAR     (If Parametrics Option.)
=D,APEXC1.2,APEXC1.3
INST,T2000, {NT1} ,STGGG (Supply valid JOB card.)
            {MT1}
ACCOUNT.        (Supply valid ACCOUNT card.)
=C,APEXC1
7/8/9
6/7/8/9
```

Error Messages

A nonfatal loader error will be encountered in loading the APEX-III system. The error message is of the form:

```
EXIT DUPLICATE ENTRY POINT IGNORED
```

Errors of any other form should be investigated.

INSTALLATION VERIFICATION

The verification deck, APEXCV, can be retrieved from tape A3CREL through the following procedure:

```
EXTV,T20, {NT1} ,STGGG.
          {MT1}
ACCOUNT.
LABEL(OLDPL, L=OOCAPEXIIIV1P1)
STAGE(OLDPL, PRE, {HD} , E, VSN=APXOOC)
                  {HY}
SKIPF(OLDPL,4,17)
UPDATE(F,*==,C=PUNCH,D,8,R)
COMMENT.  **VERIFICATION DECK ON PUNCH**
7/8/9
=D,APEXCV.2,APEXCV.3
VER,T50,STGGG.     (Supply valid JOB card.)
ACCOUNT.           (Supply valid ACCOUNT card.)
=C,APEXCV
7/8/9
6/7/8/9
```

Partition 6 of tape A3CREL can be copied to OUTPUT and used to validate correct execution of the verification run.

APEX-III MIXED INTEGER PROGRAMMING OPTION (MIP)

RELEASE DESCRIPTION

The APEX-III Mixed Integer Programming Option runs under the SCOPE 2.1 operating system when installed with APEX-III Out-of-Core System 1. The FORTRAN Extended compiler is required for compiling APEX-III MIP. APEX-III is designed to execute from permanent files.

Hardware Requirements

APEX-III MIP requires the same minimum hardware configuration as SCOPE 2.1. A minimum field length of 60K octal is required to compile and execute APEX-III MIP.

Deficiencies

None.

RELEASE MATERIALS

The APEX-III Mixed Integer Programming Option resides on the tape known as MIPREL. MIPREL is a 7- or 9-track tape with the following characteristics: labeled, 800 bpi, odd parity, BT=I, RT=W, MBL=5120. MIPAPEXIIIV1P1 is the file identifier in the HDR1 label.

Partition 1 - installation deck in UPDATE PL form,

Partition 2 - empty,

Partition 3 - empty,

Partition 4 - source code in UPDATE PL form,

Partition 5 - verification deck in UPDATE PL form, and

Partition 6 - sample output from verification run.

Listings of the installation deck (APEXC1) and the verification deck (APEXMV) may be obtained from tape MIPREL by executing the following job:

```
          {NT1}
LIST,T20, {MT1} ,STGGG.
ACCOUNT.
LABEL(OLDPL,L=MIPAPEXIIIV1P1)
                  {HD}
STAGE(OLDPL,PRE,  {HY}  ,E,VSN=APXMIP)
COPYBF(OLDPL,OLD)
UPDATE(F,*==,L=7,P=OLD)
SKIPF(OLDPL,3,17)
UPDATE(F,*==,L=7,R)
7/8/9
6/7/8/9
```

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURES

The APEX-III Mixed Integer Programming Option must be installed with APEX-III Out-of-Core System 1. The installation deck, APEXC1, on MIPREL is identical to the installation deck for APEX-III Out-of-Core System 1. The name MIP is defined for the UPDATE IF,DEF directives in deck APEXC1 to merge the MIP Option with the out-of-core system.

Reference "Installation Procedure" for APEX-III Out-of-Core System 1 for details of installation and expected error messages.

INSTALLATION VERIFICATION

The verification deck, APEXMV, can be retrieved from tape MIPREL through the following procedure:

```
          {NT1}
EXTV,T20, {MT1}  ,STGGG.
ACCOUNT.
LABEL(OLDPL,L=MIPAPEXIIIV1P1)
                  {HD}
STAGE(OLDPL,PRE,  {HY}  ,E,VSN=APXMIP)
SKIPF(OLDPL,4,17)
UPDATE(F,*==,C=PUNCH,D,8,R)
COMMENT.  **VERIFICATION DECK ON PUNCH**
7/8/9
=D,APEXMV.2,APEXMV.3
VER,T50,STGGG.       (Supply valid job card.)
ACCOUNT.             (Supply valid ACCOUNT card.)
=C,APEXMV
7/8/9
6/7/8/9
```

Partition 6 of tape MIPREL can be copied to OUTPUT and used to validate correct execution of the verification run.

APEX-III MATRIX REDUCTION OPTION (REDUCE)

RELEASE DESCRIPTION

The APEX-III Matrix Reduction Option runs under the SCOPE 2.1 operating system when installed with APEX-III Out-of-Core System 1. The FORTRAN Extended compiler is required for compiling and updating APEX-III REDUCE. APEX-III REDUCE is designed to execute from permanent files.

Hardware Requirements

APEX-III REDUCE requires the same minimum hardware configuration as SCOPE 2.1. A minimum field length of 60K octal is required to compile and execute APEX-III.

Deficiencies

None.

RELEASE MATERIALS

The APEX-III Matrix Reduction Option resides on the tape known as REDREL. REDREL is a 7- or 9-track tape with the following characteristics: labeled, 800 bpi, odd parity, BT=I, RT=W, MBL=5120. REDAPEXIIIV1P1 is the file identifier in the HDR1 label.

Partition 1 - installation deck in UPDATE PL form,

Partition 2 - empty,

Partition 3 - empty,

Partition 4 - source code in UPDATE PL form,

Partition 5 - verification deck in UPDATE PL form, and

Partition 6 - sample output from verification run.

Listings of the installation deck (APEXC1) and the verification deck (APEXRV) may be obtained from tape REDREL by executing the following job:

```
LIST,T20, {NT1} ,STGGG.
          {MT1}
ACCOUNT.
LABEL(OLDPL,L=REDAPEXIIIV1P1)
STAGE(OLDPL,PRE, {HD} ,E,VSN=APXRED)
                 {HY}
COPYBF(OLDPL,OLD)
UPDATE(F,*==,L=7,P=OLD)
SKIPF(OLDPL,3,17)
UPDATE(F,*==,L=7,R)
7/8/9
6/7/8/9
```

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURES

The APEX-III Matrix Reduction Option must be installed with APEX-III Out-of-Core System 1. The installation deck, APEXC1, on REDREL is identical to the installation deck for APEX-III Out-of-Core System 1. The name RED is defined for the UPDATE IF,DEF directives in deck APEXC1 to merge the REDUCE Option with the Out-of-Core System.

(Reference "Installation Procedure" for APEX-III Out-of-Core System 1 for details of installation and expected error messages.)

Installation Verification

The verification deck, APEXRV, can be retrieved from tape REDREL through the following procedure:

```
EXTV,T20, {NT1} ,STGGG.
          {MT1}
ACCOUNT.
LABEL(OLDPL,L=REDAPEXIIIV1P1)
STAGE(OLDPL,PRE, {HD} ,E,VSN=APXRED)
                 {HY}
SKIPF(OLDPL,4,17)
UPDATE(F,*==,C=PUNCH,D,8,R)
COMMENT.  **VERIFICATION DECK ON PUNCH**
7/8/9
=D,APEXRV.2,APEXRV.3
VER,T50,STGGG.        (Supply valid JOB card.)
ACCOUNT.              (Supply valid ACCOUNT card.)
=C,APEXRV
7/8/9
6/7/8/9
```

Partition 6 of tape REDREL can be copied to OUTPUT and used to validate correct execution of the verification run.

APEX-III PARAMETRICS OPTION

RELEASE DESCRIPTION

The APEX-III Parametrics Option runs under the SCOPE 2.1 operating system when installed with APEX-III Out-of-Core System 1. The FORTRAN Extended compiler is required for compiling and updating APEX-III Parametrics. APEX-III Parametrics is designed to execute from permanent files.

Hardware Requirements

APEX-III Parametrics requires the same minimum hardware configuration as SCOPE 2.1. A minimum field length of 60K octal is required to compile and execute APEX-III.

Deficiencies

None.

RELEASE MATERIALS

The APEX-III Parametrics Option resides on the tape known as PARREL. PARREL is a 7- or 9-track tape with the following characteristics: labeled, 800 bpi, odd parity, BT=I, RT=W, MBL=5120. PARAPEXIIIV1P1 is the file identifier in the HDR1 label.

Partition 1 - installation deck in UPDATE PL form,

Partition 2 - empty,

Partition 3 - empty,

Partition 4 - source code in UPDATE PL form,

Partition 5 - verification deck in UPDATE PL form, and

Partition 6 - sample output from verification run.

Listings of the installation deck (APEXC1) and the verification deck (APEXPV) may be obtained from tape PARREL by executing the following job:

```
           (NT1)
LIST,T20,  {   }  ,STGGG.
           (MT1)
ACCOUNT.
LABEL(OLDPL,L=PARAPEXIIIV1P1)
                   (HD)
STAGE(OLDPL,PRE,   {  }  ,E,VSN=APXPAR)
                   (HY)
COPYBF(OLDPL,OLD)
UPDATE(F,*==,L=7,P=OLD)
SKIPF(OLDPL,3,17)
UPDATE(F,*==,L=7,R)
7/8/9
6/7/8/9
```

INSTALLATION PARAMETERS

None.

INSTALLATION PROCEDURES

The APEX-III Parametrics Option must be installed with APEX-III Out-of-Core System 1. The installation deck, APEXC1, on PARREL is identical to the installation deck for APEX-III Out-of-Core System 1. The name PAR is defined for the UPDATE IF,DEF directives in deck APEXC1 to merge the Parametrics Option with the Out-of-Core System.

(Reference "Installation Procedure" for APEX-III Out-of-Core System 1 for details of installation and expected error messages.)

INSTALLATION VERIFICATION

The verification deck, APEXPV, can be retrieved from tape PARREL through the following procedure:

```
           (NT1)
EXTV,T20,  {   }  ,STGGG.
           (MT1)
ACCOUNT.
LABEL(OLDPL,L=PARAPEXIIIV1P1)
                   (HD)
STAGE(OLDPL,PRE,   {  }  ,E,VSN=APXPAR)
                   (HY)
SKIPF(OLDPL,4,17)
UPDATE(F,*==,C=PUNCH,D,8,R)
COMMENT.   **VERIFICATION DECK ON PUNCH**
7/8/9
=D,APEXPV.2,APEXPV.3
VER,T50,STGGG.           (Supply valid JOB card.)
ACCOUNT.                 (Supply valid ACCOUNT card.)
=C,APEXPV
7/8/9
6/7/8/9
```

Partition 6 of tape PARREL can be copied to OUTPUT and used to validate correct execution of the verification run.

GENERAL DESCRIPTION

APT IV is a numerical control language processor designed to generate cutter location coordinates. The user has the capability to fully describe the part to be machined and then drive the tool along the chosen path. The output from APT IV is a verification listing and a file for input to a post processor. APT IV V2 has all the capabilities of APT IV V1 in addition to new language features and extensions.

APT IV V2 runs under the SCOPE 2.1 operating system on a Control Data host computer having at least 65,000 words of central memory.

RELEASE MATERIALS

APT IV V2 for SCOPE 2.1 is contained on release tape PL59. The structure of PL59 is as follows:

File 1 - contains the source library in UPDATE format. The first 49 decks, TAPEOP through ZDDPLIN are common decks called by one or more of the succeeding decks. The next 16 decks, LLDM through LBCDBIN, are the LOAD COMPLEX. The next deck, LLCDATA, is data used by the LOAD COMPLEX routines. The next 92 decks, TAPT4T through TLCANON, are the TRANSLATOR. The next 333 decks, XAPT4E through ZZDDPICT are the EXECUTION COMPLEX. The next 45 decks, CAPT4C through CTAPEWT are the CLEDITOR. The last 18 decks, PAPT4P through PTAPEWT are the POST EXEC.

File 2 - contains the relocatable binary records of all routines needed for the STANDARD mode. It includes the binary records, COD4, PRD4, PTR4, POO4, and COO4, produced by the LOAD COMPLEX. This file is ready to be loaded by the host computer system and will produce the absolute modules contained on file 3.

File 3 - contains the absolute modules of the APT IV V2 system. The file should be placed on the system library if the user does not wish to modify the APT IV V2 processor. File 2 of the release tape is used to make file 3.

File 4 - contains the decks to install APT IV V2 on the host computer system.

File 5 - contains three identical part programs. Job control cards are provided to verify correct operation with system overlays resident in the system library, permanent files, or user library.

INSTALLATION REQUIREMENTS

The system has four general modes of operation:

1. Standard mode - offers all capabilities except parametric and sculptured surfaces. This mode requires approximately $110,000_8$ words of field length to build the system and for execution;
2. Parametric surface mode - offers all capabilities of the standard system plus the added features of parametric surface processing. This mode does not include sculptured surface processing. Approximately $110,000_8$ words are required to build the system and ($110,000_8$ plus the length of the FORTRAN parametric surface routines) words for execution;
3. Sculptured surfaces mode - offers all capabilities of the standard system plus the added features of sculptured surface processing. This mode does not include parametric surface processing. Approximately $122,000_8$ words are required to build and execute the system;
4. Parametric and sculptured mode - both parametric and sculptured surface processing may be used. Approximately $122,000_8$ words are required to build the system and ($122,000_8$ plus the length of of the FORTRAN parametric surface routine) words for execution.

INSTALLATION OPTIONS

The user may choose one of four modes:

1. standard
2. parametric surface
3. sculptured surface
4. parametric and sculptured surface

The release system operates in Standard mode.

In addition to the above modes, the user may choose one of three residences for the APT IV system. These are:

1. system library
2. permanent files
3. user library

The released system operates from the system library.

INSTALLATION PROCEDURES

The following control cards may be used to obtain the verification and installation decks from PL59:

```
IBGET,MT1
LABEL(GET,R,L=APT4*RELEASE)
   where RELEASE = SCOPE2P1
SKIPF(GET,3,17)
COPYCF(GET,PUNCH,2)
6/7/8/9
```

The following provides a description of all the installation decks included in file 4. The job name (MAPT41 - MAPT46) identifies the deck.

- Install APT IV from file 3 of the release tape.

  Deck MAPT41 installs APT IV into the host computer operating system from file 3 of the release tape. The first two files are skipped and the absolute modules of file 3 are copied to input files for the system library program.

- Install APT IV from file 2 of the release tape.

  Deck MAPT42 may be used to make modifications to any routines except the LOAD COMPLEX. The modified routines are exchanged with their counterparts on the supplied relocatable file, and the resultant file is installed on the host computer system.

- Install APT IV from file 1 of the release tape.

  Deck MAPT43 and MAPT44 may be used to build the entire APT IV system from the source file. This will be necessary if the user wishes to install the sculptured surface and/or parametric surface mode of operation. The LOAD COMPLEX routines are compiled and executed. The output from LOAD COMPLEX is five FORTRAN block data programs which are then compiled and the relocatable binaries are placed on the file LCBIN. The entire APT IV system (TAPT4T through PTAPEWT) is compiled and relocatable binaries are placed on APTBIN. The two files are merged to form the file APTLGO. This file is then input to the LOADER and the absolute overlay modules are placed in the overlay file APTOVL. The remainder of the control cards makes a new system release tape which may be used as input to deck MAPT41.

INSTALLATION OPTIONS

The user has the following two options:

1. The user must insert an UPDATE directive card *DEFINE SSSURF to cause the sculptured surface routines to be compiled into the system. This card is present in deck MAPT43.
2. The user must insert an UPDATE directive card *DEFINE PARSURF to cause the parametric surface routines to be compiled into the system. This card is present in deck MAPT44.

Install APT IV from file 2 with overlay residence on permanent files.

Deck MAPT45 may be used to modify the relocatable file so that the APT IV system overlays can reside on permanent files. This deck can be modified as noted with COMMENT cards for a relocatable file with or without sculptured surfaces defined. For purposes of this example, the standard and parametric modes are equivalent.

The permanent files are defined, UPDATE is run with the overlay load parameter in deck TSUPERVT set equal to 0, the routine BEGINT is compiled and replaced on the relocatable file. Each overlay deck is compiled so that the overlay will be placed on a separate file. The loader then produces absolute overlay modules on the permanent files which are saved for later use.

Install APT IV from file 2 with overlay residence on a user library.

Deck MAPT46 may be used to modify the relocatable file so that the APT IV system overlays can reside on a user library. The deck can be modified as noted with COMMENT cards for a relocatable file with or without sculptured surfaces defined. For purposes of this discussion, the standard and parametric modes are equivalent.

UPDATE is run with the overlay load parameter in deck TSUPERVT set equal to 7LAPT4OVL. The routine BEGINT is compiled and replaced on the relocatable file. The loader then produces absolute overlay modules on the file AFTOVL which is used as input to the EDITLIB program. EDITLIB then creates a user library on file APT4OVL.

Install APT IV from file 2 with overlay residence on the system library.

Deck MAPT42 may be used to install the system overlays since the release file has the overlay load parameter in deck TSUPERVT set equal to 6LSYSTEM.

LOAD COMPLEX EXECUTION

Deck MAPT47 may be used to modify the five relocatable decks produced by the LOAD COMPLEX. The names of the decks are COD4, PRD4, PTR4, POO4, and GOO4. Once the user modifications are determined and inserted, deck MAPT47 compiles and executes LOAD COMPLEX routines LLDM through LBCDBIN.

The deck LCCDATA, which has been modified by the user, is the input data for the LOAD COMPLEX execution. The output, five FORTRAN BLOCK DATA routines, is then compiled and the relocatable decks replace their counterparts on file 2 of the system release tape. Deck MAPT42 may then be used to install the system.

VERIFICATION

Deck MAPT48, MAPT49, or MAPT50 from file 5 should be used to verify that APT IV V2 has been correctly installed. Deck MAPT48 is used when the system overlays reside on the host computer system library. Deck MAPT49 is used when the system overlays reside on permanent files. Deck MAPT50 is used when the system overlays reside on a user library.

RELEASE DESCRIPTION

IMSL 6 (International Mathematical and Statistical Library, Version 6) consists of more than 400 FORTRAN functions and subprograms. IMSL 6 is on IMSLREL.

HARDWARE REQUIREMENTS

IMSL 6 requires the minimum hardware configuration for SCOPE 2.

NOTES AND CAUTIONS

The installation process does not provide program listings. If a program listing is needed (it is very long), the FTN card in the install deck may be appropriately modified (remove L=0).

RELEASE MATERIALS

IMSL 6 is contained on the tape known as IMSLREL. IMSLREL is a program library in UPDATE format. It has the following characteristics: 7-track (800 bpi) or 9-track (800 or 1600 cpi), binary recording mode, seven files:

File 1 - install deck in UPDATE OLDPL format,

File 2 - empty file,

File 3 - empty file,

File 4 - program deck in UPDATE OLDPL format,

File 5 - verify job control language deck in UPDATE OLDPL format,

File 6 - expected verify output ((COPYBF,OUTPUT) format), and

File 7 - verify program deck in UPDATE OLDPL format.

INSTALLATION PROCEDURES

The procedures for the installation of IMSL 6 are outlined in the following four steps:

Step 1. Generate Essential Listings

This step includes:

- Installation deck,
- Verification deck, and
- Expected verify output.

```
JOB.               }
ACCOUNT(*,*,*,*)   }  (Supplied by user.)
FILE(TAPE,RT=S)
LABEL(TAPE,R)
              (HY)
STAGE(TAPE,   {HD}  ,E,VSN=IMSLIB)
              (PE)
REWIND(TAPE)
UPDATE(P=TAPE,F,D,8,I=DUMMY,*=/,L=7)
REWIND(TAPE)
SKIPF(TAPE,4,17)
UPDATE(P=TAPE,F,D,8,I=DUMMY,*=/,R,L=7)
REWIND(TAPE)
SKIPF(TAPE,5,17)
COPYBF(TAPE,OUTPUT,1)
6/7/8/9
```

Step 2. Install IMSL 6

Installation characteristics are:

- Permanent file name = IMSLIB
- Public access (ID=PUBLIC)

```
JOB.               }
ACCOUNT(*,*,*,*)   }  (Supplied by user.)
FILE(TAPE,RT=S)
LABEL(TAPE,R)
              (HY)
STAGE(TAPE,   {HD}  ,E,VSN=IMSLIB)
              (PE)
REWIND(TAPE)
UPDATE(P=TAPE,F,D,8,*=/,C=PUNCH)
COMMENT. **INSTALLATION DECK ON PUNCH**
7/8/9
/IDENT IMSL           (Optional.)
/DEFINE SCOPE 2       (Required.)
```

Note that one of the following three tape parameters is required.

```
/DEFINE MT            (7-track, 800 bpi.)
/DEFINE NT            (9-track, 800 bpi.)
/DEFINE NTPE          (9-track, 1600 bpi.)

/DELETE IMSLIL.XX,YY        (Required.)
JOB.
ACCOUNT(*,*,*,*)
6/7/8/9
```

Step 3. Run Verification Tests

Refer to the following:

```
JOB.               )
ACCOUNT(*,*,*,*)   }  (Supplied by user.)
FILE(TAPE,RT=S)
LABEL(TAPE,R)
            ( HY )
STAGE(TAPE, { HD }  ,E,VSN=IMSLIB)
            ( PE )
REWIND(TAPE)
SKIPF(TAPE,4,17)
UPDATE(P=TAPE,F,D,8,*=/,R=C,C=PUNCH)
COMMENT.  **VERIFY DECK ON PUNCH**
7/8/9
/IDENT IMSL           (Optional.)
/DEFINE SCOPE 2       (Required.)
```

Note that one of the following three tape parameters is required.

```
/DEFINE MT          (7-track, 800 bpi.)
/DEFINE NT          (9-track, 800 bpi.)
/DEFINE NTPE        (9-track, 1600 bpi.)

/DELETE IMSLV1.XX,YY     (Required.)
JOB.               )
ACCOUNT(*,*,*,*)   }  (Supplied by user.)
6/7/8/9
```

Step 4. Verification

Do a minimal visual comparison of the expected verify output and the actual verify output.

SIMSCRIPT I.5 3

RELEASE DESCRIPTION

SIMSCRIPT I.5 runs under the SCOPE 2.1 operating system. In order to compile the SIMSCRIPT compiler, an absolute executable SIMSCRIPT compiler must be installed in the system. The FORTRAN library is necessary for execution of SIMSCRIPT programs.

HARDWARE REQUIREMENTS

SIMSCRIPT I.5 requires the same minimum hardware configuration as SCOPE 2.1.

DEFICIENCIES

None.

RELEASE MATERIALS

SIMSCRIPT I.5 resides on the tape known as SIMREL. SIMREL is a 7- or 9-track tape with the following characteristics: labeled, 800 bpi, odd parity, BT=I, RT=W, MBL=5120 and SIMREL*SCP2.1 as the file identifier in the HDR1 label.

Partition 1 - installation and verification decks in UPDATE PL form,

Partition 2 - source code of the SIMSCRIPT compiler and execution library in UPDATE PL form,

Partition 3 - absolute overlay of the SIMSCRIPT compiler, and

Partition 4 - relocatable binaries of the SIMSCRIPT execution library.

Listings of the installation decks (SIMINS and SIMGEN) and the verification deck (VSIMI5) may be obtained from tape SIMREL by executing the following job:

```
          {SM}
LIST,T20, {SN} ,STGGG.
ACCOUNT.
LABEL(OLDPL,L=$SIMREL*SCP2.1$)
                  {HY}
STAGE(OLDPL,PRE,  {HD} ,E,VSN=SIMREL)
UPDATE(F,L=7)
7/8/9
6/7/8/9
```

INSTALLATION PROCEDURES

SIMSCRIPT I.5 is installed by executing deck SIMINS or SIMGEN contained on the first file of tape SIMREL:

SIMINS   Catalogs binaries as SIMLEL (ID=SCPSCR) and SIMLIB7 (ID=PRDLIB).

SIMGEN   Regenerates SIMSCRIPT and catalogs SIMLEL (ID=SCPSCR) and SIMLIB7 (ID=PRDLIB).

The installation decks can be retrieved from tape SIMREL through the following procedure.

```
          {SM}
EXTR,T20, {SN} ,STGGG.
ACCOUNT.
LABEL(OLDPL,L=$SIMREL*SCP2.1$)
                  {HY}
STAGE(OLDPL,PRE,  {HD} ,E,VSN=SIMREL)
UPDATE(Q,C=PUNCH,D,8)
COMMENT.  **INSTALLATION DECKS ON PUNCH**
7/8/9
*DF X
*C Y
7/8/9
6/7/8/9
```

where X is HY or HD and Y is SIMINS or SIMGEN.

VERIFICATION PROCEDURE

The verification deck can be retrieved from tape SIMREL through the following procedure:

```
            {SM}
VERIFY,T20, {SN} ,STGGG.
ACCOUNT.
LABEL(OLDPL,L=$SIMREL*SCP2.1$)
                  {HY}
STAGE(OLDPL,PRE,  {HD} ,E,VSN=SIMREL)
UPDATE(Q,C=PUNCH,D,8)
COMMENT.  **VERIFY DECK ON PUNCH**
7/8/9
*C VSIMI5
7/8/9
6/7/8/9
```

SIMSCRIPT I.5 can be verified by executing the deck VSIMI5.

CONFIGURATION PARAMETERS

The following installation parameter is in common deck CDC$, which also contains the SIMSCRIPT assembly parameters:

Parameter	Released Default Value	Significance
≡LINES≡	55_{10}	Number of lines per page

NOTES AND CAUTIONS

The installation jobs, SIMINS and SIMGEN, do not purge currently existing SIMSCRIPT permanent files.

INDEX

COMMENT SHEET

MANUAL TITLE Control Data NOS 1, NOS/BE 1, SCOPE 2 Application Installation Handbook

PUBLICATION NO. 76071100 REVISION S

FROM: NAME: ________________

BUSINESS
ADDRESS: ________________

COMMENTS:

This form is not intended to be used as an order blank. Your evaluation of this manual will be welcomed by Control Data Corporation. Any errors, suggested additions or deletions, or general comments may be made below. Please include page number references and fill in publication revision level as shown by the last entry on the Record of Revision page at the front of the manual. Customer engineers are urged to use the TAR.

CUT ALONG LINE

NO POSTAGE STAMP NECESSARY IF MAILED IN U. S. A.

FOLD ON DOTTED LINES AND STAPLE

STAPLE STAPLE

FOLD FOLD

FIRST CLASS
PERMIT NO. 8241

MINNEAPOLIS, MINN.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION
Publications and Graphics Division
Data Services Publications
P.O. Box 0 HQC02C
Minneapolis, MN. 55440

CUT ALONG LINE

FOLD FOLD

CORPORATE HEADQUARTERS, P.O. BOX 0, MINNEAPOLIS, MINN. 55440
SALES OFFICES AND SERVICE CENTERS IN MAJOR CITIES THROUGHOUT THE WORLD

LITHO IN U.S.A.

CONTROL DATA CORPORATION