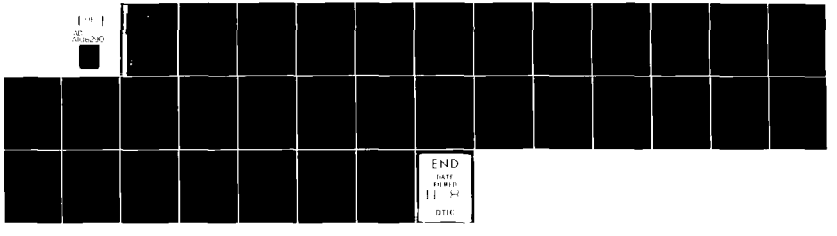
AD-A106 290 AERONAUTICAL RESEARCH LABS MELBOURNE (AUSTRALIA) F/G 9/2
COMMAND AND INFORMATION SYSTEM PROGRAMMING AND OPERATION MANUAL--ETC(U)
SEP 80 D K WARNE, T KINSELLA, H A THELANDER
UNCLASSIFIED ARL-SYS-NOTE-74 NL



END
DATE FILMED
11-81
DTIC

ARL-SYS-NOTE-74

AR-002-233

DEPARTMENT OF DEFENCE

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION

AERONAUTICAL RESEARCH LABORATORIES

MELBOURNE, VICTORIA

SYSTEMS NOTE 74

COMMAND AND INFORMATION SYSTEM PROGRAMMING AND OPERATION MANUAL

by

D. K. WARNE

with

T. KINSELLA and H. A. THELANDER

Approved for Public Release.

COPY No 22

SEPTEMBER 1980

AR-002-233

DEPARTMENT OF DEFENCE
DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION
AERONAUTICAL RESEARCH LABORATORIES

(14) [ARL-SYS-NOTE-74]

SYSTEMS NOTE 74

(6) # COMMAND AND INFORMATION SYSTEM PROGRAMMING AND OPERATION MANUAL

(11) S, 74

by

(1) D. K. WARNE

with

T. KINSELLA and H. A. THELANDER

(12) 33

SUMMARY

Command and Information System Mark 1 (CIS 1) provides for the real time control and display functions needed in the Aeronautical Research Laboratories Hybrid Computing System (HCS3). CIS 1 consists of a large cursive C.R.T. display interfaced via an A.R.L.-developed controller and a PDP-11/20 minicomputer to the A.R.L. DECsystem-10 central timesharing computer. CIS 1 also provides capability for general purpose interactive and static graphics applications. This note contains the reference manual for CISPAC, a DECsystem-10 FORTRAN-callable subroutine package for driving CIS 1. Two utility programs for on-line display of DECsystem-10 text and plot files are described. DECsystem-10 and PDP-11/20 programming for CIS 1 is covered, the hardware is described, and the process of generation of alphanumeric and symbolic character fonts for use with it is explained.

POSTAL ADDRESS: Chief Superintendent, Aeronautical Research Laboratories,
Box 4331, P.O., Melbourne, Victoria, 3001, Australia.

CCF 650

DOCUMENT CONTROL DATA SHEET

Security classification of this page: Unclassified

1. Document Numbers
   (a) AR Number:
   AR-002-233
   (b) Document Series and Number:
   Systems Note 74
   (c) Report Number:
   ARL-Sys-Note-74

2. Security Classification
   (a) Complete document:
   Unclassified
   (b) Title in isolation:
   Unclassified
   (c) Summary in isolation:
   Unclassified

3. Title: COMMAND AND INFORMATION SYSTEM PROGRAMMING AND OPERATION MANUAL

4. Personal Author(s):
   Warne, D. K.
   Kinsella, T.
   Thelander, H. A.

5. Document Date:
   September, 1980

6. Type of Report and Period Covered:

7. Corporate Author(s):
   Aeronautical Research Laboratories

8. Reference Numbers
   (a) Task:
   DST 20/022
   (b) Sponsoring Agency:

9. Cost Code:
   71 6160

10. Imprint:
    Aeronautical Research Laboratories,
    Melbourne

11. Computer Program(s)
    (Title(s) and language(s)):
    CISHCS, MACRO-11; CISPAC, CISTXT, CISPLA, MACRO-10.

12. Release Limitations (of the document)
    Approved for Public Release

12.0 Overseas:	N.O.		P.R.	1	A		B		C		D		E	

13. Announcement Limitations (of the information on this page):
    No Limitation

14. Descriptors:
    Hybrid computers
    Control
    Manuals
    Real time operations
    Display devices

15. Cosati Codes:
    0902

16.

ABSTRACT

Command and Information System Mark 1 (CIS 1) provides for the real time control and display functions needed in the Aeronautical Research Laboratories Hybrid Computing System (HCS3). CIS 1 consists of a large cursive C.R.T. display interfaced via an A.R.L.-developed controller and a PDP-11/20 minicomputer to the A.R.L. DECsystem-10 central timesharing computer. CIS 1 also provides capability for general purpose interactive and static graphics applications. This note contains the reference manual for CISPAC, a DECsystem-10 FORTRAN-callable subroutine package for driving CIS 1. Two utility programs for on-line display of DECsystem-10 text and plot files are described. DECsystem-10 and PDP-11/20 programming for CIS 1 is covered, the hardware is described, and the process of generation of alphanumeric and symbolic character fonts for use with it is explained.

CONTENTS

1. INTRODUCTION

Command and Information System Mark 1 (CIS 1) is used in the environment of the Aeronautical Research Laboratories Hybrid Computing System (HCS3), to provide a means for on-line real time monitoring of computations and their control.

CIS 1 consists of a Hewlett-Packard 1310A 483 mm (19 inch) diagonal cursive Graphic Display driven by a Digital Equipment Corporation PDP-11/20 computer through an A.R.L.-developed display controller. An alternative H–P 1310A unit coupled with a microfilm camera may be connected to the controller for recording purposes. Picture generation and overall system control is carried out by user programs in the A.R.L. DECsystem-10 central timesharing computer, which is interfaced to the PDP-11/20 by the A.R.L.-developed XIX interface. Interactive capability is available to the user through his DECsystem-10 timesharing terminal.

Four basic software packages are available for use with CIS 1, namely CISPAC, CISHCS, CISTXT, and CISPLA. CISPAC is a set of subroutines designed for use with FORTRAN or MACRO-10 user programs in the DECsystem-10. CISPAC performs all display code generation and communicates with the PDP-11/20 using the intercommunication and load-checking program module LOKXIX. CISHCS is a multi-purpose program package which runs in the PDP-11/20. It communicates with the DECsystem-10 program packages CISPAC for CIS 1 functions, H3PAC for hybrid computation, and RKPAC for DECsystem-10 use of the PDP-11/20's disk storage subsystem. In the CIS 1 application, CISHCS manages the picture data arrays in the PDP-11/20's core storage, and operates the display controller to provide a refreshed picture. CISTXT and CISPLA are DECsystem-10 user programs which utilize CISPAC to display text and plotter files respectively.

This Note describes CISPAC in Chapter 2. Chapters 3 and 4 present user's guides to the programs CISTXT and CISPLA, while Chapter 5 explains the programming in the DECsystem-10, and Chapter 6 the programming in the PDP-11/20, for the CIS applications. Chapter 7 covers the process of generation of alphanumeric and symbolic character fonts for use in CISPAC.

2. PROGRAMMING WITH CISPAC

CISPAC is a software package written in the MACRO-10 assembler language containing subroutines which may be called from DECsystem-10 FORTRAN (F10 compiler only) or MACRO-10 programs for general purpose use of the CIS 1 display. As far as possible CISPAC has been written in such a way that it can directly replace the equivalent package operating the 338 display (VISTRAN). CISPAC routines have names similar to those of their equivalents in VISTRAN, with the addition of the prefix 'C' to distinguish them. It would thus be possible, although probably not very useful, to write a program which used both the CIS and 338 Displays simultaneously. The major differences between the two packages arise from the fact that CISPAC generated pictures are stored only in the PDP-11/20, while VISTRAN pictures are stored in the DECsystem-10 and copied into the PDP-8 as required, so that a different system of transmission is required. CISPAC contains a transmission buffer which is loaded with display code during the execution of a CISPAC subroutine, and which will in general be sent to the PDP-11/20 on exit from the subroutine. However transmission may be inhibited until the buffer is full, when transmission will always take place.

CISPAC operation is initialised by a call to CERASA and terminated by a call to CFINIS, and all other calls must lie between these two. Pictures are stored as lists of 16 bit instructions in the area of PDP-11 core reserved for that picture by the CCLEAR or CCLERS call. There are two types of pictures, viz: main pictures and subroutine pictures, which differ only in that a main picture is always drawn relative to the origin (the bottom left hand corner of the screen) and has its intensity and scale initially set at the default values, while a subroutine is drawn

relative to the spot position at the time of the call to the subroutine and uses the intensity and scale settings existing at the time of the call unless they are explicitly set in the subroutine.

Corresponding to each picture in the PDP-11/20, the user must set up a 20-word data array in the DECsystem-10. This array is used to identify the picture in the various CISPAC subroutine calls and holds control information. This header array name is written as VSPACE, SSPACE, or PSPACE in the following CISPAC call descriptions depending on whether it is a main picture, subroutine picture or either respectively. In calls where the identification of the picture is optional, the current picture (generally the last picture specified in a call) is used when the picture is not specified by the user. Optional parameters in the calls are denoted by a bar over the top of the parameter in the ensuing descriptions.

The user is not limited to drawing his pictures on screen, however any off screen picture elements will not be displayed. To keep track of the user's picture, CISPAC makes use of both spot and beam coordinates, where spot is taken to mean the position specified by the user, which may be off screen, and beam signifies the physical location of the beam which will be the last on-screen spot position in the picture.

Use of the XIX interface for communication with the PDP-11/20 for operation of the CIS Display, for Hybrid Computation, or for access to the PDP-11/20's RK05 disks, requires that the user's job be appropriately privileged to exercise the DECsystem-10 operating system 'Real Time' features. To minimize the risk to DECsystem-10 integrity, the XIX communications program module LOKXIX contains code which enforces a program loading strategy which enables potentially hazardous data transfers to be detected and suppressed. For any program using CISPAC, or the Hybrid Computation package H3PAC, or the RK05 disk utility package RKPAC, the only acceptable loading order is LOKXIX first (nearest to virtual address 0), followed by the 'Real-Time' packages (CISPAC etc.) required, with the user's code last.

CISPAC may be used in conjunction with the hybrid computation system software package H3PAC, permitting real-time display of the results of hybrid computations to be performed. The last section of this Chapter details the use of CISPAC with H3PAC.

The following sections of this Chapter describe the various subroutines in CISPAC in detail.

2.1 CCLEAR (VSPACE, NDIM)

VSPACE is the name of the 20-word header array allocated by the programmer to a picture.

NDIM is the size of the picture in PDP-11 words.

This call must be used to indicate to CISPAC that the programmer is about to create a picture to be stored as a list of PDP-11 instructions in VSPACE. If VSPACE has already been validated by a CCLEAR call this CCLEAR call will mean that the old instructions will be deleted from VSPACE and subsequent CISPAC calls will create a new picture in VSPACE. If the old picture is in the display list it will be erased. If the new NDIM is not greater than the old NDIM, CISPAC will use the previously allocated core, otherwise this core will be lost to the user until the next CERASA call.

NDIM must not be less than 6 otherwise CISPAC will type an error message and the picture space will not be cleared. CCLEAR will automatically limit the size of the picture space to the available space in the PDP-11/20's core.

This call will also set the X and Y scales for the picture to one raster interval per unit, set the origin and the position of the beam and spot to (0, 0), set the lower X and Y limits to 0, the upper X limit to 1296, the upper Y limit to 2047, the brightness to 3 (maximum), set the resolution to 1 (i.e. 2 raster intervals), and the text font number to 0.

Once a legal CALL CCLEAR (VSPACE, NDIM) has been made, then VSPACE will be treated as a valid picture storage.

2.2 CAXES (X0, Y0, XS, YS, $\overline{\text{XL, XU, YL, YU, PSPACE}}$)

This call sets the origin, the X and Y scales and, if required, a restriction on the effective size of the screen used for PSPACE.

(X0, Y0) is the raster position to be used for the origin of co-ordinates relative to the lower left hand corner of the screen. XS and YS denote the number of raster intervals per unit

in the X (left to right) and Y (upwards) directions. The raster co-ordinates on the screen range from 0 to 1296 for X and 0 to 2047 for Y. The relationship between a programmers point (X, Y) and the corresponding (I, J) on the CIS Display screen will be:

$$I = X \times XS + X0$$

$$J = Y \times YS + Y0$$

XL, XU, YL and YU are parameters which must all be included or all omitted. If they are included the screen will be virtually restricted in size for elements inserted in the PSPACE by subsequent calls.

The numerical parameters can be REAL or INTEGER. X0 and Y0 need not be on screen. The origin, scales and restrictions will remain unchanged for PSPACE until altered by either a CAXES call or a CCLEAR call. If a CAXES call is made during the creation of a picture, the new origin, scale and restrictions will apply only to subsequent elements; elements which have already been inserted in the picture will not be altered.

2.3 CMARK (MK, $\overline{\text{VSPACE}}$)

The CMARK call causes CISPAC to insert into a 6 word array MK:

1. The address of VSPACE header.
2. The number of PDP-11 instructions in VSPACE.
3. The last data and parameter setting words sent.
4. The X and Y co-ordinates of the beam, and
5. The X and Y co-ordinates of the spot.

If the optional parameter VSPACE is included in the MARK call it will become the current picture name in CISPAC.

2.4 CMARKE (MK)

The CMARKE call causes the picture creation process in the picture whose address is in the first word of the array MK to proceed from the state of the picture at the last CMARK call. After the CMARKE call the picture referred to in array MK will become the current picture name in CISPAC.

These two calls (CMARK and CMARKE) can be used to erase and replace picture elements at the end of a list of display elements. They are illustrated by the following example:

FORTRAN CALLS	Effect.
CALL CCLEAR (A, 200)	Prepares for picture A.
—	Miscellaneous instructions and CISPAC calls.
CALL CMARK (MK)	Stores indicators in array MK as to the state picture A.
—	Miscellaneous instructions and CISPAC calls.
CALL CMARKE (MK)	Restores picture A back to the state at the CMARK call.

2.5 CDISPL ($\overline{\text{VSPACE1}}$, $\overline{\text{VSPACE2}}$, . . .)

This call adds pictures VSPACE1, VSPACE2, . . . to the list of pictures being displayed. If the call has no parameters the current picture (last picture referred to by a CISPAC call) will be used. If the call has a string of parameters the first of the parameters will become the

current picture after the call has been obeyed. If the value of the inhibit parameter is 1 or less the new display list specifications will be transmitted to the PDP-11/20.

2.6 CERASE (VSPACE1, VSPACE2, . . .)

This call erases pictures VSPACE1, VSPACE2, . . . from the list of pictures being displayed. If the call has no parameters the current picture (last picture referred to by a CISPAC call) will be used. If the call has a string of parameters the first of the parameters will become the current picture after the call has been obeyed. If the value of the inhibit parameter is 1 or less, the new display list specifications will be transmitted to the PDP-11/20.

Note that a CERASE call does not clear the display instructions from the PDP-11/20 core storage locations allocated to VSPACE1, VSPACE2, . . .; it merely removes the pictures from the display list.

2.7 CTRANS

This call transmits to the PDP-11/20 the current contents of the transmission buffer.

2.8 CERASA (ERASOK, FILMOK)

This call initializes CISPAC and the CIS part of CISHCS, erasing any pictures that were being displayed. Note that it does not clear the instructions from the picture storages; it only removes the pictures from the display list. A CERASA call does not change the current picture name in CISPAC. On return from the call the first optional argument, ERASOK, will be .TRUE. if the initialization was successful, and the second optional argument, FILMOK, will be .TRUE. if at least 1 metre of film is available in the microfilm camera, .FALSE. otherwise.

CERASA must be the first CISPAC subroutine called because it initializes DECsystem-10/PDP-11/20 communications. If a CALL CCLEAR is made after a CALL CERASA, the newly cleared picture will overwrite previous pictures in the PDP-11.

2.9 CFINIS (KEEP)

This routine stops the display and releases it for other users. If the optional argument KEEP is present and .TRUE., the currently displayed pictures are left on the screen, otherwise they are erased. After a CFINIS call, CERASA must be called to re-initialize the display and CISPAC if display operation is to be resumed.

2.10 CINHIB (I)

This call sets the value of the inhibit parameter to control the transmission of the current buffer as follows:

I .AND. 3 = 0 — No inhibit.

I .AND. 3 = 1 — Transmission from CDISPL, CERASE, CRFRSH, CPHOTO, CERASA, CFINIS and CTRANS.

I .AND. 3 = 2 — Transmission from CERASA, CFINIS, CRFRSH, CPHOTO and CTRANS.

I ≥ 4 — Error messages inhibited.

Note that transmission will always occur when the buffer is full irrespective of the inhibit parameter setting.

CERASA sets the inhibit parameter to 0.

2.11 CLENGT (L, $\overline{\text{PSPACE}}$)

This call sets L equal to the current number of 16 bit words used by the picture in PSPACE.

2.12 CFONT (N, $\overline{\text{PSPACE}}$)

This call determines the font (i.e. the character set) to be used in subsequent CTEXT calls referring to PSPACE. The font number is set to zero at clear time and whenever an undefined font number is requested. At the time of writing, Fonts 0 and 1 were available and contained the characters set out in Section 2.28.

2.13 CPHOTO (N)

CPHOTO causes a microfilm photo to be taken of the currently displayed pictures with the shutter being held open for N 'paints' of the picture, the film being advanced one frame before exposure. If N is zero, or there is no picture being displayed, the film will be advanced one frame and no exposure made.

If N is negative the shutter is held open for −N 'paints' of the picture and the film is not advanced. N must lie in the range −128 to 127 (only the least significant 8 bits are used). CPHOTO does not return to the calling program until the photo has been taken.

2.14 CRFRSH (N)

CRFRSH is used to control the refreshing of the CIS Display. If it is called with the argument N equal to zero, or if it has not been called since the last CERASA call, then normal refresh, under control of a clock in the CIS controller, occurs. If CRFRSH is called with the argument N not equal to zero, then N (modulo 128) 'paints' of the currently displayed picture are performed and the controller then waits for the next CRFRSH (or CERASA) call before resuming either normal refreshed display of the picture (CRFRSH(0) or CERASA) or refresh of N 'paints' (CRFRSH(N)).

2.15 CMOVE (X, Y, $\overline{\text{PSPACE}}$)

The CMOVE call moves the spot in PSPACE to position (X, Y) relative to the origin and with the scales set for the picture in PSPACE. The origin and scales must be set by a previous CCLEAR, CCLERS or CAXES call. This CMOVE call does not insert any visible element in the picture.

X and Y can be REAL or INTEGER.

2.16 CPOINT (X, Y, $\overline{\text{PSPACE}}$)

The CPOINT call moves the spot in PSPACE to position (X, Y). If the spot is within the limits set by a CCLEAR or CAXES call, it will insert a point at the nearest raster point to this position. If (X, Y) is outside these limits the spot will be moved to (X, Y) but the point will not be inserted in the picture.

X and Y can be REAL or INTEGER.

2.17 CVECTO (X, Y, $\overline{\text{PSPACE}}$)

The CVECTO call moves the spot in PSPACE to (X, Y) and inserts in PSPACE the vector between the last position of the spot and (X, Y). Vectors or parts of vectors outside the limits set by a CCLEAR, CCLERS or CAXES call will not be inserted.

X and Y can be REAL or INTEGER.

2.18 CRMOVE (X, Y, $\overline{\text{PSPACE}}$)

2.19 CRPOIN (X, Y, $\overline{\text{PSPACE}}$)

2.20 CRVECT (X, Y, $\overline{\text{PSPACE}}$)

The CRMOVE, CRPOIN and CRVECT calls do the same as the CMOVE, CPOINT and CVECTO calls respectively but X and Y are interpreted as being relative to the last position of the spot.

2.21 CTEXT (TEXT, NC, NS, $\overline{\text{PSPACE}}$)

This call paints in the picture the characters that the programmer has stored in TEXT, and that are within the limits set by a CCLEAR, CCLERS or CAXES call.

NC denotes the number of characters to be processed.

NS denotes the size of characters and indicates whether they should be rotated or not.

The character size may be in the range 0–3. If rotation, through 90 degrees counter clockwise, is desired, NS should be set to 4 plus the desired size. The actual displayed character size is determined by the matrix size for the font under consideration multiplied by 2^{NS}.

Characters interpreted by CTEXT are: Tab, Backspace, Carriage Return, Line Feed, and ASCII Codes 40–176 (octal). All other characters are ignored except for 'null' (octal value 0), which will immediately terminate the text string if encountered before NC characters have been processed. Carriage return sets the spot position to the X-origin (Y-origin if rotated), and tab positions are set after every eighth character from this origin.

The bottom L.H.S. of the first character will be placed at latest position of the spot in the picture. When the program returns from the CTEXT call the spot will be moved on to the bottom L.H.S. of the next character space (inside or outside the screen limits) after the last character in the TEXT.

2.22 CBRIGH (I, $\overline{\text{PSPACE}}$)

This call is used to set the brightness of subsequent elements painted in PSPACE. There are 4 levels of intensity (0–3) denoted by the value of I.

This brightness setting for PSPACE remains unchanged until altered by either a CBRIGH call or a CCLEAR call. A CCLEAR call sets I = 3, so that if full intensity is required a CBRIGH call is not needed after a CCLEAR call.

2.23 CRESOL (I, $\overline{\text{PSPACE}}$)

This call is used to set the raster resolution for incremental mode operations of subsequent elements painted in PSPACE. Coordinates set by point mode operations are suitably rounded to ensure compatibility of spot positioning. There are 4 levels of resolution denoted by the value I. The increment size is taken to be 2^I. The default resolution setting is I = 1 (i.e. 2 raster intervals). The operations affected are those generated by CMOVE, CPOINT, CVECTO, CRMOVE, CRPOIN and CRVECT calls.

2.24 CSETPA (INT, IRES, $\overline{\text{PSPACE}}$)

The CSETPA call enables the user to set both intensity and resolution parameters at the same time. It is more efficient than separate calls to CBRIGH and CRESOL if both parameters are to be changed.

2.25 User Generated Symbols

Subroutine pictures are generally less efficient than main pictures because all positioning is performed by incremental moves, and hence main pictures should be used wherever possible.

Subroutines are useful for pictures which are to be moved about the screen intact, or to be displayed simultaneously at several places on the screen (i.e. symbols).

2.25.1 CCLERS (SSPACE, NDIM)

SSPACE is the name of the header array, and
NDIM is the size of the picture in PDP-11 words.

CCLERS is similar to CCLEAR except that it specifies that this picture space be treated as a subroutine (symbol). It does not insert in the picture any instructions to set brightness or resolution, and hence unless the user alters these during the creation of the symbol, the brightness and resolution will be the same as that of the calling picture at the time that it calls the subroutine picture. If they are altered in the subroutine, the values of these parameters in the calling picture will not be changed. If only one parameter is set the other will be set to its normal default value.

Note that unless the user alters the resolution, CISPAC will use the default resolution for calculations in generating picture elements, so that scaling by a power of two will occur if the calling picture is not at the default resolution when it calls the symbol.

2.25.2 CCALL (SSPACE, $\overline{\text{PSPACE}}$)

CCALL is used to display a user generated symbol. The symbol must have been cleared as a subroutine by a CCLERS call prior to the CCALL.

No further elements can be added to a symbol once it has been called with CCALL, and it is thereafter treated as a full picture. The exception to this is that an already called symbol may be re-cleared with CCLERS and have new picture instructions inserted into it. This technique should be used cautiously, as it may result in the shifting of picture components generated by CISPAC calls following the CCALL to the symbol which is altered.

2.26 Storage Layout of Picture Space

Picture Space Address	Contents
+0	(left) Flags. *(see below) (right) Picture Space address.
+1	(left) Address of next picture in display list. (right) Address of previous picture in display list.
+2	PDP-11 address of the transmitted picture.
+3	(left) Maximum number of PDP-11 words allowed in this picture space. (right) Number of PDP-11 words in this picture space.
+4	(left) Last parameter word sent. (right) Last incremental data word sent.
+5	(left) X co-ordinates of beam. (right) Y co-ordinates of beam.
+6 and +7	X and Y co-ordinates of spot.
+8 and +9	X and Y co-ordinates of origin.
+10 and +11	X and Y scales.
+12 and +13	X Lower and Upper limits.
+14 and +15	Y Lower and Upper limits.
+16	(left) Text matrix size. (right) Text character table base address.

+17 Address of previous PSPACE in list of all valid PSPACEs.

+18 and +19 Reserved.

* The PSPACE flags are as follows

B0: PSPACE full

B1: PSPACE is a subroutine

B4: Beam is at spot position in CTEXT

B5: Parameters have been initialized in CTEXT

B7–B8: Used for packing incremental mode words together from separate calls where possible.

2.27 CISPAC Errors

There are several errors detected by CISPAC which may not be significant to further program operation (for example running out of space in a picture), and the CISPAC typeout of these error messages may be inhibited. Other errors detected by CISPAC can be assumed to be fatal to correct program operation, and following a typeout of an appropriate error message, program execution is terminated by CISPAC. The XIX interface communications managing and load checking module LOKXIX may also detect errors during the use of CISPAC. These may be fatal, or they may require action from the user before the program can proceed.

2.27.1 Inhibitable Errors

Inhibitable errors typeouts are of the form:
ERROR TYPE n IN xxxxxx CALLED FROM mmmmmm
where n is the error code as follows:

1 — Attempt to use an uncleared picture space

2 — Wrong type of picture space (attempt to CCALL a main picture or to CMARK a subroutine picture)

4 — Attempt to specify too small a picture space

5 — Mismatch of CMARK and CMARKE calls

6 — Attempt to use CISPAC calls prior to a CERASA call

8 — Attempt to overfill a picture space

10 — Incorrect number of arguments supplied

11 — Illegal font number in CFONT call

12 — Illegal argument address (argument inside Real Time packages, only checked for arguments written in by CISPAC).

xxxxxx is the CISPAC subroutine in which the error was detected, and

mmmmmm is the address of the call to xxxxxx in the program.

2.27.2 CISPAC Errors with Specific Messages

Fatal errors in this class, which result in program execution termination are:

CISPAC PHOTO CALL FROM R-T PART

This error only arises when CISPAC is in use with the Hybrid Computation software

package H3PAC, and an attempt is made to call the CISPAC CPHOTO subroutine from the Real Time part of the hybrid program.

CISPAC ROUTINE CALLED FROM T-S PART IN HYBRID ITERATE

This error occurs when a program using the Hybrid Computation software package H3PAC issues a call to a CISPAC subroutine from its Timesharing part while the program is in Hybrid Iterate state.

CISPAC ROUTINE CALLED BY F40-COMPILED CODE

CISPAC is not compatible with the old FORTAN compiler F40.

Occasionally, CISPAC may type the error message

CISPAC TRANSMISSION ROUTINE ERROR

This results from (usually) a failure in the PDP-11/20. Control is returned to the calling program, but CISPAC is de-initialized in the process (returned to pre-CERASA state), so that subsequent calls to CISPAC subroutines will provoke the 'ERROR TYPE 6 IN . . .' message.

2.27.3 Errors Detected by LOKXIX

If the program modules are not loaded in the correct order, LOKXIX will respond to a command to execute the program with the message:

JOB UNRUNNABLE
LOAD LOKXIX FIRST, THEN ANY R/T PACKAGES, THEN USER CODE

This message is fairly self-explanatory.

LOKXIX may be unable to initialize the PDP-11/20 for reasons which can be rectified by the user. The error messages are:

PDP-11 POWER DOWN — TURN IT ON AND TYPE C⟨CR⟩ (self-explanatory) and

CAN'T RE-INITIALIZE PDP-11—DO IT MANUALLY AND TYPE C⟨CR⟩ TO CONTINUE

This second message usually indicates that the PDP-11/20 is halted. It may be re-started at memory location 0.

Other error messages from LOKXIX indicate problems arising in the use of DECsystem-10 UUOs (monitor calls). Those involved may be GETTAB, WAKE, HIBER, CORE, LOCK, RTTRP, or TRPSET, and the cause of the error is usually that the job is not privileged to execute the UUO, or that another job is using the XIX interface. The LOKXIX error m essage

LOKXIX ERROR — AMBIGUOUS R/T DEV COUNT

is not expected to occur in normal operation. Systems programmer assistance should be sought if it is encountered.

2.28 CTEXT Character Sets

ASCII CODE	FONT 0	FONT 1
040	space	space
041	!	!
042	"	∀
043	#	≠
044	$	≈
045	%	∓
046	&	±
047	'	√
050	(	(
051	)	)
052	*	*
053	+	+
054	,	,
055	−	−
056	.	.
057	/	/
060	0	0
061	1	1
062	2	2
063	3	3
064	4	4
065	5	5
066	6	6
067	7	7
070	8	8
071	9	9
072	:	:
073	;	≤
074	<	<
075	=	=
076	>	>
077	?	≥
100	@	≠
101	A	A
102	B	B
103	C	Γ
104	D	Δ
105	E	E
106	F	Z
107	G	H
110	H	Θ
111	I	I
112	J	K
113	K	Λ
114	L	M
115	M	N
116	N	Ξ
117	O	O

ASCII CODE	FONT 0	FONT 1
120	P	Π
121	Q	P
122	R	Σ
123	S	T
124	T	ϒ
125	U	Φ
126	V	X
127	W	Ψ
130	X	Ω
131	Y	∞
132	Z	∫
133	[	[
134	\	∂
135	]	]
136	^	↑
137		
140	‾	←
141	a	α
142	b	β
143	c	γ
144	d	δ
145	e	ε
146	f	ζ
147	g	η
150	h	θ
151	i	ι
152	j	κ
153	k	λ
154	l	μ
155	m	ν
156	n	ξ
157	o	ο
160	p	π
161	q	ρ
162	r	σ
163	s	τ
164	t	υ
165	u	φ
166	v	χ
167	w	ψ
170	x	ω
171	y	∪
172	z	∩
173	{	{
174	|	|
175	}	}
176	~	
177	ignored	ignored

2.29 Use with H3PAC

CISPAC may be used in hybrid computation programs using the H3PAC system software package to perform, for example, continuous real time display of hybrid problem variables or solutions. The rules for use of CISPAC and H3PAC by the same program derive from the requirements of both packages to use the XIX interface for communication with the PDP-11/20. H3PAC has time-criticality requirements which dictate that CISPAC should not tie up the interface for long periods, and also operates in a regime where the DECsystem-10 is in a partial slave relationship with the PDP-11/20 and must be ready to accept interrupts from the PDP-11/20.

Hybrid computation program operation takes place in three program states: normal timesharing, hybrid ready, and hybrid iterate. These are fully described in the H3PAC manual.

There are no restrictions on the use of CISPAC subroutines when the program is in normal timesharing or hybrid ready states. In hybrid iterate state, CISPAC subroutines may not be called by the timesharing part of the program, but may, with the exception of CPHOTO, be called from the real time part.

To minimize real time part execution time requirements, it is preferable that as much CISPAC preparatory activity as possible be performed before the iterations of the real time part are started and the program enters hybrid iterate state. Calls to CERASA, CCLEAR (and CCLERS), and CAXES should all be possible prior to commencing hybrid activity, and parts of pictures drawn (and CMARKed) for adding to (or altering) from the real time part.

The program loading order rules outlined at the start of this Chapter should be observed when CISPAC and H3PAC are used together. LOKXIX must be first, followed by CISPAC and H3PAC (either one first) and user program modules must be loaded last. Failure to observe these rules will provoke an error message from LOKXIX when either CISPAC or H3PAC is initialized, and program execution will be terminated.

3. CISTXT USER'S GUIDE

CISTXT may be used for displaying text files on the CIS 1 display. To run CISTXT type:

RU CISTXT

when the job is in monitor mode. The program will return with the prompt character '*', to which a command string must be typed.

3.1 Command String

The general form of the command string is:

DEV:FILENAME.EXT [nnnn, nnnn]/S1/S2

or

DEV:FILENAME.EXT [nnnn, nnnn] (S1S2)

where:

DEV is the device on which the file is to be found (default device is DSK),

FILENAME.EXT is the name of the file to be displayed

[nnnn, nnnn] is the project programmer number (default is current user), and S1, S2 are switches.

On receipt of a valid command in this form, the program displays the first 'page' of the specified file and signifies its readiness to accept paging commands (see Section 3.3) by typing the prompt character ':'.

Other programs may be run from CISTXT with the command:

DEV:FILENAME.EXT [nnnn, nnnn]! In this case the default device is SYS rather than DSK.

To exit from CISTXT, an ESCAPE should be typed to the '*'.

Note: 'Wild' fields or characters are not acceptable in CISTXT commands.

3.2 Switches

SWITCH	ACTION	DEFAULT			
#B	Set Brightness to #	3			
#S	Set Scale to #	1			
#W	Set line Width to # characters	144	72	32	16
#L	Set page Length to # lines	170	85	42	21
	For character size:	0	1	2	3
X	Set picture length to maximum space available in PDP-11/20	4500 words			

CISTXT adds lines of text to the displayed page until a form feed is encountered in the input file, or the page or picture length limit is reached, when it awaits the next paging command (see Section 3.3). The default picture length limit gives a minimum refresh rate of approximately 40 Hz, providing a flicker free display. Use of the maximum picture space available allows viewing of more text simultaneously, at the expense of increased flicker.

3.3 Paging Commands

CISTXT types a ':' when it is ready to accept a paging command. The following commands are used:

F	Move forward one page
#F	Move forward # blocks (applies only to disk files)
B	Move backward one block (applies only to disk files)
#B	Move backward # blocks (applies only to disk files)
#	Go to block # (applies only to disk files)
P	Advance the microfilm one frame and photograph the current page. (A default 'paint' count in CISTXT determines the number of times the picture is refreshed while being photographed).
#P	Same as P, but the picture is refreshed # times while being photographed.
M and #M	Same as P and #P respectively except that the microfilm is not advanced, so that the current page is photographically superimposed on the previously photographed image.
Note:	P is the normal photo command. #P, M, and #M are provided primarily for experimental use. CISTXT limits # for these commands to 127.
A	Advance the microfilm one frame
#A	Advance the microfilm # frames.

The microfilm supply is checked whenever CISTXT is started, and if less than 1 metre remains, a warning message (MICROFILM SUPPLY LOW) is typed on the first subsequent execution of a microfilming command (and the user is given the option of having the command executed or ignored).

All commands except F, #F, B and #B must be terminated by an ESCAPE character. Two ESCAPEs will cause the program to return to the '*', ready to receive a new input specification. Carriage returns and linefeeds are ignored and are not required after a paging command. Rubouts will not work, but a multi-character command may be deleted by typing an I before the ESCAPE. An attempt to move backwards in the file beyond block 1, using B, #B or 0⟨ESC⟩ commands will result in a move to block 1.

The block number in which the currently displayed text commenced is displayed in the bottom left-hand corner of the screen. The block number display may be overwritten by Size 0 or Size 1 text in long pages if the lines are short enough and/or the picture length limit is set high enough to allow the default page length limit to be reached. This will not usually happen when the default picture length limit is used.

The block number display is erased from the screen while a microfilm photograph is taken and restored afterwards, so that only the information in the source file is photographed.

4. CISPLA USER'S GUIDE

CISPLA transfers PLOTTER files to the CIS 1 display.

4.1 Operation Procedures

4.1.1 Calling and Exiting

To run CISPLA, at the monitor level, the following is typed:

RU CISPLA

After initializing, CISPLA responds with its prompt character, '*', signalling the user to enter a command string. Then CISPLA executes the command string and types an asterisk again, allowing another command string to be typed. To exit from CISPLA, a Control C, a Control Z or an ESCAPE is typed.

If CISPLA is called from another program using the RUN UUO and with a starting address increment of 1, then the command string is read from the TMPCOR file CIS. If no TMPCOR file of that name exists the command string is read from the disk file nnnCIS.TMP where nnn is the user's job number in decimal with leading zeros. If no disk file exists of that name the command string is accepted from the teletype in the usual way.

4.1.2 CISPLA Switches

Some of the facilities of CISPLA are selected by switches in the command string.

CISPLA SWITCHES (ALPHABETICAL ORDER) SUMMARY

#A	The picture is rotated through # degrees.
#B	The picture is displayed with brightness #.
G	A 1 inch grid is included in the picture.
#H	Picture height is scaled by the ratio # to 1.
I	Ignores read errors.
M	A motion build up of the picture is produced.
#S	The picture is scaled by the ratio # to 1.
#W	Picture width is scaled by the ratio # to 1.
#X	The picture is shifted in the X direction # inches.
#Y	The picture is shifted in the Y direction # inches.

4.2 Commands

4.2.1 Command Strings

CISPLA recognizes command strings in both upper and lower case letters. Command strings may be up to 160 characters long. When the end of a line is reached the monitor supplies a RETURN and LINE-FEED and the user may continue typing the same command string.

When the command string is completed the user types a RETURN, and the command is executed.

All commands to CISPLA are formatted to contain any number of input deviecs. The character '=' or '_' delimits the output (destination) side of the command string from the input (source) side. The output side of the command string is only necessary when common switches are required.

For example:

DEV:NAME.EXT [PROJ, PROG], . . .

where:

DEV	Is a device mnemonic, e.g., DTA0, MTA1, or logical device name. If not specified DSK is assumed.
NAME	Is a filename, e.g. TEST, PLOT, DATA, or FILNAM.
EXT	Is the filename extension, e.g., PLT, DAT, TMP, separated from the filename by a dot. If a lookup failure occurs using the filename and extension specified, a second lookup is done with the extension PLT.
[PROJ, PROG]	Is a project programmer pair if other than the users project programmer pair. The two octal numbers are separated by a comma and are always enclosed in square brackets.

Note: 'Wild' fields or characters are not acceptable in CISPLA commands.

4.2.2 Switches for Optional CISPLA Functions

Any one of the options available in CISPLA may be selected by use of a switch, and entered into the command string by preceding it with a slash. If more than one switch is used they may be separated by slashes, or a group may be enclosed in parentheses.

Switches appearing in the destination side of the command string refer to all files in the command string. Switches on the source side of the command string refer to a particular file.

4.2.3 Shifting, Scaling and Rotating

The symbol # in the following switch definitions represents a signed floating point number.

#X, #Y	These switches shift the picture by # inches in the X and Y directions respectively.
#S	This switch scales the picture by a ratio of # to 1. Scaling is carried out after X and Y shifting. As with rotation, scaling is carried out about a point 5 inches from the initial position of the pen in the +X direction.
#W, #H	These switches scale the picture by a ratio of # to 1 in Width and Height respectively. They allow the use of separate scales in the X and Y directions.
#A	This switch rotates the picture # degrees Anticlockwise about a point 5 inches from the initial pen position in the +X direction after shifting and scaling.

4.2.4 Input Error Recovery

If the I switch is not used, any input error will type an error message and the current request to CISPLA is terminated. If the I switch is used processing will continue as through no error had occurred.

5. DECsystem-10 PROGRAMMING FOR CISHCS

5.1 PDP-11/20 Picture Formats

The first word of a picture in the PDP-11/20 is a pointer to the next picture in the display list, subsequent words contain display instructions, and the picture is terminated by two interrupt words with low order bytes equal to zero. Both main pictures and subroutines have the same form, being distinguished from each other by the CISHCS software when the end of picture interrupt is encountered. Hence the distinction between the two types of pictures is a matter for the DECsystem-10 program to decide since CISHCS can call either type as a subroutine or alternatively can display either. A call to subroutine consists of two interrupt words followed by the subroutine address. The lower order byte of the interrupt words must be negative in the first word and positive in the second word. The subroutine address is the address of the first display data word, not the display linkage word, i.e. it points to the second word of the picture.

5.2 Display List Linkage

The sequence in which pictures are displayed is determined by the first word of the pictures in the PDP-11/20. This word is a pointer to the next picture in the display list. Within CISHCS there is a 'null' picture (ISTPIC) that is always displayed and when no other pictures are being displayed this picture points to itself. To add a picture to the display list, the following procedure should be carried out:

(i) Deposit the address of the new picture in the first word of the last picture in display list.

(ii) Deposit the address of ISTPIC in the first word of the picture being added.

Both of the above steps should be carried out in one DECsystem-10 transmission to prevent the PDP-11/20 resuming displaying on a half completed display list alteration.

To erase a picture from the display list insert the address of the next picture in the display list in the first word of the previous picture in the display list.

5.3 DECsystem-10/PDP-11/20 Communications

5.3.1 Load Data Command (001 nnn)

The load data command consists of an 8-bit op code and an 8-bit word count. The word following the load data command is taken as the PDP-11 address at which the first word of the data is to be loaded. The next nnn (nnn is 377 octal maximum) words are treated as data to be loaded. When the load is complete, CISHCS automatically appends two end of picture interrupt words.

5.3.2 Load Word Command (002000)

The load word command causes CISHCS to treat the next word received as a load address and the subsequent word as the data to be loaded in that location. It is useful for altering display linkage pointers.

5.3.3 End of Transmission Command (003000)

This command causes the PDP-11 to dismiss the interrupt initiated by the transmission and resume displaying.

5.3.4 Initialise Command (004000)

On initialisation CISHCS erases all user pictures from the screen and sends to the DECsystem-10 a four PDP-11 word reply containing the following information: Top of Available PDP-11 core; Bottom of Available Core; Microfilm supply flag (−1 if more than 1 metre

remaining, 0 otherwise); and the address of the CISHCS picture ISTPIC. The notification to the DECsystem-10 of the available core is the only protection CISHCS provides for itself. It is left up to the program in the DECsystem-10 to ensure that no loading of data is attempted outside this area of PDP-11 core.

5.3.5 Photo Command (005 nnn)

This command causes CISHCS to advance the film one frame, open the shutter and use nnn as a signed 8 bit (2's complement form) count determining the number of 'paints' of the picture while the shutter is open. The next word of the transmission is not taken until the shutter has had time to close. A count of zero advances the film without operating the shutter. Negative counts suppress the film advance. CISHCS always waits for a clock interrupt (see Section 6.3) before commencing a 'paint' while executing a Photo command. This ensures mains synchronization to eliminate 'swimming' of the display and consequent blurring of the photographic image.

5.3.6 Refresh Command (006 nnn)

CISHCS normally refreshes the displayed picture as fast as program execution allows, or once per clock cycle (see Section 6.3), whichever is slower. CISHCS interprets the nnn in the Refresh command as a signed 8-bit (2's complement) 'paint' count. A Refresh command with a positive count causes CISHCS to paint the picture nnn times, as fast as program execution allows, and then cease refreshing until a further Refresh (or Initialize) command is received. A Refresh command with a negative count, or an Initialize command, restores normal refresh operation. (CISPAC never sends CISHCS a Refresh command with a zero count, as this would just stop CISHCS from refreshing the display.) Refresh commands do not affect the operation of Photo commands.

6. PDP-11 PROGRAMMING FOR THE DISPLAY CONTROLLER

6.1 Operating Principles

A schematic of the display controller is set out in Figure 1. The PDP-11 transmits an instruction (Fig. 2) to the controller via a DR11-A interface. Decision logic then identifies the type of instruction and takes the appropriate action as follows:

6.1.1 Interrupt Instructions

Interrupt instructions interrupt the PDP-11 processor at priority level 5. These instructions enable the controller to signal the processor that some action is required on its part and are used in such cases as end of picture and call to subroutine picture.

6.1.2 Parameter Setting Instructions

These instructions are used to set resolution and intensity parameters and to control the microfilm camera. The resolution and intensity registers are hardwired to the interface's input buffer with the parameter setting op-code and so can be read at any time by the processor. The op-code is included so that the word read may be used to restore the parameters. Intensity control is obtained by adjusting the voltage level of the intensity pulse while resolution control, which affects incremental mode operations only, determines by how many points the coordinate registers will be counted in each increment.

6.1.3 Point Mode Instructions

These instructions load a particular value into either the X or Y coordinate registers with the option of incrementing the other register. In addition, one bit of the instruction determines whether the point should be intensified or not.

6.1.4 Incremental Mode Instructions

Incremental mode instructions contain up to three increments in a single instruction. Timing circuitry within the controller controls the sequential implementation of the three increments. Each incremental move may operate on either or both X and Y coordinates, in either direction and with the option of intensifying the point or not.

6.2 Instruction Formats

The four types of instructions are identified by the first three bits of the instruction word as follows:

Incremental Mode Instruction : 0xx

Point Mode Instruction : 10x

Parameter Setting Instruction : 110

Interrupt Instruction : 111

'x' signifies the bit is not used in instruction classification. A schematic of bit functions is given in Figure 2.

Two interrupt words must always be transmitted consecutively to guarantee an interrupt.

6.3 Interrupts and Addressing

The display controller can interrupt via either of the two interrupt request lines of the DR11A. An interrupt request A is generated by an interrupt instruction within the received data as set out above, while an interrupt request B is generated once per A.C. mains cycle, enabling picture repetition rate control and mains synchronization. (Alternatively, an internal oscillator can be selected and its frequency set by front panel controls, to provide maximum repetion rate limits in the range 40 to 500 Hz for experimental purposes.)

For programming purposes, the relevant addresses are:

Interrupt Request A Vector = 310

Interrupt Request B Vector = 314

DR11A Status Register = 767760

DR11A Output Buffer Register = 767762

DR11A Input Buffer Register = 767764.

6.4 CISHCS Operation

CISHCS normally requires no operator intervention. It is arranged to start itself automatically, using the Power Up/Down Interrupt capability of the PDP-11/20, when power is applied. The XIX interface rack should be powered up before the PDP-11/20 to avoid a processor halt which sometimes occurs if this is done in the reverse order. CISHCS may also be started at address zero if it is halted for some reason.

7. FONT GENERATION FOR CISPAC

The text code for CISPAC generated pictures is stored in font tables within CISPAC. A font table consists of 24 words of offsets followed by the display code for the ASCII graphic characters (Octal codes 41 to 176). The characters carriage return, line feed, tab, and back space are handled specially by CTEXT, the delete character (octal code 177) is ignored, and a null (code 0) is taken to terminate the text argument. The 24 words at the beginning of the table are broken up into four 9-bit bytes, each byte being the offset to the relevant character taken from the base of the font table.

Corresponding to each font there is a pointer in a table FONTAB. The right half of the pointer contains the base address of the font table while the left half is broken up into two 9-bit bytes containing the character and line spacing. The number of fonts allowed to the user is determined by a parameter MFONT within CISPAC. The font number requested in a FONT subroutine call must be between 0 and MFONT inclusive, otherwise CISPAC gives an error message and assumes font number 0.

To aid in font generation, two programs have been developed viz: CHRGEN and STGEN. CHRGEN is an interactive program used for the generation of text codes and STGEN processes the output of CHRGEN into a suitable table form. CHRGEN generates the file FONTn.COD (where n is the font number) and this file should be kept in case of future updates of the font. STGEN processes FONTn.COD to eliminate any superseded characters from it creating a new FONTn.COD and renaming the old file FONTn.BAK, and in addition produces the table file FONTn.TAB which contains MACRO-10 source code for insertion into CISPAC.MAC using TECO.

7.1 CHRGEN Users' Guide

7.1.1 Function of the Program

The program CHRGEN enables a user to generate codes for the PDP-11 generation of characters with the CISPAC subroutine CTEXT. The character is produced using the 338 interactive display by intensifying points on a matrix of predetermined size, and the moves involved are converted to the required codes, which are then stored on a PDP-10 disk file for subsequent processing.

7.1.2 Running the Program

Before attempting to run the program, the operator should ensure that the file CHRGEN.SAV is on his disk area. The program can then be started by typing:

RU CHRGEN

when the job is in monitor mode.

7.1.2.1 Initial Dialogue

Font Number

Before characters can be produced, the characteristics of the font (the set of characters) are needed. The program firstly searches the operator's disk area for the file FONT.NUM, and if found, reads the number characterising the font. If the file is not found, the program asks:

FONT NO. IS(DECIMAL):

to which the operator should type a (decimal) integer between 0 and 99. If the number is outside this range, the program types an error message and reasks the question.

Character Dimensions

When the program has a legal font number (either from reading it or from teletype input), it then makes (or supersedes) the file FONT.NUM with the font number in it, and searches the user's disk area for the file FONTn.COD, where n is the font number. If the file is found, the program reads the matrix width and height (by which the character size is determined) and the character and line spacing (by which the character spacing on a page is determined). If the file is not found, the program asks:

TYPE MATRIX WIDTH AND HEIGHT (DECIMAL):

to which two decimal integers separated by a comma should be typed. The integer for the width should be in the range 1 to 19 and for the height between 1 and 29. The program then asks:

TYPE CHARACTER AND LINE SPACER (DECIMAL):

to which two integers separated by a comma should also be typed. The character spacer should be greater than the character width but not greater than 29, and the line spacer greater than the matrix height but not greater than 39. If any of the above four parameters is out of range, the program responds with:

?ERROR IN WIDTH AND/OR HEIGHT. TRY AGAIN

and asks for the four parameters again.

7.1.2.2 Commands

If the four above parameters were requested from teletype, the program then makes the FONTn.COD file (which will also contain the character codes) and then asks:

TYPE ASCII CODES IN OCTAL TO THE '*'

and types an asterisk, signifying it is ready for character generating. At this stage the operator may then type in an octal number between 41 and 176 (octal), which will label the subsequently produced character, or one of the letters N, S, E, C, P or H whose effect will be described below.

The 'N' Command

When the files FONT.NUM and FONTn.COD exist on the operator's disk area, the questions concerning font number and character dimensions are not asked, which prevents the operator from changing these if necessary. Typing N to the asterisk causes these questions to be asked, and hence a new font can be produced. Typing the current font number after an N command has no effect.

The 'S' Command

Typing S to the asterisk causes the program to type out the font number and character dimensions, i.e. the current status, after which another asterisk is typed.

The 'E' Command

Typing E to the asterisk causes the program to exit. This can also be effected by typing Control-C.

The 'C' Command

Typing C has no effect but to produce another asterisk. Use of this command is envisaged after typing Control-C and CONTINUE (for instance, if the operator changes his mind about finishing) to reassure the operator of the program's current state.

The 'P' Command

Typing P to the asterisk causes the program to close all files and run the program STGEN, which processes the information in FONTn.COD. If the file STGEN.SAV is not on the operator's area, the message

?STGEN.SAV NOT FOUND

is typed and control is returned to the monitor.

The 'H' Command

Typing H to the asterisk causes the program to type out a summary of the above commands and their effect, after which another asterisk is typed.

7.1.2.3 Character Generation

Any type-in (to the asterisk) which is not one of the letters N, S, E, C, P or H is treated as an octal number labelling the character to be produced. If it is not in the range 41 to 176 octal inclusive, or is an illegal character, the program responds:

?ASCII NO. OUT OF RANGE. TRY AGAIN OR TYPE H FOR HELP

and types another asterisk (no other effect occurs). The same ASCII character may be generated several times, the last one being the code finally taken by STGEN.

When the program has a legal ASCII number, a question is asked on the display screen, thus:

INTENSIFY?
YES NO

and refers to the point in the bottom left corner of the matrix on which the character is to be generated. This point is the starting point for all characters. By penning the appropriate word with the light pen, the point will be intensified with a white square or left alone accordingly. The first point will then be surrounded by crosses on the surrounding eight points and one of these should be selected with the light pen. The cross chosen will then be replaced by a cross-hatch (#) and the other crosses erased. The question will then be asked on the display:

INTENSITY?
YES NO RUBOUT

in the top left corner and selecting the appropriate word with the light pen will cause the point to be intensified, left alone or the previous cross selection erased accordingly, (the last would be chosen if a mistake had been made). By continuing in this manner, selecting crosses and intensifying or not intensifying, characters can be produced. It will be noted that as points are intensified, three characters are drawn at the bottom of the screen to show how the character will appear when called from CTEXT. The three characters correspond to text sizes 1–3.

7.1.2.4 Starting Again

If the character produced is not satisfactory, or an irretrievable mistake is made, the character can be started again by depressing the manual interrupt button. The display will then ask:

START AGAIN FINISHED

and the words 'START AGAIN' should be penned. The effect of penning 'FINISHED' will be discussed below. The display will then ask:

SAME ASCII CODE?
SAME NEW

Penning the word 'NEW' causes an asterisk to be typed, which can then receive commands, or an ASCII code; penning 'SAME' causes the program to return to the beginning of the character.

7.1.2.5 Finishing

When a character is completed to the operator's satisfaction the centre of the crosses must be moved to the bottom of the matrix and the end of the character space. Then the operator can depress the manual interrupt button and pen the word "FINISHED", at which time an asterisk will be typed and the program will be ready for another ASCII code or a command.

If the character is not finished at the bottom of the matrix, the program will type:

?CHARACTER NOT FINISHED AT BOTTOM OF MATRIX
Y COUNT = n PLEASE CONTINUE

where n represents the number of points the centre of the crosses is from the bottom of the matrix (positive for above, negative for below). The program will then return to the state waiting for a cross to be penned.

If the character is not finished at the end of the character space, the program will type:

?CHARACTER NOT FINISHED AT END OF CHARACTER SPACE

X COUNT = n PLEASE CONTINUE

where n represents the number of points the centre of the crosses is from the left of the matrix. The program will then return to the state waiting for a cross to be penned.

It should be noted that the method of drawing the character in the least number of moves is the best. Consequently, if the character is not finished in the right place and moving to the correct end points involves retracing steps, the character should be re-drawn.

7.1.3 Miscellaneous

This program will only allow a maximum of 100 moves per character. If the number of moves goes beyond 100, the program returns to the 'START AGAIN' state, and the codes for that character will be lost.

It should also be noted that should the DECsystem-10 system 'crash' only the codes for the current character will be lost.

7.2 STGEN Users' Guide

7.2.1 Function of the Program

STGEN processes the output file FONTn.COD produced by CHRGEN to produce a new FONTn.COD and a Font table for use in CISPAC in the file FONTn.TAB. The old version of FONTn.COD is renamed FONTn.BAK. If CISPAC.MAC is on the user's disk area STGEN will also check the legality of the font number n.

7.2.2 Running the Program

If the file STGEN.SAV appears on the user's disk area, STGEN may be run by either typing a 'P' command to CHRGEN or by typing RU STGEN to the monitor.

7.2.3 Operating Dialogue

Generally dialogue will be minimal if the FONTn.COD file has not been tampered with since CHRGEN produced it, and the font number specification to CHRGEN was correct.

STGEN initially searches for the FONT.NUM file produced by CHRGEN in order to obtain the font number. If the file is found it is read and deleted, if not, the user is requested to enter the font number with:

FONT.NUM NOT FOUND SPECIFY FONT NO.

If the font number obtained from either the FONT.num file or the user does not lie in the range 0–99, STGEN returns with:

FONT NO mmm IS ILLEGAL SPECIFY NEW ONE:

where mmm is the illegal font number specification. STGEN then looks up the appropriate FONTn.COD file and checks that its font number and file name match. If there is a mismatch it types the following error message and exits:

MISMATCHING FONT NO. AND FILE.

For this reason it is inadvisable to try and correct errors in the FONTn.COD file with TECO, but rather any correction should be made with STGEN or CHRGEN as appropriate.

STGEN then looks for CISPAC.MAC on the user's disk area to check the legality of the font number. If CISPAC.MAC is not found, STGEN types:

CISPAC.MAC NOT FOUND, NO CHECK MADE ON FONT NO. LEGALITY
DO YOU WISH TO CONTINUE:

A 'Y' reply will cause the program to continue, however, the TECO instructions given at the end of the program will be meaningless. A carriage return will cause the program to exit.

If the new font number is not greater than the MFONT parameter in CISPAC, the program assumes you are superseding an old font, while if the font number equals MFONT+1 it will assume you are adding a new font. However, before proceeding it will ask the user to confirm with either of the following as appropriate:

SUPERSEDING FONT n
CONFIRM:

or

CREATING NEW FONT NO: n
CONFIRM:

A 'Y' response will cause the program to commence processing the text code. If the user does not confirm, or if the font number is greater than MFONT+1, STGEN will attempt to sort out the situation asking the following questions as appropriate. Once enough information has been obtained it will once again ask the user to confirm as set out above.

FONT NO OUT OF BOUNDS, DO YOU WANT TO ADD THIS AS NEW FONT:
IS xxxxxx THE FILE YOU WANT:

where xxxxxx is the name of the file currently being processed.

WHAT NO. WOULD YOU LIKE ME TO CALL IT:

Once the program has confirmation of the action required, the only further response required of the user is confirmation that any superseded characters are in fact intentional. This confirmation will be requested with:

CONFIRM SUPERSEDING ASCII CODE nnn:

If the user does not confirm, the program will exit to allow the user to carry out appropriate action with CHRGEN.

Once processing is complete the program will give instructions for insertion of the table in CISPAC.

7.3 Checking a Font

Once a font table has been generated and inserted in CISPAC it may be checked as follows:

(i) Load CISPAC and TSTFNT

(ii) Start the program

(iii) To the '*' type the font number

(iv) Check the character set displayed on the screen

(v) To the next '*' type another font number, or type ESCAPE to exit.

ACKNOWLEDGMENTS

The subroutine package CISPAC and the programs CISTXT and CISPLA were written by D. K. Warne, and are based on similar programs developed by B. D. L. Neil for use with the A.R.L. DECsystem-10 type 338 refreshed interactive graphics display. Hardware design of the CIS 1 system was done by D. K. Warne in collaboration with T. Kinsella and H. A. Thelander. T. Kinsella revised CISPAC, CISTXT and CISPLA extensively, and H. A. Thelander also modified them, and D. K. Warne's PDP-11/20 program CIS11, to make them compatible with the FORTRAN-10 compiler and his Hybrid Computing system software H3PAC. L. N. Lester wrote and documented the program CHRGEN.

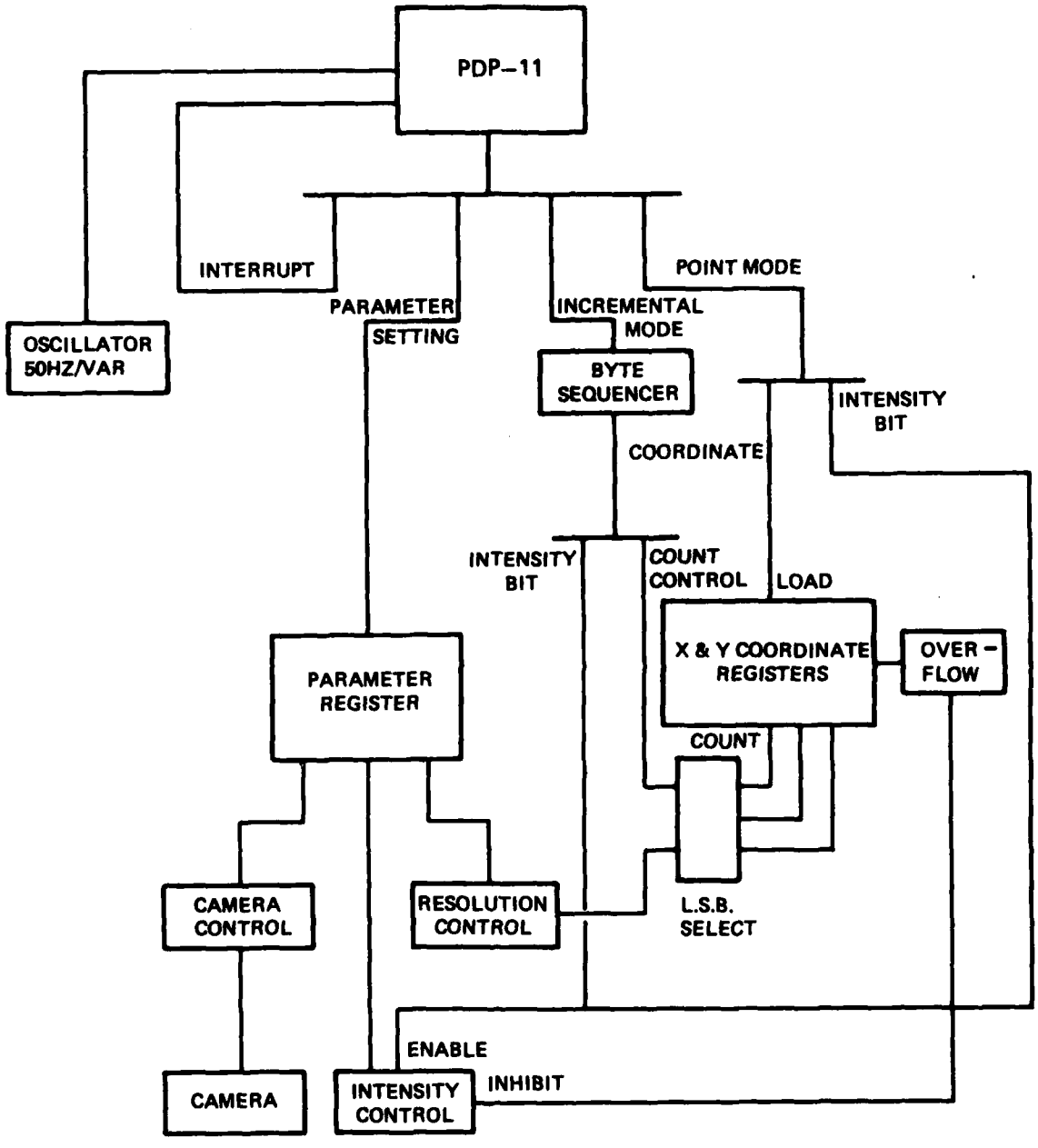


FIG. 1 CIS 1 DISPLAY CONTROLLER – SCHEMATIC

OUTPUT

INCREMENTAL MOVE

15	14 – 10	9 – 5	4 – 0
0	1ST MOVE	2ND MOVE	3RD MOVE

MOVE	1ST	2ND	3RD	
	14	9	4	COUNT X COORDINATE
	13	8	3	0=RIGHT, 1=LEFT
BITS	12	7	2	COUNT Y COORDINATE
	11	6	1	0=UP, 1=DOWN
	10	5	0	INTENSIFY

SET COORDINATE

15	14	13 – 12 – 11 – 10	0
1	0		COORDINATE VALUE

BIT 13 0=SET X, 1=SET Y
BIT 12 INCREMENT OTHER COORDINATE
BIT 11 INTENSIFY

SET PARAMETERS

15	14	13	12 – 7	6	5	4	3	2	1	0
1	1	0					I		S	

BIT 6 ADVANCE FILM
BIT 4 OPEN SHUTTER
BIT 5 CLOSE SHUTTER
BITS 2&3 SET INTENSITY TO MAX * (I+1)/4
BITS 0&1 SET RESOLUTION TO 2^S RASTER UNITS

REQUEST INTERRUPT

15	14	13	12 – 0
1	1	1	REASON (IGNORED BY HARDWARE)
1	1	1	REASON (IGNORED BY HARDWARE)

2 CONSECUTIVE WORDS ARE REQUIRED.

INPUT

PARAMETERS

15	14	13	12	7	6	5	4	3	2	1	0
1	1	0	0	F	0	0	0	I		S	

CURRENT SETTING OBTAINABLE BY READING DR11-A INPUT TRANSMITTERS
BIT 7=F=MICROFILM SUPPLY FLAG = 1 IF ≥1 METRE (APPROX) OF FILM REMAINS

LAST OUTPUT OBTAINABLE BY READING DR11-A DATA BUFFER REGISTER

FIG. 2 CIS 1 DISPLAY CONTROL DATA FORMATS

DISTRIBUTION

	Copy No.
AUSTRALIA	
Department of Defence	
Central Office	
Chief Defence Scientist	1
Deputy Chief Defence Scientist	2
Superintendent, Science and Technology Programs	3
Joint Intelligence Organisation	4
Defence Library	5
Document Exchange Centre, D.I.S.B.	6–23
DGAD (NCO)	24–27
Aeronautical Research Laboratories	
Chief Superintendent	28
Library	29
Superintendent — Systems Division	30
Divisional File — Systems	31
PO/FSG	32–37
Authors: D. K. Warne	38
T. Kinsella	39
H. A. Thelander	40
Materials Research Laboratories	
Library	41
Defence Research Centre, Salisbury	
Library	42
Central Studies Establishment	
Information Centre	43
Engineering Development Establishment	
Library	44
RAN Research Laboratory	
Library	45
Navy Office	
Naval Scientific Adviser	46
Army Office	
Army Scientific Adviser	47
Royal Military College Library	48
R.M.C., Mr M. Smith	49
Air Force Office	
Aircraft Research and Development Unit, Scientific Flight Group	50
Air Force Scientific Adviser	51

END

DATE
FILMED

11-81

DTIC