ORNL/TM-7353

ornl

OAK
RIDGE
NATIONAL
LABORATORY

UNION
CARBIDE

User and System Considerations for the TCSTEK Software Library: A Graphics Library for Displaying Images on Tektronix 4ØØØ Series Storage Tube Terminals

W. H. Gray

OPERATED BY
UNION CARBIDE CORPORATION
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road, Springfield, Virginia 22161
NTIS price codes—Printed Copy: A04; Microfiche A01

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

ORNL/TM-7353
Dist. Category UC-32

Contract No. W-74Ø5-eng-26

FUSION ENERGY DIVISION

USER AND SYSTEM CONSIDERATIONS FOR THE TCSTEK SOFTWARE LIBRARY: A GRAPHICS LIBRARY FOR DISPLAYING IMAGES ON TEKTRONIX 4ØØØ SERIES STORAGE TUBE TERMINALS

W. H. Gray

Date Published - November 1980

Prepared by the
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37830
operated by
UNION CARBIDE CORPORATION
for the
DEPARTMENT OF ENERGY

DISCLAIMER

This book was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

CONTENTS

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

PREFACE

This work was sponsored by the Oak Ridge National Laboratory Fusion Energy Division Large Coil Program and was done in conjunction with system development on the GIFTS 4B[1] suite of structural analysis computer programs.

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

ABSTRACT

This report documents the idiosyncrasies of the Tektronix PLOT 1Ø Terminal Control System (TCS) level 3.3 software as it currently exists on the Oak Ridge National Laboratory Fusion Energy Division DECsystem-1Ø computer. It pertains to the graphics software library called TCSTEK, which may be used to convert TCS subroutine calls into display images on the screens of Tektronix 4ØØØ series storage tube terminals. Fully described here are several user-oriented enhancements and system efficiency modifications to the vendor-supplied TCS software library. It is intended that this report serve as a reference for future TCS updates so that continuity between releases of TCS PLOT 1Ø software may be maintained.

1. INTRODUCTION

Described here is the implementation of the Tektronix PLOT 1Ø Terminal Control System (TCS)[2,3] software as it currently exists on the Oak Ridge National Laboratory (ORNL) Fusion Energy Division (FED) DECsystem-1Ø computer. Superseding all previous reports on this subject, this document specifically applies to the TCS level 3.3 software library called TCSTEK. The TCS level 3.3 software has been implemented without any user-discernible differences between it and the vendor-supplied installation instructions,[4] successfully passing the vendor test program.[5]

This software library can be used to produce display images on the screens of Tektronix 4ØØØ series storage tube terminals. Unlike several of the other graphics libraries available on the FED DECsystem-1Ø, the TCSTEK software is a computationally efficient set of subroutines requiring a small amount of the user's core image during execution.

Several enhancements have been made to the ORNL FED TCS software. They entail making the software library automatically determine the terminal's transmission baud rate and making the software efficiently pause during a page erasure and a hard-copy operation. This report discusses these enhancements. Also presented is the locally written machine-dependent software required by TCS for the translation of ASCII text strings into ASCII decimal equivalent (ADE) characters and for the input and output (I/O) of arrays of ADE characters.

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

2. HOW TO USE THIS DOCUMENT

If the reader of this report is interested in the system implementation of the TCS subroutines, then all sections should be read. For the average reader, however, only four sections are required reading. They are (1) Sect. 3, where accessing the TCSTEK software library is discussed; (2) Sect. 4, where an example program demonstrates how to use this software library; (3) Sect. 5, where local enhancements to several TCS subroutines are discussed; and (4) Sect. 7, where the TCS I/O software is discussed.

This report specifically pertains to the software library on REL: called TCSTEK.REL and is organized into the following sections.

(1) In Sect. 3 the reader may find out how to access the TCSTEK software library.

(2) Section 4 presents an example program in order to demonstrate the correct way to initialize and finalize the TCSTEK software.

(3) Discussed in Sect. 5 are several of the system considerations that prompted the modification of the TCS software library and therefore the writing of this document. Subroutines discussed are INITT, FINITT, HDCOPY, ERASE, NEWPAG, AOUTST, and TCSLEV.

(4) Section 6 presents the locally written machine-dependent character-manipulating software required by TCS. Subroutines discussed are KA12AS, KAM2AS, KAS2A1, and KAS2AM.

(5) Section 7 presents the locally written machine-dependent terminal I/O software required by TCS and specifically enumerates its advantages and disadvantages. Subroutines discussed are ADEIN and ADEOUT.

(6) In Sect. 8 are discussed the local enhancements to the TCS software — in particular the program that sets the transmission speed code for all teletypes connected to the FED DC1Ø communications interface and the corresponding subroutine modifications in the TCS source program necessary to access the transmission speed code.

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

3. ACCESSING THE TCSTEK SOFTWARE LIBRARY

The TCSTEK software library is available to the users of the FED DECsystem-1∅ by using a command file located on device PUB: in a manner similar to that used to access all other graphics software. Such .CMD files cause the inclusion of all the necessary relocatable object modules (.REL files) without the user's having to concern himself with the location of the .REL files or the order in which they are loaded.

To execute a FORTRAN program, 'yourpg' in this example, and to use the TCSTEK software library, type

```
.EXECUTE yourpg,@PUB:TCSTEK
```

Two inflexible requirements must be met by the user's FORTRAN code:

(1) There must be a call to subroutine INITT(∅). This is the standard way to initialize the TCS software. (A call to subroutine INITT with an argument of zero forces TCS to determine automatically the TTY transmission speed in characters per second.) This subroutine call should be made only once and must be the first TCS subroutine call in the user's program.

(2) There must be a call to subroutine FINITT(IX,IY). This is the standard way to terminate the TCS software. (The arguments IX and IY represent the final position of the alphanumeric cursor in absolute screen coordinates.) This subroutine call should be made only once and must be the last TCS subroutine call in the user's program so that the last internal TCS character buffer is written to the Tektronix screen.

The command file PUB:TCSTEK.CMD loads the relocatable object module REL:TCSTEK.REL with 'yourpg' using the /SEARCH switch. This is the only .REL file necessary for this software library.

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

4. A SIMPLE EXAMPLE

This section demonstrates an example execution of a FORTRAN program to draw a display image on a Tektronix. The following program uses TCS to draw an exponentially decaying cosine function. This program is internally commented and should be sufficiently self-explanatory. A listing follows.

```
      PROGRAM RNGTCS
C
      COMMON /DRWPAR/ ISW(4),RVW(4),CON
C
      DATA ISW/2ØØ,4ØØ,2ØØ,4ØØ/
      DATA RVW/Ø.,2Ø.,-1.,1./
      DATA CON/Ø.2/
C
C  INITIALIZE THE TCS SOFTWARE LIBRARY
      CALL INITT (Ø)
C
C  SET UP A SCREEN AND VIEWING WINDOW
      CALL SWINDO (ISW(1),ISW(2),ISW(3),ISW(4))
      CALL DWINDO (RVW(1),RVW(2),RVW(3),RVW(4))
C
C  DRAW RINGER
C
C  FIRST CALCULATE AND MOVE TO BEGINNING POSITION
      Y = COS(RVW(1)) * EXP(CON * -RVW(1))
      CALL MOVEA (RVW(1),Y)
C
C  NOW LOOP THROUGH ALL THE POINTS
      DO 1ØØ T=RVW(1),RVW(2),.Ø5
      Y = COS(T) * EXP(CON * -T)
      CALL DRAWA (T,Y)
1ØØ   CONTINUE
C
C  DRAW A GRID
      DO 2ØØ X = RVW(1),RVW(2),2.
      CALL MOVEA (X,RVW(3))
      CALL DRAWA (X,RVW(4))
2ØØ   CONTINUE
      DO 21Ø Y = RVW(3),RVW(4),.2
      CALL MOVEA (RVW(1),Y)
      CALL DRAWA (RVW(2),Y)
21Ø   CONTINUE
C
C  TERMINATE TCS
      CALL FINITT (Ø,78Ø)
      STOP
      END
```

Successful execution of this program should yield an image similar to the curve in Fig. 1.

ORNL-DWG 80-3018 FED

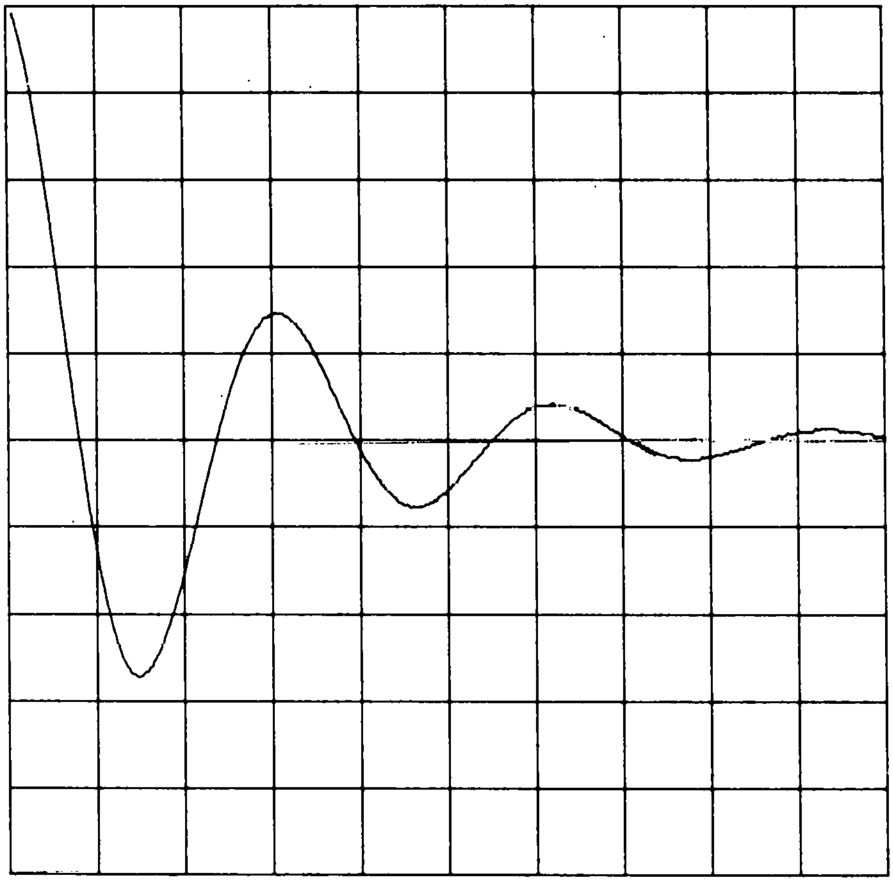

Fig. 1. The display image for the sample problem in Sect. 4.

5. MODIFIED TCS SOFTWARE

There are six major differences between the TCS software as received and that which is available in relocatable object module form. These differences are as follows.

(1) Subroutine INITT has been modified to access the transmission speed of a user's terminal (information that is available in the monitor table for any job number).

(2) Subroutines ERASE and NEWPAG have been modified to replace a call to the TCS subroutine IOWAIT with a call to subroutine SLP. This efficiently handles the timing delays required by a Tektronix terminal during a page erasure and refresh.

(3) Subroutine HDCOPY has been modified to accept an argument (if one is present) that represents the number of seconds to pause during a hard-copy operation. This modification was made because similar timing delay problems exist for hard-copy operations.

(4) The FORTRAN STOP statement in subroutine FINITT has been removed.

(5) At the request of several users, a call to subroutine TSEND has been inserted into subroutine AOUTST. This causes any characters placed into the internal TCS character buffer by subroutine AOUTST to be output promptly to the user's terminal.

(6) Finally, subroutine TCSLEV has been updated to reflect local version 6 of the TCS software.

The reasons for these modifications are discussed below.

5.1 MODIFICATIONS TO SUBROUTINE INITT

In order that the TCS software be initialized, a call to subroutine INITT with a single argument representing the terminal transmission baud rate (number of characters per second) is required. However, the transmission speed is known by the computer; how else could the computer communicate with a teletype? Therefore, a modification was made to subroutine INITT to have it call a function subprogram named ITRMBD.

Subroutine ITRMBD accepts one argument, the job number, and it returns the value of the transmission baud rate. This frees the user from having to worry about this value, especially since a user's program usually is executed from many terminals, each with a different transmission speed. The user need only call INITT with an argument of Ø to have the transmission speed determined by subprogram ITRMBD. (Subroutine ITRMBD is fully described in Sect. 8.1.)

However, this modification did not completely solve the problem of TCS internally determining transmission speeds. Numerous teletype lines are hardwired into the FED DC1Ø communications interface, which does not have the hardware/software to set transmission speed codes in the monitor table for a job connected to it. To alleviate this difficulty a program (SETSPD) was written to software-set the transmission speed codes of all the hardwired lines connected to the DC1Ø. The program is run from the system AUTO file when the system is initialized. (Program SETSPD is discussed further in Sect. 8.2.)

Taking advantage of the way in which subroutine ADEOUT (see Sect. 7.2) is written, another modification was made to the ORNL version of subroutine INITT for the purpose of making the TCSTEK software library more efficient. The vendor-supplied version of subroutine INITT initializes the TCS buffer type to 1. This buffer type forces the TCS software to solve the problems associated with system-supplied interline characters (especially carriage return-line feed sequences) by generating beam-positional moves back to the currently stored beam position before every TCS buffer is output, thus creating extra computational and I/O activity. The ORNL version of subroutine INITT initializes the TCS buffer type to 3. This buffer type forces the TCS software to assume that no system-supplied interline characters are appended to the output buffer as is the case with the ORNL version of subroutine ADEOUT, thus reducing the computational and I/O activity.

Calling sequence:

CALL INITT (IBAUD)

Argument list:

IBAUD is the transmission baud rate in characters per second. If IBAUD is Ø, then the transmission baud rate will be automatically determined.

Required subroutines: ITRMBD, SETBUF, RESET, and NEWPAG

Language: FORTRAN-1Ø

Subprogram listing:

```
C
C----------SUBROUTINE--INITT-------------------------TEKTRONIX, INC.----
C
      SUBROUTINE INITT(IBAUD)
      COMMON /TKTRNX/ TMINVX,TMINVY,TMAXVX,TMAXVY,TREALX,TREALY,
     + TIMAGX,TIMAGY,TRCOSF,TRSINF,TRSCAL,TRFACX,TRFACY,
     + TRPAR1,TRPAR2,TRPAR3,TRPAR4,TRPAR5,TRPAR6,KMOFLG(2),
     + KGNMOD,KPADV,KACHAR,KOBLEN,KTRAIL,KLEVEL,KPAD2,
     + KBAUDR,KGNFLG,KGRAFL,KHOMEY,KKMODE,KHORSZ,KVERSZ,KTBLSZ,
     + KSIZEF,KLMRGN,KRMRGN,KFACTR,KTERM,KLINE,KZAXIS,KBEAMX,KBEAMY,
     + KMOVEF,KPCHAR(5),KDASHT,KMINSX,KMINSY,KMAXSX,KMAXSY,KEYCON,
     + KINLFT,KOTLFT,KUNIT
      KBAUDR=IBAUD
C
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\                                                               \/\
C \/\ THE ENCLOSED CODE HAS BEEN MODIFIED FOR THE ORNL FED          \/\
C \/\ DECSYSTEM-1Ø IMPLEMENTATION OF THE TCSTEK SOFTWARE LIBRARY.   \/\
C \/\                                                               \/\
C
C  MODIFIED 17MAR8Ø
        ITRMBM = ITRMBD(-1)
        IF (ITRMBM .GT. Ø) KBAUDR = ITRMBM
        IF (IBAUD .GT. Ø) KBAUDR = IBAUD
C \/\                                                               \/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C
      KPAD2=KBAUDR/3Ø8+1
      KGNMOD=Ø
      KPADV=Ø
      KOBLEN=72
      KTERM=1
      KFACTR=4
C
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\                                                               \/\
C \/\ THE ENCLOSED CODE HAS BEEN MODIFIED FOR THE ORNL FED          \/\
C \/\ DECSYSTEM-1Ø IMPLEMENTATION OF THE TCSTEK SOFTWARE LIBRARY.   \/\
C \/\                                                               \/\
C
C * SET THE OUTPUT BUFFER FORMAT
C
C  MODIFIED 17MAR8Ø
      CALL SETBUF(3)
C \/\                                                               \/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C
      KINLFT=Ø
      KOTLFT=1
      CALL RESET
      CALL NEWPAG
      RETURN
      END
```

5.2 MODIFICATIONS TO SUBROUTINES ERASE AND NEWPAG

Subroutine IOWAIT's function is to transmit synchronous (SYN) ASCII characters (ASCII character code 37_8; numbers subscripted by 8 in this report refer to octal, or base 8, notation) to the terminal to prevent loss of output while the terminal is busy doing the previous instruction, such as a page erasure. To perform this task it multiplies the ITIME (the only argument to subroutine IOWAIT) by the transmission baud rate divided by ten and uses this figure as the number of SYN characters to transmit to the terminal. (The SYN ASCII character is a no-operation instruction for the terminal.) This method is at best inaccurate; at worst it is a tremendous unnecessary overhead on the system. Consider, for example, a call to IOWAIT with ITIME equal to 18Ø (such a call used to be made from subroutine HDCOPY). If a user is transmitting 96Ø characters per second, then 18Ø * 96Ø/1Ø, or 17,28Ø, SYN characters are being sent to the terminal — clearly a wasteful operation. By empirical determination it was found that a pause of 2 s was sufficient for terminals during a page erase and refresh. Therefore, in subroutines NEWPAG and ERASE the call to IOWAIT was replaced with a call to subroutine SLP with an argument of 2 s. (Subroutine SLP's function is simply to put the user's program to sleep for the specified number of seconds.) Also, before the call to SLP in these subroutines, a call to the TCS subroutine TSEND is included to ensure that the terminal operation begins at the prescribed time. (Subroutine TSEND forces out to the terminal the internal TCS buffer.)

Calling sequence for subroutine ERASE:

CALL ERASE

Argument list: None

Required subroutines: TOUTST, TSEND, SLP, and RECOVR

Language: FORTRAN-1Ø

Subprogram listing:

```
C
C----------SUBROUTINE--ERASE-------------------------TEKTRONIX, INC.-----
C
      SUBROUTINE ERASE
      DIMENSION ICODE(2)
      DATA ICODE(1),ICODE(2)/27,12/
      CALL TOUTST(2,ICODE)
C
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\                                                               \/\
C \/\ THE ENCLOSED CODE HAS BEEN MODIFIED FOR THE ORNL FED          \/\
C \/\ DECSYSTEM-1Ø IMPLEMENTATION OF THE TCSTEK SOFTWARE LIBRARY.   \/\
C \/\                                                               \/\
              CALL IOWAIT(1Ø)
C
C  WHG 18JLY8Ø
        CALL TSEND
        CALL SLP(2)
C \/\                                                               \/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C
      CALL RECOVR
      RETURN
      END
```

Calling sequence for subroutine NEWPAG:

CALL NEWPAG

Argument list: None

Required subroutines: TOUTST, TSEND, SLP, MOVABS, and ALFMOD

Language: FORTRAN-1Ø

Subprogram listing:

```
C
C----------SUBROUTINE--NEWPAG------------------------TEKTRONIX, INC.----
C
      SUBROUTINE NEWPAG
      COMMON /TKTRNX/ TMINVX,TMINVY,TMAXVX,TMAXVY,TREALX,TREALY,
     + TIMAGX,TIMAGY,TRCOSF,TRSINF,TRSCAL,TRFACX,TRFACY,
     + TRPAR1,TRPAR2,TRPAR3,TRPAR4,TRPAR5,TRPAR6,KMOFLG(2),
     + KGNMOD,KPADV,KACHAR,KOBLEN,KTRAIL,KLEVEL,KPAD2,
     + KBAUDR,KGNFLG,KGRAFL,KHOMEY,KKMODE,KHORSZ,KVERSZ,KTBLSZ,
     + KSIZEF,KLMRGN,KRMRGN,KFACTR,KTERM,KLINE,KZAXIS,KBEAMX,KBEAMY,
     + KMOVEF,KPCHAR(5),KDASHT,KMINSX,KMINSY,KMAXSX,KMAXSY,KEYCON,
     + KINLFT,KOTLFT,KUNIT
      DIMENSION ICODE(2)
      DATA ICODE(1),ICODE(2)/27,12/
      IF(KKMODE .NE. Ø)CALL ALFMOD
C * OUTPUT (ESC) (FF) FOR NEW PAGE
      CALL TOUTST(2,ICODE)
C
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\                                                              \/\
C \/\ THE ENCLOSED CODE HAS BEEN MODIFIED FOR THE ORNL FED          \/\
C \/\ DECSYSTEM-1Ø IMPLEMENTATION OF THE TCSTEK SOFTWARE LIBRARY.   \/\
C \/\                                                              \/\
              CALL IOWAIT(1Ø)
C
C  WHG 18JLY8Ø
        CALL TSEND
        CALL SLP(2)
C \/\                                                              \/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C
      IF(KLMRGN.EQ.Ø)GO TO 1Ø
      CALL MOVABS(KLMRGN,KHOMEY)
      CALL ALFMOD
      GO TO 2Ø
1Ø    KBEAMX=Ø
      KBEAMY=KHOMEY
2Ø    RETURN
      END
```

5.3 MODIFICATIONS TO SUBROUTINE HDCOPY

A similar timing problem exists during a software-generated hard-copy request. The user's program must delay output to the Tektronix terminal until the hard-copy operation is finished to ensure that no graphics are lost. Usually, 9 s is sufficient for this delay. However, if several terminals are multiplexed to a single hard-copy device, a longer delay is required. The data acquisition group for ORNL's Impurity Study Experiment has such a setup. Furthermore, terminal age also influences the time required for a hard-copy operation. To alleviate this timing problem, the following solution was programmed.

With the exception of 4ØØ6 series terminals, Tektronix hardware provides a facility for requesting a terminal to indicate the completion of a hard-copy operation. Subroutine HDCOPY has been rewritten in MACRO to make such an enquiry, thereby automatically ensuring that the exact hard-copy timing delay is used for the user's specific operational environment without user interaction. Warning — the default terminal type for the TCSTEK software library is a series 4Ø1Ø. If a user is plotting on a 4ØØ6 series terminal and does not tell the software this fact via subroutine TERM, the TCSTEK software will be waiting for five characters to be typed at the end of the hard-copy operation. Any five characters may be typed by the user (they will not be echoed); then the user's program will proceed normally.

In order not to penalize other users by having excessive delays during hard-copy operations and in order to make hard copies convenient for users of 4ØØ6 terminals, subroutine HDCOPY checks for the presence or absence of a single argument. If an argument is present, then regardless of the terminal type its value will be used as the integer number of seconds to put the user's program to sleep. If no argument is present and the terminal is a series 4ØØ6, then a 9-s delay will be used. If no argument is present and the terminal is a series 4Ø1Ø or greater, then the automatic enquiry will be made for the exact delay time.

Calling sequence with no arguments:

```
CALL HDCOPY
```

Calling sequence with one argument:

CALL HDCOPY (ISECS)

Argument list:

ISECS is the integer number of seconds to delay output during a hard-copy operation.

Required subroutines: TOUTST, TSEND, ADEIN, and SLP

Language: MACRO-1Ø

Subprogram listing:

```
        TITLE HDCOPY

REPEAT  0,<

        HDCOPY, A SUBPROGRAM TO MAKE A HARD COPY OF THE
        DISPLAY IMAGE ON THE SCREEN OF A TEKTRONIX TERMINAL

        AUTHOR:  W.H. GRAY
                 OAK RIDGE NATIONAL LABORATORY
                 OAK RIDGE, TENNESSEE 37830

>

        ENTRY HDCOPY

T0=0
T1=1
L=16
P=17

        .COMMON  TKTRNX[^D60]

KTERM=TKTRNX+50
KGNMOD=TKTRNX+25

SIXBIT  /HDCOPY/
HDCOPY:
        MOVEI    T0,11                 ;DEFAULT NUMBER OF SECONDS
        PUSH     P,L                   ;SAVE L
        MOVE     T1,-1(L)              ;GET NUMBER OF ARGUMENTS
        JUMPE    T1,STICK              ;IS IT ZERO
        MOVE     T0,@0(L)              ;NO, CALLER IS TELLING NUMBER OF SECONDS
STICK:  MOVEM    T0,SECS               ;STORE NUMBER OF SECONDS
        MOVEI    L,HDCS                ;ADDRESS FOR HARD COPY BLOCK
        PUSHJ    P,TOUTST##            ;INSERT HARD-COPY INSTRUCTIONS
        MOVEI    L,TSND                ;ADDRESS FOR TSEND BLOCK
        PUSHJ    P,TSEND##             ;SEND BUFFER
        POP      P,L                   ;RESTORE L
        MOVE     T1,-1(L)              ;GET NUMBER OF ARGUMENTS AGAIN
        JUMPN    T1,SLPIT              ;WAS AN ARGUMENT SPECIFIED
        SKIPE   KTERM                   ;NO.  IS TERMINAL >= 4010?
          JRST  ENQUIR                  ;YES, DON'T SLEEP
SLPIT:                                  ;NO, FORCED SLEEP
        MOVEI   L,SLPA                  ;ADDRESS FOR SLEEP BLOCK
        PUSHJ   P,SLP##                 ;GO TO SLEEP
        POPJ    P,

ENQUIR:                                 ;HERE IF ENQUIRE IS AVAILABLE
        MOVEI   L,ENQBLK                ;ADDRESS FOR ENQUIRE BLOCK
        PUSHJ   P,TOUTST                ;INSERT ENQUIRE INSTRUCTIONS
        MOVEI   L,TSND                  ;ADDRESS FOR TSEND BLOCK
        PUSHJ   P,TSEND                 ;SEND BUFFER
        SETOM   KGNMOD                  ;TURN ON GIN MODE FLAG
        MOVEI   L,ADEINP                ;ADDRESS FOR ADEIN BLOCK
        PUSHJ   P,ADEIN##               ;RECEIVE CHARACTERS FROM TERMINAL
        IONEOU  [7]                     ;TELL HIM ABOUT IT
        SETZM   KGNMOD                  ;TURN OFF GIN MODE FLAG
        POPJ    P,

        -2,,0
HDCS:   0,,NUM
        0,,ICODE

        0,,0
TSND:   0,,0

        -1,,0
SLPA:   0,,SECS

        -2,,0
ENQBLK: 0,,NUM
        0,,ECODE

        -2,,0
ADEINP: 0,,NCHR
        0,,IADE

NUM:    2
ICODE:  ^D27                            ;ADE CODE FOR ESCAPE
        ^D23                            ;ADE CODE FOR HARD-COPY
ECODE:  ^D27                            ;ADE CODE FOR ESCAPE
        5                               ;ADE CODE FOR ENQUIRE
SECS:   0
NCHR:   Z                               ;NUMBER OF ADE CHARACTERS RETURNED
IADE:   BLOCK   5                       ;ADE ARRAY FOR ADEIN ARGUMENT RETURN

        END
```

5.4 MODIFICATIONS TO SUBROUTINE FINITT

For some unexplained reason, the vendor-supplied version of subroutine FINITT included a FORTRAN STOP statement as the last executable statement in the subprogram. This statement has been removed and users are encouraged to include a call to subroutine FINITT as the last TCS software calls in their program. Among other functions, subroutine FINITT calls subroutine TSEND, ensuring that all of the internal TCS character buffer will be sent to the terminal before program termination.

Calling sequence:

CALL FINITT (IXEND,IYEND)

Argument list:

IXEND is the ending X position of the alphanumeric cursor. Units are in absolute Tektronix screen coordinates.

IYEND is the ending Y position of the alphanumeric cursor. Units are in absolute Tektronix screen coordinates.

Required subroutines: MOVABS, ALFMOD, and TSEND

Language: FORTRAN-1Ø

Subprogram listing:

```
C
C----------SUBROUTINE--FINITT-------------------------TEKTRONIX, INC.----
C
      SUBROUTINE FINITT(IX,IY)
      CALL MOVABS(IX,IY)
      CALL ALFMOD
      CALL TSEND
C
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\                                                                \/\
C \/\ THE ENCLOSED CODE HAS BEEN MODIFIED FOR THE ORNL FED            \/\
C \/\ DECSYSTEM-1Ø IMPLEMENTATION OF THE TCSTEK SOFTWARE LIBRARY.     \/\
C \/\                                                                \/\
        STOP
        RETURN
C \/\                                                                \/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C
      END
```

5.5 MODIFICATIONS TO SUBROUTINE AOUTST

At the request of several FED users, subroutine AOUTST has been modified to include a call to subroutine TSEND. This causes any characters placed into the internal TCS character buffer by subroutine AOUTST to be output promptly to the user's terminal.

Calling sequence:

CALL AOUTST (NCHAR,IAM)

Argument list:

NCHAR is the number of characters to be translated (unmodified by this subroutine).

IAM is the left-justified ASCII character array stored five characters per word (unmodified by this subroutine).

Required subroutines: KAM2AS, ANSTR, and TSEND

Language: FORTRAN-1Ø

Subprogram listing:

```
C
C----------SUBROUTINE--AOUTST------------------------TEKTRONIX, INC.----
C
      SUBROUTINE AOUTST(NCHAR,IAM)
      DIMENSION IADE(72),IAM(1)
      DATA MAXLEN/72/
      LEN=NCHAR
      IF(LEN .GT. MAXLEN)LEN=MAXLEN
      CALL KAM2AS(LEN,IAM,IADE)
      CALL ANSTR(LEN,IADE)
C
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\                                                               \/\
C \/\ THE ENCLOSED CODE HAS BEEN MODIFIED FOR THE ORNL FED          \/\
C \/\ DECSYSTEM-1Ø IMPLEMENTATION OF THE TCSTEK SOFTWARE LIBRARY.   \/\
C \/\                                                               \/\
C
C  FLUSH BUFFER
        CALL TSEND
C \/\                                                               \/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C
      RETURN
      END
```

5.6 MODIFICATION TO SUBROUTINE TCSLEV

Subroutine TCSLEV has been modified to reflect version 6 of the ORNL FED TCS software library.

Calling sequence:

CALL TCSLEV (LEVEL)

Argument list:

LEVEL is a three-word integer array. The first word will be returned with the value of the year in which the last modification to the TCS library took place. The second word will be returned with the value of the month and day in which the last modification took place. The third word will be returned with the local version number of the TCS library.

Required subroutines: None

Language: MACRO-1Ø

Subprogram listing:

```
C
C----------SUBROUTINE--TCSLEV------------------------TEKTRONIX, INC.----
C
      SUBROUTINE TCSLEV(LEVEL)
C * TCSLEV RETURNS THE LEVEL AND THE LAST MODIFICATION DATE OF THE
C * SOFTWARE. IT SHOULD BE UPDATED WHENEVER A MODIFICATION IS MADE.
      DIMENSION LEVEL(3)
C
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\                                                              \/\
C \/\ THE ENCLOSED CODE HAS BEEN MODIFIED FOR THE ORNL FED          \/\
C \/\ DECSYSTEM-1Ø IMPLEMENTATION OF THE TCSTEK SOFTWARE LIBRARY.   \/\
C \/\                                                              \/\
C * LEVEL (1) IS THE YEAR OF LAST MODIFICATION
      LEVEL(1)=8Ø
C * LEVEL (2) IS THE DAY OF LAST MODIFICATION
      LEVEL(2)=717
C * LEVEL (3) IS THE LEVEL NUMBER
      LEVEL(3)=6
C \/\                                                              \/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C \/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\/\
C
      RETURN
      END
```

6. TCS CHARACTER-MANIPULATING SOFTWARE

This section describes four machine-dependent subroutines for translating ASCII character strings required by the TCS software. Their names are constructs of the following mnemonics: convert, K; left-justified ASCII one character per word, A1; to, 2; right-justified ASCII one character per word, AS; left-justified ASCII five characters per word, AM. Therefore, to convert a left-justified ASCII text string to a right-justified array of characters in ADE format, the appropriate subroutine to call is KAM2AS. The next four subsections describe the calling sequence, argument list, and subprogram listing of subroutines KA12AS, KAM2AS, KAS2A1, and KAS2AM.

6.1 SUBROUTINE KA12AS

Subroutine KA12AS performs to Tektronix TCS specifications and may be used to convert left-justified ASCII characters (stored one character per word) into right-justified, zero-filled characters (stored one character per word).

Calling sequence:

CALL KA12AS (NCHAR,KA1,KADE)

Argument list:

NCHAR is the number of characters to be translated (unmodified by this subroutine).

KA1 is the left-justified ASCII character array stored one character per word (unmodified by this subroutine).

KADE is the right-justified, zero-filled ASCII character array stored one character per word (returned by this subroutine).

Required subroutines: None

Language: MACRO-1∅ (see Ref. 6)

Subprogram listing:

```
        TITLE    KA12AS

T∅=∅
T1=1
T2=2
```

```
L=16
P=17

        ENTRY    KA12AS

SIXBIT  /KA12AS/
KA12AS: PUSH     P,T2                ;SAVE T2
        HRRZ     T2,@Ø(L)            ;GET NUMBER TO TRANSLATE
        JUMPE    T2,BYEBYE           ;IS IT Ø
        MOVN     T2,T2               ;NO, MAKE IT NEGATIVE
        HRL      T2,T2               ;IN LEFT HALF
        HRRI     T2,@2(L)            ;NOW PUT IN THE OUTPUT ADDRESS
        HRRZI    T1,@1(L)            ;GET ADDRESS OF A1 CHARACTERS
LOOP:   MOVE     TØ,(T1)             ;GET AN A1 CHARACTER
        LSH      TØ,-^D29            ;MAKE IT ADE
        MOVEM    TØ,(T2)             ;PUT THE ADE CHARACTER AWAY
        AOJ      T1,                 ;INCREMENT ADDRESS FOR A1
        AOBJN    T2,LOOP             ;ARE WE FINISHED
        POP      P,T2                ;RESTORE T2
BYEBYE: POPJ     P,                  ;WE ARE DONE
        END
```

6.2 SUBROUTINE KAM2AS

Subroutine KAM2AS performs to Tektronix TCS specifications and may be used to convert left-justified ASCII characters (stored five characters per word) into right-justified, zero-filled characters (stored one character per word).

Calling sequence:

CALL KAM2AS (NCHAR,KAM,KADE)

Argument list:

NCHAR is the number of characters to be translated (unmodified by this subroutine).

KAM is the left-justified ASCII character array stored five characters per word (unmodified by this subroutine).

KADE is the right-justified, zero-filled ASCII character array stored one character per word (returned by this subroutine).

Required subroutines: None

Language: MACRO-1Ø

Subprogram listing:

```
        TITLE   KAM2AS

TØ=Ø
T1=1
T2=2
L=16
P=17

        ENTRY   KAM2AS

SIXBIT  /KAM2AS/
KAM2AS: PUSH    P,T2                    ;SAVE T2
        HRRZ    T2,@Ø(L)                ;GET NUMBER TO TRANSLATE
        JUMPE   T2,BYEBYE               ;IS IT Ø
        MOVN    T2,T2                   ;NO, MAKE IT NEGATIVE
        HRL     T2,T2                   ;IN LEFT HALF
        HRRI    T2,@2(L)                ;NOW PUT IN THE OUTPUT ADDRESS
        HRRZI   T1,@1(L)                ;GET ADDRESS OF AM CHARACTERS
        HRLI    T1,44Ø7ØØ               ;GET A GOOD BYTE POINTER
LOOP:   ILDB    TØ,T1                   ;GET AN AM CHARACTER
        MOVEM   TØ,(T2)                 ;PUT THE ADE CHARACTER AWAY
        AOBJN   T2,LOOP                 ;ARE WE FINISHED
        POP     P,T2                    ;RESTORE T2
BYEBYE: POPJ    P,                      ;WE ARE DONE
        END
```

6.3 SUBROUTINE KAS2A1

Subroutine KAS2A1 performs to Tektronix TCS specifications and may be used to convert right-justified ASCII characters (stored one zero-filled character per word) into left-justified, blank-filled characters (stored one character per word).

Calling sequence:

CALL KAS2A1 (NCHAR,KADE,KA1)

Argument list:

NCHAR is the number of characters to be translated (unmodified by this subroutine).

KADE is the right-justified, zero-filled ASCII character array stored one character per word (unmodified by this subroutine).

KA1 is the left-justified ASCII character array stored one character per word (returned by this subroutine).

Required subroutines: None

Language: MACRO-1Ø

Subprogram listing:

```
        TITLE    KAS2A1

TØ=Ø
T1=1
T2=2
L=16
P=17

        ENTRY    KAS2A1

SIXBIT  /KAS2A1/
KAS2A1: PUSH     P,T2                ;SAVE T2
        HRRZ     T2,@Ø(L)            ;GET NUMBER TO TRANSLATE
        JUMPE    T2,BYEBYE           ;IS IT Ø
        MOVN     T2,T2               ;NO, MAKE IT NEGATIVE
        HRL      T2,T2               ;IN LEFT HALF
        HRRI     T2,@2(L)            ;NOW PUT IN THE OUTPUT ADDRESS
        HRRZI    T1,@1(L)            ;GET ADDRESS OF ADE CHARACTERS
LOOP:   MOVE     TØ,(T1)             ;GET AN ADE CHARACTER
        LSH      TØ,^D29             ;MAKE IT A1
        MOVEM    TØ,(T2)             ;PUT THE A1 CHARACTER AWAY
        AOJ      T1,                 ;INCREMENT ADDRESS FOR ADE
        AOBJN    T2,LOOP             ;ARE WE FINISHED
        POP      P,T2                ;RESTORE T2
BYEBYE: POPJ     P,                  ;WE ARE DONE
        END
```

6.4 SUBROUTINE KAS2AM

Subroutine KAS2AM performs to Tektronix TCS specifications and may be used to convert right-justified ASCII characters (stored one zero-filled character per word) into left-justified, blank-filled characters (stored five characters per word).

Calling sequence:

CALL KAS2AM (NCHAR,KADE,KAM)

Argument list:

NCHAR is the number of characters to be translated (unmodified by this subroutine).

KADE is the right-justified ASCII character array stored one character per word (unmodified by this subroutine).

KAM is the left-justified, blank-filled ASCII character array stored five characters per word (returned by this subroutine).

Required subroutines: None

Language: MACRO-1Ø

Subprogram listing:

```
        TITLE   KAS2AM

TØ=Ø
T1=1
T2=2
L=16
P=17

        ENTRY   KAS2AM

SIXBIT  /KAS2AM/
KAS2AM: PUSH    P,T2                ;SAVE T2
        HRRZ    T2,@Ø(L)            ;GET NUMBER TO TRANSLATE
        JUMPE   T2,BYEBYE           ;IS IT Ø
        MOVN    T2,T2               ;NO, MAKE IT NEGATIVE
        HRL     T2,T2               ;IN LEFT HALF
        HRRI    T2,@1(L)            ;NOW PUT IN THE OUTPUT ADDRESS
        HRRZI   T1,@2(L)            ;GET ADDRESS OF AM CHARACTERS
        HRLI    T1,44Ø7ØØ           ;GET A GOOD BYTE POINTER
LOOP:   MOVE    TØ,(T2)             ;GET THE ADE CHARACTER AWAY
        IDPB    TØ,T1               ;PUT INTO AM
        AOBJN   T2,LOOP             ;ARE WE FINISHED
        POP     P,T2                ;RESTORE T2
BYEBYE: POPJ    P,                  ;WE ARE DONE
        END
```

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

7. TCS I/O SOFTWARE

This section describes two subroutines for character transmission between a user's program and a Tektronix terminal. These are subroutines ADEIN and ADEOUT, which mnemonically represent ADE I/O, respectively. Both subroutines are written in MACRO-1Ø; therefore, the TCS software library does not require the FORTRAN REAL TIME OPERATING SYSTEM[7] (FOROTS) to be in the user's high segment during execution.

7.1 SUBROUTINE ADEIN

Subroutine ADEIN performs to Tektronix TCS specifications and may be used to input right-justified, zero-filled ASCII characters from the user's terminal. The complete ASCII character set may be transmitted with this subroutine; however, there are certain characters that have a special meaning. These are discussed below.

(1) ^U If a ^U (ASCII character code 25_8; numbers subscripted by 8 in this report refer to octal, or base 8, notation) is typed, the input line is erased and the input may be started over.

(2) ^R If a ^R (ASCII character code 22_8) is typed, the current line is retyped to the user's terminal for visual inspection of mistakes. The contents of the input buffer are not altered by this character.

(3) (DEL) If a DEL (ASCII character code 177_8) is typed, the previously input character is thrown away and the terminal input line is backspaced one character. (If the terminal type has been set as Ø via TCS subroutine TERM, indicating 4ØØ6 terminal hardware, then a backspace is impossible; therefore, none is output to the terminal. However, the deleted character will still be discarded.)

(4) (BS) If a backspace (ASCII character code $1Ø_8$) is typed, it has the same effect as typing a DEL character.

(5) (CR) If a carriage return (ASCII character code 15_8) is typed, it is ignored. (Remember that on most terminals the

carriage return key actually transmits two characters to the host computer — both a carriage return and a line feed.)

(6) (LF) If a line feed (ASCII character code 12_8) is typed, input is terminated without inserting the line feed character into the input buffer, and program control is passed back to the calling subroutine.

Remember that during any single call a maximum of 72 characters may be input by this subroutine.

Tektronix 4∅1∅ series models have the capability of being used in the graphic input (GIN) mode. When an ESC SUB character sequence (ASCII character codes 33_8 and 32_8, respectively) is transmitted to a terminal via subroutine ADEOUT, the terminal's graphic cross-hair cursor will be turned on as a prompt for GIN mode input. By positioning the graphic cursor (using the thumb wheels located on the terminal) to a user-selected screen position and then striking a character, both the character (in ADE format) and four more characters are transmitted to subroutine ADEIN. The last four characters are decoded by the TCS software to locate the screen position of the graphic cursor.

Subroutine ADEIN is written to accept five and only five characters when the GIN mode switch is set; therefore, only a single character need be typed to transmit all the graphic cursor coordinate position information to the host computer. (The other four characters are internally generated and transmitted by the Tektronix terminal hardware.) This differs from previous implementations of the TCS software for the FED DECsystem-1∅, but it reflects the author's preference of execution.

The following list enumerates three special considerations of which all users of GIN mode should be aware.

(1) If either a carriage return or a line feed key is struck during GIN mode, then the terminal is returned to alphanumeric mode immediately because of the terminal hardware. This version of subroutine ADEIN sets the software alphanumeric output flag to zero, attempting to simulate the hardware action of the

terminal. However, setting the flag to zero may not work under all circumstances — be careful or simply do not transmit line feed characters during GIN mode.

(2) If a carriage return is struck, two characters are transmitted to the software — both a carriage return and a line feed. This version of subroutine ADEIN discards the carriage return character. The line feed will be passed back to the calling subroutine as the GIN mode response character. (See the previous special consideration.)

(3) During GIN mode the terminal echo is temporarily turned off, preventing the response character, as well as the four hardware-generated characters, from being echoed back to the terminal by the monitor. Under certain circumstances, if this were not done, very interesting errors could occur. However, if a user ^C's out of his program during GIN mode, he will have to turn the terminal echo back on manually by typing

```
.SET TTY ECHO
```

Upon normal return from subroutine ADEIN, the terminal echo characteristic will be restored to whatever it was prior to the call to this subroutine.

Calling sequence:

```
CALL ADEIN (NCHAR,KADE)
```

Argument list:

NCHAR	is the number of characters to be translated (unmodified by this subroutine).
KADE	is the right-justified ASCII character array stored one character per word (unmodified by this subroutine).

Required subroutines: None

Language: MACRO-1Ø

Subprogram listing:

```
        TITLE   ADEIN
        ENTRY   ADEIN

TØ=Ø
T1=1
T2=2
L=16
P=17

        .COMMON TKTRNX[^D6Ø]            ;ADDRESS OF TKTRNX COMMON

KGNMOD=TKTRNX+25                        ;ADDRESS OF GIN MODE FLAG
KKMODE=TKTRNX+4Ø                        ;ADDRESS OF VECTOR MODE FLAG
KTERM=TKTRNX+5Ø                         ;ADDRESS OF TERMINAL TYPE FLAG

SIXBIT  /ADEIN/
ADEIN:
        MOVE    TØ,KGNMOD               ;GET GIN MODE FLAG
        JUMPE   TØ,START ;DON'T CHANGE ECHO IF NOT GIN
        HRRZI   T1,-1                   ;SET UP LINE NUMBER FOR GETLCH
        GETLCH  T1                      ;GET THE LINE CHARACTERISTICS
        MOVEM   T1,LINCHR               ;SAVE THEM
        TLO     T1,4                    ;YES, TURN OFF ECHO
        HRRI    T1,-1                   ;SET UP LINE NUMBER FOR SETLCH
        SETLCH  T1                      ;SO
START:
        PUSH    P,T2                    ;SAVE T2
BEGIN:
        HRREI   T1,-^D72                ;WE NEED 72 ADE CHARACTERS
        SKIPE   KGNMOD                  ;UNLESS GIN MODE INPUT
        HRREI   T1,-5                   ;IN WHICH CASE WE ONLY WANT 5
        HRRZI   T2,@1(L)                ;ADDRESS OF IARRAY
        SETZM   @Ø(L)                   ;ZERO OUT NCHAR
        SETZM   @1(L)                   ;PLUS FIRST ADE CHARACTER
LOOP:   INCHRW  TØ                      ;GET A CHARACTER
        CAIN    TØ,15                   ;IS IT A CARRIAGE RETURN
          JRST  LOOP                    ;YES,IGNORE IT
        SKIPE   KGNMOD                  ;ARE WE IN GIN MODE
          JRST  GINCHK                  ;THEN THIS COULD BE A COORDINATE
        CAIN    TØ,25                   ;IS IT A ^U
          JRST  CONTLU                  ;YES, START OVER
        CAIN    TØ,22                   ;IS IT A ^R
          JRST  CONTLR                  ;YES, TYPE OUT
        CAIN    TØ,177                  ;IS IT A RUBOUT
          JRST  RUBOUT                  ;YES, GO PROCESS A RUBOUT
        CAIN    TØ,1Ø                   ;IS IT A BACKSPACE
          JRST  RUBOUT                  ;YES, TREAT AS IF RUBOUT
        CAIN    TØ,12                   ;IS IT A LINE FEED
          JRST  LF                      ;THEN INPUT FOR CALL IS DONE
GINCHK:
        MOVEM   TØ,(T2)         ;STORE CHARACTER
        AOS     @Ø(L)           ;BUMP NCHAR
        AOJ     T2,             ;INCREMENT ADDRESS
          JRST  SKIP            ;GO TO THE END OF THE LOOP
CONTLU:
        OUTSTR  CRLF            ;OUTPUT A CR-LF
          JRST  BEGIN           ;START OVER
CONTLR:
        PUSH    P,T1            ;SAVE T1
        OUTSTR  CRLF            ;OUTPUT A CR-LF
        MOVN    T1,@Ø(L)        ;GET NEGATIVE NUMBER OF CHAR.
        HRL     T1,T1           ;IN LEFT HALF
        HRRI    T1,@1(L)        ;GET ADDRESS OF CHARS.
LOP:    MOVE    TØ,(T1)         ;GET A CHARACTER
        IONEOU  TØ              ;OUTPUT IT
        AOBJN   T1,LOP          ;ARE WE FINISHED
        POP     P,T1            ;RESTORE T1
          JRST  SKIP
RUBOUT:
        SKIPG   @Ø(L)           ;HAVE WE GOT ANY CHARACTERS YET
          JRST  LOOP            ;NO, THEN GO BACK TO LOOP
        SOJ     T1,             ;DECREMENT COUNTER
        SOJ     T2,             ;DECREMENT ADDRESS
        SETZM   (T2)            ;ZERO RUBBED-OUT CHARACTER
        SOS     @Ø(L)           ;DECREMENT NCHAR
        CAIN    TØ,1Ø           ;WAS IT A REAL BACKSPACE
          JRST  LOOP
        SKIPE   KTERM           ;DO NOT BACKSPACE IF 4ØØ6 TERMINAL
        IONEOU  [1Ø]            ;BACKSPACE TERMINAL
SKIP:
        AOJL    T1,LOOP         ;ARE WE FINISHED
LF:
BYEBYE:
        POP     P,T2            ;RESTORE T2
        MOVE    TØ,KGNMOD       ;GET GIN MODE FLAG
        JUMPE   TØ,FIN          ;IF NOT GIN, DO NOT CHANGE ECHO
        MOVE    T1,LINCHR       ;RESTORE THE LINE CHARACTERISTICS
        HRRI    T1,-1           ;SET UP LINE NUMBER FOR SETLCH
        SETLCH  T1              ;SO
        CLRBFI                  ;THEN CLEAR THE INPUT BUFFER
        MOVE    TØ,@1(L)        ;GET FIRST TYPED CHARACTER
        CAIE    TØ,12           ;WAS IT A LINE FEED
          JRST  FIN             ;NO
        SETZM   KKMODE          ;YES, TERMINAL IS NOW IN ALPHAMODE
FIN:
        POPJ    P,              ;RETURN

CRLF:   ASCIZ/
/
LINCHR: Ø

        END
```

7.2 SUBROUTINE ADEOUT

Subroutine ADEOUT performs to Tektronix TCS specifications and may be used to output right-justified, zero-filled ASCII characters to the user's terminal. The complete ASCII character set may be transmitted with this subroutine because it uses the IONEOU machine instruction for output of a single packed image mode (PIM) character. This instruction bypasses several monitor level checks; e.g., after every 72 characters the monitor automatically inserts a carriage return-line feed sequence into the output stream. It is, therefore, an efficient way to transmit characters.

<u>Calling sequence:</u>

CALL ADEOUT (NCHAR,KADE)

<u>Argument list:</u>

NCHAR is the number of characters to be translated (unmodified by this subroutine).

KADE is the right-justified ASCII character array stored one character per word (unmodified by this subroutine).

<u>Required subroutines:</u> None

<u>Language:</u> MACRO-1Ø

<u>Subprogram listing:</u>

```
        TITLE   ADEOUT
        ENTRY   ADEOUT

TØ=Ø
T1=1
L=16
P=17

SIXBIT  /ADEOUT/
ADEOUT:
        MOVN    T1,@Ø(L)          ;GET NCHAR
        HRL     T1,T1             ;PUT IN LEFT HALF
        HRRI    T1,@1(L)          ;ADDRESS OF ADE CHARACTERS
LOOP:   MOVE    TØ,(T1)           ;GET A CHARACTER
        IONEOU  TØ,               ;OUTPUT IT
        AOBJN   T1,LOOP           ;ARE WE FINISHED
        POPJ    P,
        END
```

Because under the 6Ø3a monitor, buffered PIM I/O appears to be functioning correctly,[8] future system development of the TCS software library should consider replacing this subroutine with a subroutine that uses buffered PIM output.

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

8. TCS LOCAL SOFTWARE ENHANCEMENTS

This section presents two auxiliary subprograms and one program, completing the TCS system software on the FED DECsystem-1Ø. Program SETSPD is a privileged program that will set the transmission speed codes of teletypes connected to the FED DC1Ø communications interface, integer function ITRMBD will return the transmission baud rate for any job number currently active in the system, and subroutine SLP will put the controlling job to sleep for a specified number of seconds.

8.1 SUBROUTINE ITRMBD

Subroutine ITRMBD may be used to return the transmission baud rate for a particular system job number. The baud rate, that is, the number of characters per second being transmitted and received by the terminal, is approximately the transmission speed in bits per second divided by ten.

Calling sequence:

```
     IBAUD = ITRMBD(JOBNM)
```

Argument list:

JOBNM   is the system job number for which the baud rate is to be obtained. If JOBNM is equal to -1, then function ITRMBD returns the baud rate for the controlling terminal.

Required subroutines: None

Language: MACRO-1Ø

Subprogram listing:

```
        TITLE ITRMBD

        SEARCH UUOSYM
TØ=Ø
T1=1
L=16
P=17
        ENTRY ITRMBD
SIXBIT  /ITRMBD/
ITRMBD: SETZ     TØ,                ;CLEAR RETURN VALUE
        MOVE     T1,@Ø(L)           ;GET JOB NUMBER
        TRMNO.   T1,                ;GET JOBS UDX
          JRST   BYEBYE             ;BUMMER
        MOVEM    T1,UDX             ;STORE IN TRMOP BLOCK
        MOVE     T1,[XWD 2,CODE]    ;FORM TRMOP WORD
        TRMOP.   T1,                ;DO IT
          JRST   BYEBYE             ;ANOTHER POSSIBLE BUMMER
        MOVE     TØ,SPEED(T1)       ;GOOD RETURN LOOK UP CORRECT SPEED
        IDIVI    TØ,^D1Ø            ;CONVERT TO BAUD
BYEBYE: POPJ     P,                 ;WE ARE DONE
CODE:   1Ø31                        ;FUNCTION CODE FOR TRANSMIT SPEED
UDX:    Z
SPEED:  Ø
        ^D5Ø
        ^D75
        ^D11Ø
        ^D134
        ^D15Ø
        ^D2ØØ
        ^D3ØØ
        ^D6ØØ
        ^D12ØØ
        ^D18ØØ
        ^D24ØØ
        ^D48ØØ
        ^D96ØØ
        Ø
        Ø
        END
```

8.2 SUBROUTINE SLP

Subroutine SLP may be used to put a program to sleep for a specified number of seconds. This subroutine has been substituted for subroutine IOWAIT in the TCS subroutines HDCOPY, NEWPAG, and ERASE. The reason for this substitution is discussed in the system considerations section of this report.

Calling sequence:

```
    CALL SLP (ISEC)
```

Argument list:

ISEC   is the number of seconds to sleep.

Required subroutines: None

Language: MACRO-1Ø

Subprogram listing:

```
        TITLE    SLP
        ENTRY    SLP

TØ=Ø
L=16
P=17

SIXBIT  /SLP/
SLP:
        MOVE     TØ,@Ø(L)           ;NUMBER OF SECONDS TO SLEEP
        SLEEP    TØ,                ;GOOD NIGHT
        POPJ     P,                 ;GOOD MORNING
        END
```

8.3 PROGRAM SETSPD

Program SETSPD may be used to set the transmission speed code[9] (TSC) for any terminal connected to the FED computer system. This program is intended to set the TSC of all hardwired terminals connected to the DC1Ø communications interface because the DC1Ø's hardware cannot autobaud detect. The table named TABLE located under SUBTTL TTY-TABLE contains the TSC's and TTY line numbers set by this program.

This program is executed every time the system is initialized from the system AUTO file. The current source is located in a UFLIP file under account DSKA:SETSPD.UFL[5,2Ø2ØØ2,SETSPD].

Execution:

```
    .R SETSPD
```

Restrictions:

In order for this program to perform correctly, it must be executed from a privileged project-programmer account.

Language: MACRO-1Ø

Program listing:

```
        TITLE   SETSPD
        SEARCH  UUOSYM


.JBVER=137

VERNO==Ø1               ;MAJOR VERSION NUMBER
VEDIT==3                ;MAJOR EDIT NUMBER
VMINOR==Ø               ;MINOR EDIT NUMBER
VWHO==Ø6                ;WHG LAST EDITOR

VSTSPD==BYTE (3)VWHO(9)VERNO(6)VMINOR(18)VEDIT

        LOC     .JBVER
        VSTSPD
        RELOC

TØ=Ø
T1=1
T2=2
T3=3
T4=4
T5=5
T6=6
T7=7
T1Ø=1Ø
T11=11
L=16
P=17


        SUBTTL  THE-BIG-CODE

DEBUG:
SETSPD: JFCL
        RESET
        MOVE    P,[XWD PDLEN,PDLST]
        MOVEI   T1,NTBL                 ;GET NUMBER OF TTY'S TO SET
        JUMPE   T1,BYEBYE               ;ARE THERE ANY?
        MOVE    T3,[XWD 3,SXCODE]       ;YES, SET UP TRMOP FUNCTION
        MOVN    T1,T1                   ;NEGATE NUMBER OF TTY'S
        HRL     T1,T1                   ;IN LEFT HALF
        HRRI    T1,TABLE                ;GET ADDRESS OF TTY TABLE
OUTSTR  [ASCIZ/HELLO, MY NAME IS SETSPD AND I AM GOING TO SET THE XMTSPEEDS
OF THE FOLLOWING TERMINALS WHICH ARE CONNECTED TO THE DC1Ø
/]

LOOP:   MOVE    T2,(T1)                 ;GET TTY SPEED CODE AND #
```

```
        HLRM    T2,SPDCDE                   ;STORE SPEED CODE IN TRMOP BLOCK
        HRRZ    T2,T2                       ;GET TTY #
        ADDI    T2,.UXTRM                   ;MAKE IT A UNIVERSAL DEVICE INDEX
        MOVEM   T2,UDX                      ;PLACE IN TRMOP BLOCK
        MOVEI   TØ,2Ø31                     ;[3]GET SET XMTSPEED CODE
        MOVEM   TØ,SXCODE                   ;[3]PUT INTO TRMOP BLOCK
        TRMOP.  T3,                         ;SET IT
          JRST  BAD
        MOVEI   TØ,2Ø3Ø                     ;[3]GET SET RCVSPEED CODE
        MOVEM   TØ,SXCODE                   ;[3]PUT INTO TRMOP BLOCK
        TRMOP.  T3,                         ;[3]SET IT
          JRST  BAD
        OUTSTR  [ASCIZ/TTY/]
        SUBI    T2,.UXTRM                   ;GET TTY # AGAIN
        HRL     T2,T2                       ;MOVE TO LEFT HALF
        PUSHJ   P,OCTOUT                    ;OUTPUT TTY #

SPDOUT:
        HLRZ    T2,(T1)                     ;GET SPEED CODE AGAIN
        PUSH    P,T1                        ;SAVE T1
        MOVE    T1,SPEED(T2)                ;GET SPEED IN CHARACTERS/S
        OUTSTR  [ASCIZ/  TO        /]
        PUSHJ   P,DECOUT                    ;TYPE IT OUT
        POP     P,T1                        ;RESTORE T1
        OUTSTR  [ASCIZ/
/]
          JRST  CONT
BAD:
        OUTSTR  [ASCIZ/?ERROR - YOU ARE NOT PRIVILEGED TO SET TTY/]
        SUBI    T2,.UXTRM                   ;GET TTY NUMBER AGAIN
        HRL     T2,T2                       ;MOVE TO LEFT HALF
        PUSHJ   P,OCTOUT                    ;OUTPUT TO TTY
        JRST    SPDOUT                      ;TYPE OUT SPEED

CONT:
        AOBJN   T1,LOOP

BYEBYE:
        EXIT


        SUBTTL  SUBROUTINES

OCTOUT:
        MOVE    T6,[POINT 3,T2]             ;POINTER TO T2
        SETZ    T5,                         ;SET NONZERO OUT FLAG
        MOVEI   T4,6                        ;ONLY 1/2 WORD WORTH
OCTLOP: ILDB    T7,T6                       ;GET OCTAL CHARACTER
        SKIPN   T7                          ;NONZERO?
        JUMPE   T5,.+4                      ;NO, HOW ABOUT NONZERO FLAG
        ADDI    T7,6Ø                       ;MAKE IT ASCII
        OUTCHR  T7                          ;PRINT IT
        SETO    T5,                         ;NONZERO FLAG ON
        SOJG    T4,OCTLOP                   ;MORE CHARACTERS
        SKIPN   T5                          ;WAS IT Ø
        OUTCHR  ["Ø"]                       ;TYPE IT OUT
        POPJ    P,                          ;RETURN

DECOUT:
        IDIVI   T1,12                       ;DIVIDE BY 1Ø
        HRLM    T2,(P)                      ;SAVE REMAINDER
        SKIPE   Ø,T1                        ;SEE IF ANYTHING LEFT
        PUSHJ   P,DECOUT                    ;YES, GO BACK
        HLRZ    T1,(P)                      ;GET DIGIT BACK
        ADDI    T1,"Ø"                      ;MAKE IT ASCII
        OUTCHR  T1                          ;TYPE IT
        POPJ    P,                          ;LOOP BACK UNTIL DONE


        SUBTTL  TTY -TABLE

REPEAT  Ø,<

THE FOLLOWING LIST CONTAINS THE XMTSPEED CODE AND TERMINAL LINE
NUMBER IN HALF WORD FORMAT FOR ALL THE TERMINALS WHOSE XMTSPEEDS
ARE NOT SET BY HARDWARE, I.E., THE LINES CONNECTED TO THE DC1Ø.

LEFT HALF CONTAINS THE XMTSPEED CODE
RIGHT HALF CONTAINS THE TTY #

THE TRANSMISSION CODE IS LISTED BELOW

CODE    CHARACTERS PER S
Ø       Ø
1       5Ø
2       75
3       11Ø
4       134
5       15Ø
6       2ØØ
7       3ØØ
1Ø      6ØØ
11      12ØØ
12      18ØØ
13      24ØØ
14      48ØØ
15      96ØØ
16      EXTERNAL A
17      EXTERNAL B

FOR EXAMPLE SUPPOSE THE DC1Ø IS TRANSMITTING TO TTY1 AT 48ØØ CPS,
THEN THE CORRESPONDING ENTRY IN TABLE WOULD BE

        14,,1                               ;TTY1 XMTSPEED OF 48ØØ CPS
```

```
>
TABLE:
        15,,Ø
        15,,1
        15,,2
        7,,3
        7,,4
        7,,5
        7,,6
        7,,7
        7,,1Ø
        11,,11
        11,,12
        11,,13
        11,,14
        11,,15
        11,,16
        11,,17
        7,,2Ø
        7,,21
        7,,22
        7,,23
        7,,24
        7,,25
        7,,26
        7,,27
        11,,3Ø
        11,,31
        11,,32
        11,,33
        11,,34
        11,,35
        11,,36
        11,,37
NTBL=.-TABLE

        SUBTTL   MEMORY
SXCODE: 2Ø31                                ;CODE TO SET SPEED
UDX:    Z
SPDCDE: Z
SPEED:  Ø
        ^d5Ø
        ^d75
        ^d11Ø
        ^d134
        ^d15Ø
        ^d2ØØ
        ^d3ØØ
        ^d6ØØ
        ^d12ØØ
        ^d18ØØ
        ^d24ØØ
        ^d48ØØ
        ^d96ØØ
        Ø
        Ø
PDLST:  Ø,,PDERR
        BLOCK     1ØØ
PDLEN=.-PDLST
PDERR:  ASCIZ/?PDL OVERFLOW
/
        END       SETSPD
```

9. CONCLUSIONS

Hopefully, this report will sufficiently document the FED system dependencies of the TCS software so that future updates to this software product will be upwards compatible, thus ensuring system continuity between user application programs.

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

ACKNOWLEDGMENTS

I would like to express my appreciation to C. L. Thompson for his comments on the internal operation of the TCS software, to K. H. Carpenter for his suggestion to use PIM output in subroutine ADEOUT, to R. D. Burris for his patience in helping me debug MACRO subroutines, and to D. W. Swain for his empirical determination of the delay time for a Tektronix hard-copy unit.

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

REFERENCES

1. H. A. Kamel and M. W. McCabe, "The GIFTS System: Version 4B User's Manual," University of Arizona, Tuscon, Arizona (March 1978).
2. "Tektronix PLOT 1Ø Terminal Control System User Manual," Tektronix, Inc., Beaverton, Oregon (February 1977).
3. "Tektronix PLOT 1Ø Terminal Control System System Manual," Tektronix, Inc., Beaverton, Oregon (February 1977).
4. "Tektronix PLOT 1Ø Terminal Control System Installation Guide," Tektronix, Inc., Beaverton, Oregon (February 1977).
5. "Tektronix PLOT 1Ø Terminal Control System Verification Routine," Tektronix, Inc., Beaverton, Oregon (February 1977).
6. "DECsystem-1Ø MACRO Assembler Reference Manual," Digital Equipment Corporation, Maynard, Massachusetts (April 1978).
7. Appendix D in "DECsystem-1Ø FORTRAN Programmer's Reference Manual," Digital Equipment Corporation, Maynard, Massachusetts (June 1977).
8. C. L. Thompson (Oak Ridge National Laboratory), private communication, April 1979.
9. "DECsystem-1Ø Monitor Calls Manual," Digital Equipment Corporation, Maynard, Massachusetts (January 1978).

THIS PAGE

WAS INTENTIONALLY

LEFT BLANK

ORNL/TM-7353
Dist. Category UC-32

INTERNAL DISTRIBUTION

1. F. W. Baity
2. T. V. Baudry
3. W. B. Bigney
4. B. A. Clark
5. R. J. Colchin
6. W. A. Davis
7. R. A. Dory
8. L. C. Emerson
9. K. N. Fischer
10. G. E. Gorker
11-15. W. H. Gray
16. D. E. Greenwood
17. S. L. Halsted
18. C. E. Hammons
19. W. R. Hendrich
20. L. M. Jenkins
21. T. C. Jernigan
22. H. E. Ketterer
23. P. W. King
24. A. Komori
25. M. S. Lubell
26. L. B. Maddox
27. B. F. Maskewitz
28. R. W. McGaffey
29. G. S. McNeilley
30. J. K. Munro
31. D. R. Overbey
32. V. K. Paré
33. C. E. Parker
34. R. L. Reid
35. J. A. Rome
36. S. D. Scott
37. J. Sheffield
38. D. A. Spong
39. D. W. Swain
40. C. E. Thomas
41. T. C. Tucker
42. T. Uckan
43-44. Laboratory Records Department
45-46. Central Research Library
47. Document Reference Section
48. Laboratory Records, ORNL-RC
49. ORNL Patent Office
50-51. Fusion Energy Division Library
52. Fusion Energy Division Reports Office

EXTERNAL DISTRIBUTION

53. D. J. Anthony, General Electric Company, 1 River Road, Building 23, Room 290, Schenectady, NY 12345
54. S. O. Dean, Science Applications, Inc., 2 Professional Drive, Suite 249, Gaithersburg, MD 20760
55. H. K. Forsen, Exxon Nuclear Company, Inc., 777 106th Avenue, NE, Bellevue, WA 98009
56. H. P. Furth, Princeton Plasma Physics Laboratory, P.O. Box 451, Princeton, NJ 08540
57. R. L. Hirsch, Exxon Research and Engineering Company, P.O. Box 101, Florham Park, NJ 07932
58. Office of Assistant Manager for Energy Research and Development, Department of Energy, Oak Ridge Operations Office, Oak Ridge, TN 37830
59-241. Given distribution as shown in TID-4500, Mathematics and Computers (Distribution Category UC-32)