1049 AY

Bell Laboratories

Cover Sheet for Technical Memorandum

Title- GLANCE Terminals on UNIX Time-Sharing

Date- March 3, 1975

TM- 75-1352-3

Other Keywords- Graphics, Editing, Plotting, Document Preparation

Author(s)	Location and Room	Extension
Lycklama, H.	MH 7C-211	6170

Charging Case- 39394

Filing Case- 39394-11

ABSTRACT

GLANCE terminals (1) have been interfaced to the PDP-11/45 computer (2) and programmed to run under the UNIX time-sharing system both as "login" terminals and as general-purpose graphic devices. Each terminal has two local display memories consisting of 2K 16-bit words each. One memory is a character memory, which may be loaded with any character set which the user wishes to use; the other is a display list memory which controls the display sequence. The display list may contain vectors as well as characters. The terminals are currently being used for the following purposes:

(1) editing of text files using a simple line-editor

(2) display of "TPLOT-like" plots (hard copies may be obtained by a connection to STARE on the HIS-6070 computer)

(3) scanning of ASCII file contents (backward and forward)

(4) preparation of documents using "NROFF" and a paginating GLANCE "filter" program

(5) display of user-definable character sets

This document describes the characteristics of the GLANCE terminal in enough detail to enable users to write their own application programs. Currently available application programs are described to serve both as a user's guide for the GLANCE terminal and as a guide for writing further programs. Enhancements to some of the application software described here are currently being programmed.

Pages Text 25 Other 5 Total 30

No. Figures 2 No. Tables 0 No. Refs. 13

DISTRIBUTION
(REFER GEI 13.9-3)

COMPLETE MEMORANDUM TO

CORRESPONDENCE FILES

OFFICIAL FILE COPY
PLUS ONE COPY FOR
EACH ADDITIONAL FILING
CASE REFERENCED

DATE FILE COPY
(FORM E-1328)

10 REFERENCE COPIES

ALLES,HAROLD G
ANDERSON,ROBERT V
ANDERSON,WILLIAM A
*ARDIS,R B
BAYER,DOUGLAS L
BILINSKI,D J
BOYD,GARY D
BREECE,HARRY T III
BROWN,W STANLEY
BUCHSBAUM,S J
CAMLET,J V JR
CHRISTENSEN,C
CLOGSTON,A M
CONDON,J H
CONNOLLY,C V
CUTLER,C CHAPIN
DICK,GEORGE W
DOLAN,MRS MARIE T
FISCHER,W C
FREENY,S L
GILLETTE,DEAN
GIORDANO,PHILIP P
GOLDSTEIN,A JAY
HAGELBARGER,D W
HAMMING,R W
HANNAY,N B
HASKELL,BARRY G
HAUSE,A D
KAISER,J F
KAMINSKI,WILLIAM
KUBIK,P S
LARSEN,ARTHUR B
LIMB,J O
LOZIER,JOHN C
LYCKLAMA,HEINZ
MALTHANER,W A
MC ILROY,M DOUGLAS
MCDONALD,H S
MILLER,S E
MORGAN,S P
NINKE,WILLIAM H
PATEL,C K N
PINSON,ELLIOT N
PLAUGER,P J
*PRIM,ROBERT C
ROBERTS,CHARLES S
SLICHTER,W P
SWARTZWELDER,JOHN C
TERRY,M E

COMPLETE MEMORANDUM TO

TEWKSBURY,S K
THOMPSON,JOHN S
THURSTON,R N
TILLOTSON,L C
VAN LAAR,MRS A
WELLER,DAVID R
YAMIN,MRS E E
YOUNG,JAMES A
57 NAMES

COVER SHEET ONLY TO

CORRESPONDENCE FILES

4 COPIES PLUS ONE
COPY FOR EACH FILING
CASE

ABRAHAM,STUART A
ACKERMAN,A F
AHO,A V
AHRENS,RAINER B
ALBERTS,BARBARA A
ALCALAY,DAVID
ALLEN,JAMES R
ALLEN,R C
ALT,MISS DOROTHY L
AMOSS,JOHN J
AMRON,I
ANDERSON,MS K J
ARCHER,N P
ARNDT,DENNIS L
ARNOLD,GEORGE W
ARNOLD,S L
ARNOLD,THOMAS F
ARTHURS,EDWARD
ATAL,B S
BADURA,DENNIS C
BAKER,BRENDA A
BALDWIN,G L
BALDWIN,GARY L
BALLARD,EDWIN D JR
BARISH,B T
BARTELT,J L
BARTLETT,WADE S
BASEIL,RICHARD J
BAUER,MS H A
BAUGH,C R
BECKER,R A
BECKETT,J T
BELL,R E
BENJAMIN,O CONNELL J
BERGLAND,G DAVID
BERING,D E
BERNSTEIN,LAWRENCE
BERRANG,JAMES E
BERTH,R P
BEYER,JEAN-DAVID
BIAZZO,MARTIN R

COVER SHEET ONLY TO

BICKFORD,N D
BILLINGTON,MISS M J
BILOWOS,RICHARD M
BIRCHALL,R H
BIREN,MRS IRMA B
BISHOP,MISS V L
BLINN,JAMES C
BLUE,J L
BLUM,MRS MARION
BLY,JOSEPH A
BOCKUS,R J
BODEN,F J
BOHACHEVSKY,I O
BOHNING,J R
BOLSKY,MORRIS I
BONANNI,LORENZO E
BORKIN,S A
BOSWORTH,R H
BOULIN,DAVID M
BOURNE,STEPHEN R
BOWEN,EDWARD G
BOWERS,J L
BOWYER,L RAY
BOYCE,K J
BOYCE,W M
BOYLE,GERALD C
BRAINARD,RALPH C
BRANDT,RICHARD B
BRANTLEY,C L
BRENNAN,RICHARD J
BROOKS,RICHARD D
BROWN,COLIN W
BROWN,EARL F
BROWN,WILLIAM R
BULFER,ANDREW F
BULLEY,RAYMOND M
BURROWS,T A
BUTLETT,D L
BUTZIEN,PAUL E
BYRNE,EDWARD R
CABLE,GORDON G JR
CAMPBELL,J H
CAMPBELL,STEPHEN T
CANADAY,RUDD H
CANDY,JAMES C
CARAWAY,R E
CARDOZA,WAYNE M
CAREY,J H
CARLSON,C A
CARLSON,MRS H V
CARRAN,J H
CARROLL,J DOUGLAS
CARR,DAVID C
CASEY,JOSEPH P
CASPERS,MRS BARBARA E
CASTELLANO,MRS M A
CAVINESS,JOHN D
CHAMBERS,J M
CHAMBERS,MRS B C
CHANDRA,R
CHANG,MS J J V

COVER SHEET ONLY TO

CHAPMAN,W P JR
CHASZK,MARVIN B
CHAWNER,JOHN K
CHEN,EDWARD
CHEN,STEPHEN
CHERNAK,JESS
CHERRY,MS L L
CHIANG,T C
CHICK,ARTHUR J
CHIN,GEN M
CHODROW,MARK M
CHRIST,C W JR
CLAYTON,D P
CLIFFORD,ROBERT M
CLOUTIER,J E
COBEN,ROBERT M
COHEN,HARVEY
COHEN,HARVEY S
COKE,MISS E U
COLDREN,LARRY A
COLE,LOUIS M
COLLIER,ROBERT J
COOK,D D
COOK,THOMAS J
COOPER,A E
COPP,DAVID H
COSTELLO,PETER E
COSTON,WALTER P
COULTER,J REGINALD
COURTNEY PRATT,J S
CRAGUN,D W
CRUME,LARRY L
CRUM,TED A
CUNNINGHAM,STEPHEN J
D ANDREA,MRS LOUISE A
DAVIS,D R
DE GRAAF,D A
DE JAGER,D S
DENTON,R T
DESENDORF,JUDITH
DESMOND,J P
DEUTSCH,DAVID N
DEVLIN,MRS SUSAN J
DI PIETRO,RICHARD S
DICKMAN,B N
DIETZEL,K ROBERT
DIMMICK,JAMES O
DIVINE,C H
DOLOTTA,T A
DOMBROWSKI,F J
DOUGHER,J P JR
DOUGHTY,DAVID W
DOWD,PATRICK G
DRAKE,MRS L
DRAPER,DON R
DUDLEY,MRS E H
DUFFY,FRANCIS P
EDDY,MICHAEL R
EDELSON,D
EDMUNDS,T W
EIGEN,D

COVER SHEET ONLY TO

EILENBERGER,ROBERT L
EITELBACH,DAVID L
ELLIOTT,R J
ELY,T C
EMMOTT,J T
ERDLE,K W
ERNST,P A
ERRICHIELLO,PHILIP M
ESSERMAN,ALAN R
ESTOCK,R G
FABISCH,MICHAEL P
FARLOW,C W
FAULKNER,R A
FEDER,J
FELDMAN,STUART I
FELS,ALLEN M
FERIDUN,K K
FIGLIUZZI,MISS M E
FISCHER,H B
FLANAGAN,J L
FLEISCHER,HERBERT I
FLYNN,JOHN F
FLYNN,MS M L
FORTNEY,MRS VIRGINIA J
FOSS,JOHN W
FOUNTOUKIDIS,A
FOWLER,BRUCE R
FOWLER,C F
FOWLKES,E B
FOX,PHYLLIS
FOY,J C
FRANK,H G
FRANK,MISS A J
FRANK,RUDOLPH J
FRASER,A G
FREEDMAN,M I
FREEMAN,K GLENN
FREEMAN,R DON
FREIDENREICH,MRS B
FRETWELL,LYMAN J
FROST,H BONNELL
FUCHS,EDWARD
FULTON,ALAN W
GABBE,JOHN D
GALVIN,M F
GARCIA,R F
GARDINIER,LEE R
GATES,G W
GAY,FRANCIS A
GEARY,M J
GEPNER,JAMES R
GERARD,ALLAN
GERGOWITZ,MRS ELISE B
GERSHO,ALLEN
GEYLING,F T
GIBB,KENNETH R
GILBERT,G W
GILBERT,MRS HINDA S
GIMPEL,JAMES F
GITHENS,JOHN A
GLASER,W A
...
727 TOTAL

* NAMED BY AUTHOR

> CITED AS REFERENCE SOURCE

MERCURY SPECIFICATION..

COMPLETE MEMO TO:
135-DPH 13-DIR 11-EXD 15-EXD 16-EXD 135-SUP 127-SUP 1352 1356

COVER SHEET TO:
823-SUP 915-SUP 127 135
COHAGT = COMPUTING/HARDWARE/GRAPHIC TERMINALS
COPDIO = COMPUTING/PROGRAM DEVELOPMENT, MAINTENANCE/INPUT-OUTPUT
COPLGR = COMPUTING/PROGRAMMING LANGUAGES/GRAPHICS
COPLTP = COMPUTING/PROGRAMMING LANGUAGES/TEXT PROCESSING, EDITING
UNED = UNIX/EDITING AND DOCUMENT PREPARATION

RADY,J E
MH 7B201

TM-75-1352-3
TOTAL PAGES 20

TO GET A COMPLETE COPY:

1. BE SURE YOUR CORRECT ADDRESS IS GIVEN ON THE OTHER SIDE.
2. FOLD THIS SHEET IN HALF WITH THIS SIDE OUT AND STAPLE.
3. CIRCLE THE ADDRESS AT RIGHT. USE NO ENVELOPE.

PLEASE SEND A COMPLETE COPY TO THE ADDRESS SHOWN ON THE OTHER SIDE
NO ENVELOPE WILL BE NEEDED IF YOU SIMPLY STAPLE THIS COVER SHEET TO THE COMPLETE COPY.
IF COPIES ARE NO LONGER AVAILABLE PLEASE FORWARD THIS REQUEST TO THE CORRESPONDENCE FILES.

Bell Laboratories

subject: GLANCE Terminals on UNIX Time-Sharing

date: March 3, 1975

from: H. Lycklama

TM-75-1352-3

Memorandum for File

Introduction

This memo describes the pertinent features of a general purpose graphics terminal interfaced to the PDP-11/45 computer and programmed to run under the UNIX operating system as a general "login" terminal and a general-purpose graphics device. This particular terminal, the GLANCE scope, has been used for a number of years on the 516 TSS systems at Bell Laboratories (3). Included in the various user application programs which have been written for the 516 TSS system are a display editor (4) and a general purpose plotting facility (5). The GLANCE terminals are interfaced to the Honeywell DDP-516 computer through a serial I/O loop (6) which permits one to station the terminals at remote locations up to 1000 feet apart.

The loop has recently been re-designed (2) and interfaced to the PDP-11 family of computers through the DR11-C (7). The loop can support up to 63 nodes. It now supports two GLANCE terminals as well as four PDP-11 mini-computers on the PDP-11/45 computer in Dept. 1352. Some of the software packages which have been written for 516 TSS have being re-written to run under the UNIX time-sharing system. The GLANCE terminal has other capabilities than those associated with a tty terminal and these have been and will be further exploited.

This document contains information on the GLANCE terminal itself as well as a user's guide to the application software available. The following aspects are discussed:

(1) details of the I/O loop to which the GLANCE terminals are attached (from a user's point of view)

(2) capabilities of the GLANCE terminal itself

(3) software written to support the GLANCE terminal under the UNIX time-sharing system as a general "login" terminal.

(4) a user's guide to the application programs written to support the development of further software for the GLANCE terminal.

(5) future plans for the GLANCE terminal.

A note on nomenclature is included here. Five types of "commands" are described in this document. First, there are "loop commands" which control the operation of the I/O loop and its interface to the computer. Second, there are "node commands" which control the operation of a node on the I/O loop (a node is the interface between the I/O loop and a terminal). Third, there are "terminal commands", which instruct the terminal to perform some action. Fourth, there are "display commands", which are the data in the display memory which determine the display sequence. Fifth, there are "system commands" which may be issued by a program running under the UNIX operating system. An effort has been made to use the appropriate command labels whenever the type of command under consideration is not clear from the context.

High Speed Serial I/O Loop

The serial input-output loop has been designed as an extended I/O bus for connecting devices to the DEC PDP-11 UNIBUS. The loop is capable of driving various devices up to 1000 feet apart along a single coaxial cable. Each device is interfaced to the loop via a send/receive station at a "node". Parity generation and checking is provided at each node. The loop will support up to 63 nodes.

Messages sent around the loop consist of a header (8 bits), data (16 bits) and some parity bits. A message requires about 12 micro-seconds to go around the loop. A comparison of the message sent with that received or a check of the parity status bits will determine whether an error occurred during transmission. When the loop is not being used for data transmission it is placed in an automatic polling mode, which repeatedly scans the nodes for a service request (interrupt). Upon receiving a service request, the automatic polling terminates and the computer may read the requesting node number from the loop interface.

Interactions with the loop are of four types, i.e. four different "loop commands" may be loaded in the header which consists of 2 bits of opcode and 6 bits of node identification number. Each such interaction involves sending and/or receiving a message on the loop. The four "loop commands" are:

00 write command
01 read status
02 write data
03 read data

The use of these commands are covered in a later section. The following four mask bits are associated with each node:

0001 awake interrupt enabled
0010 write interrupt enabled
0100 read interrupt enabled
1000 status interrupt enabled.

These bits determine which loop transactions will cause an interrupt on the loop. For each device on the loop the appropriate command will determine which interrupts are enabled at any given time. By reading the status of the node at interrupt time, one can determine the reason for the interrupt. The 16-bit status word for the node is structured as follows (LSB to MSB):

0 - 5 device name
6 - 7 which bits are enabled
8 - 9 reason for interrupt
10-15 device status.

The device name is a unique 6-bit code for each type of device on the loop. The device status bits are unique for each device on the loop. However, bit 15 is always up for "power on" at the node. The read and write data "loop commands" receive from or send to the node a 16-bit data word. One may read the status of the loop interface after a transaction to determine whether there was an error in transmission. This status word is to be distinguished from the status of the node itself. The reader is referred to reference 2 for further details.

GLANCE Terminal

The GLANCE terminal is a general purpose graphics terminal consisting of a display screen 9 inches wide and 10.5 inches high with a keyboard and 8 buttons located directly below the display screen. It is interfaced to a node on the I/O loop. The terminal itself has two fast-access read/write memories (2K 16-bit words each) one of which contains character definitions, and the other contains the display list from which the display is refreshed.

The GLANCE memories may be written or read under control of a 16-bit terminal command word, which contains an address in the 12 most significant bits and a command opcode in the 4 least significant bits:

terminal command	
05	write memory 1 starting at address
07	write memory 2 starting at address
015	read memory 1 starting at address
017	read memory 2 starting at address

Any one of these terminal commands is followed by one or more read data or write data node commands, as appropriate. The address register of the memory being accessed is automatically incremented after each read or write to that memory. The display is not running while commands or data are being transmitted to or from the terminal. To start the display, a "write memory 1" command is issued which contains the address of the start of the display list. Then a special "start display" command (0221) is issued to the terminal. The display begins in character mode(see below).

The X(horizontal) and Y(vertical) display registers each contain 12 bits, which implies a display raster of 4096 x 4096 points. However, the deflection hardware is normally adjusted to a scale of 256 raster points per inch centered such that x=896 (01600) is at the left edge, x=3200 (06200) is at the right edge, y=512 (01000) is at the bottom edge, and y=3200 (06200) is at the top edge. As this implies, positive X is to the right and positive Y is up.

Memory 1 contains the display list which determines the format of the picture on the display screen. During the display sequence, the display commands are interpreted according to one of three modes: control, short vector or character. There are 16 control-mode display commands. Each command consists of a 4-bit opcode (low 4 bits) and 12 bits of data. The formats are as follows:

OPCODE	DATA	
00	0	escape to character mode
01	jump address	jump and wait for 60 Hz. sync signal
02	jump address	jump
03	0	goto short vector mode
04	+/- delta X (invisible)	manhattan vectors
05	+/- delta Y (invisible)	
06	+/- delta X (visible)	
07	+/- delta Y (visible)	
10	X (invisible)	load registers absolute
11	Y (invisible)	
12	(unused)	

13	Y (visible)
14	brightness,scale,blink set parameter register
15	wait count (invisible) wait 250ns per count
16	wait count (visible)
17	(unused)

The brightness parameter varies from 0 (brightest) to 7 (least bright). The scale parameter varies from 0 (not displayed) to 15 (maximum size) and affects the vector and increment display commands only. The blink parameter varies from 0 (no blink) to 7 (maximum blink rate). The reader is referred to reference (9) for further details.

In short vector mode the 16 bits in a display command have the following meaning (from least significant bit to most significant bit):

bit(s)	content
0-5	delta Y
6	+/- (Y), (sign, magnitude)
7-12	delta X
13	+/- (X)
14	visible/invisible (1/0)
15	change to control mode (1)

In character mode the 16-bit display command consists of two 8-bit characters. The 8 bits are used as the address of a function in memory 2 (locations 0-0377) which determines how the character is to be displayed. By convention, the character DC1 (021) is coded so as to change from character to control mode.

Memory 2 contains a character look-up table in locations 0 to 377(octal). This table contains a function word for each character. The function word consists of a 4-bit opcode (least significant bits) and 12 bits of data as defined below:

OPCODE	DATA
16	pointer to incremental display code
00	0; null, fetch next character
01	0; DC1, set to control
17	pointer to control mode display code

The incremental mode in memory 2 treats the list as a compact string of 2- and 4- bit bytes (8,9). The following binary codes are used:

10	increment x
01	increment y
11	increment x and y
0000	reverse direction of x
0010	reverse direction of y
0011	reverse beam intensification (on or off)
0001	end of character.

The control mode in memory 2 treats the list as a set of 16-bit words which are similar in format to the control commands in memory 1:

OPCODE	DATA	
00	0	escape (fetch next ascii char.)
01	0	(unused)
02	0	(unused)
03	0	(unused)
04	+/- delta X (invisible)	manhattan vectors
05	+/- delta Y (invisible)	
06	+/- delta X (visible)	
07	+/- delta Y (visible)	
10	X (invisible)	load registers absolute
11	Y (invisible)	
12	0	(unused)
13	Y (visible)	
14	brightness,scale,blink	set parameter register
15	wait count	wait invisible (250ns per count)
16	0	(unused)
17	0	(unused).

In the above discussion, the size of an increment is determined by the "scale" parameter (1 to 15 raster points). At the start of a character display, the increment directions are set positive (right and up), and the beam intensification is turned off. To display a dot, the beam intensification must be turned on and then an increment command must be given, causing the beam position first to move and then display a dot.

The GLANCE terminal has a keyboard associated with it, serving as the means for inputting characters. In addition to the keyboard, there are eight buttons located directly below the display screen which may be used for input as well. These eight buttons are assigned the octal codes, 030 to 037.

GLANCE I/O Driver

The GLANCE terminal is treated as a general "login" terminal under UNIX, i.e. as a general "tty" character device. Text is displayed on the screen in "scroll" mode and a "line editor" is built into the driver using the labelled buttons on the bottom of the scope screen. Each GLANCE terminal has a "tty" file associated with it as well as a "gl" file with general read/write permissions for accessing the GLANCE memories. At start-up time, the "init" process in UNIX executes "/etc/getgl" for GLANCE terminals instead of the standard "/etc/getty" for regular "tty" terminals. This program proceeds to load up memory 2 with the standard character set as defined in "/etc/cset", initializes memory 1, starts the display and then waits for the user to log in.

The I/O driver for GLANCE is a character device driver using the system library routines to pass characters to and from the user. The "tty" library routines are also used to perform the appropriate character queueing tasks. A "glsgtty" routine sets and gets the modes of the GLANCE terminal in much the same way as for other "tty" terminals. One word in the three word buffer passed to "glsgtty" is used to set the flag bits for the terminal, which have the same meaning as for "tty" terminals (10). However the other two words have different meanings:

```
stty    1 3 state bits
        2 not used
        3 tty flag bits


gtty    1 3 state bits
        2 node status
        3 tty flag bits
```

where the state bits are defined as follows:

```
0       0 - write data directly to GLANCE
        1 - treat terminal as a tty terminal and
          let scrolling driver display the characters
1       0 - read from GLANCE memories
        1 - read from keyboard
2       0 - write command mode
        1 - write data mode.
```

To write 100 bytes of data into memory 1 starting at address 040, the following sequence of commands in the C language may be issued:

```
stty(1,cmode);
write(1,&command,2);
stty(1,dmode);
write(1,buf,100);
```

where:

```
int cmode[]{0,0,0};
int dmode[]{4,0,0};
int command 01005; /* address 040 in 12 most significant bits */
```

In normal tty mode the flags are set to 0333 and the control buttons at the bottom of the screen (030-037) are not treated as text input, but act as line edit control buttons. However, putting the terminal in raw mode or no echo mode, one can read these control buttons directly.

Line-Editing Features

The line-editing features of the GLANCE terminal are always active when the terminal is in normal "tty" mode and being used as any other "tty" control terminal. Characters input by a user are displayed on the lowest line of text on the display screen just above the control button labels. A rectangular blinking cursor indicates where the user's next character will be displayed. When a "line-feed" or a "carriage-return" character is input, the display of text is moved up one line, the old top line (if any) is lost and the cursor appears at the beginning of the new lowest text line. Up to 48 lines of 80 characters may be displayed on the display screen at any one time.

The line-editing features are active not only in the UNIX text editor but also when any command line or data input line is being entered at the terminal keyboard. The eight edit control buttons just below the screen are labelled at the bottom of the screen as follows:

BEGLINE DELCHAR DELLINE <-- --> INSERT REMOVE ENDLINE

Four of the buttons are used as cursor control buttons. These buttons may be used to move the cursor left or right leaving the contents of the user's input line unchanged. The BEGLINE and ENDLINE buttons move the cursor to the beginning and end of the user's input line, respectively. The middle two buttons, "<--" and "-->", move the cursor one character left and right, respectively. The DELCHAR and DELLINE buttons perform the analogous functions of the UNIX editing characters, "#" and "@", respectively. The difference is that the user's input character(s) is(are) actually removed and the cursor is backed up by the appropriate number of characters. The DELCHAR button always deletes the last character of the user's input line which is not necessarily the one to which the cursor points. The other two control buttons are used to insert and remove characters at the current cursor position. The INSERT button leaves the cursor at the current position but creates a one character gap in the user's input line each time it is pushed. This enables the user to insert characters in the middle of his input line without having to erase the whole line. The REMOVE button on the other hand removes the character at the present cursor position and fills in the gap so created by moving all others characters to the right of the cursor one position to the left. This allows the user to selectively remove a character anywhere in his input line without having to erase the whole line. The TAB character is treated as a special character in that when it is input, the cursor is moved to the next multiple of eight character positions from the beginning of the line. The TAB character is passed up to the user in his input buffer in any case. Upon removing the TAB character, the variable length gap is filled in appropriately.

Character Set Definition, Generation and Loading

A format has been specified to create character definitions for GLANCE memory 2. The format definition for one character is a line of text that begins with either an octal number or an ascii character followed by an optional number of spaces and terminated by a colon. The colon must be followed by a definition control character:

```
LF      - no definition ( null line )
C       - control mode definition list
E       - set to control mode for memory 1
space   - incremental mode definition list
```

The definition of the character follows the definition control character. If the definition requires more than one line, a back slash is used to escape the newline character and the definition continues on the next line. The codes may appear in any arbitrary order. An example of the character (014) defined as a form feed to position the beam in the upper left hand corner of the screen could be done in control mode as:

```
014:C 40010 140011 2015 0.
```

To define a character in incremental mode requires giving the proper sequence of 2-bit codes. These codes may be specified as a sequence of numerical codes(base 4) or as a sequence of single letter codes (one letter per incremental code byte). The following 7 codes are used:

CHAR	CODE(base 4)	MEANING
x	2	move X one increment
y	1	move Y one increment
d	3	move X and Y one increment each (diagonal)
h	00	reverse direction of X
v	02	reverse direction of Y
b	03	reverse beam (off->on or on->off)
e	01	end of character definition

The following example shows how all this works. At the beginning of displaying a character, the beam is off and the direction of both X and Y increments is positive. Let the beam be positioned at the top left hand corner of the character matrix, in this case 6x9 according to our standard format. To display a dot, one must turn the beam on before moving the beam to that dot. The letter 'T' can be defined in any of the following ways:

```
T   : 2032 2222 0300 0220 3311 1111 1030 0023 3331 1110 1
0124: 2032 2222 0300 0220 3311 1111 1030 0023 3331 1110 1
T   : xbxxxxxbhvxbdyyyyyyybhvddddyyyye
0124: xbxxxxxbhvxbdyyyyyyybhvddddyyyye
```

Notice that the beam has been left at the top left hand corner of the next character matrix. For non-standard characters, the user is free to leave the beam in any position he desires.

Having defined the character set one must now generate the character set into a form which can be directly loaded into the character memory. Four different programs have been written to generate and load the character sets:

```
mcs cdef cset
lcs cset
```

```
gcs -i cset
wcs cdefl [cdefn]
```

The first one "mcs cdef cset" will read the character definitions from the file "cdef" and generate a character set in file "cset" which may be loaded directly into character memory on the GLANCE terminal. This program involves no direct reading and writing of the GLANCE memories itself. To load the character set "cset" into memory 2 from one's own GLANCE terminal, one may invoke the program "lcs cset". To load the character set "cset" into memory 2 of an arbitrary GLANCE terminal, one may invoke the program "gcs -i cset", where the option "-i" specifies which GLANCE terminal is being loaded. Sometimes it is desirable to re-define only a few characters or to define a few new characters in the character set, keeping the standard character set as well. This may be done by invoking the program "wcs cdef" to rewrite only part of the character memory. The file "cdef" is in the same format as discussed before. Since the standard character set does not use up any more than 1024 words of the character memory, the new character definitions are written into character memory starting at word 1024.

GLANCE File Scanner

Since a GLANCE terminal in normal "tty-like" mode displays just 48 lines of text, it is not possible to "print" a long file in one operation and see anything except the final 48 lines. Therefore a program "gfs" (GLANCE file scanner) was written to allow one to scroll through a file. Upon invoking "gfs fname", the first 48 lines of the file are displayed and the eight control buttons at the bottom are labelled as follows:

BEGFILE SCREENUP LINEUP MIDFILE LINEDN SCREENDN ENDFILE DONE

BEGFILE, MIDFILE and ENDFILE will position the 48-line display window to the beginning, middle and end of the file, respectively. SCREENUP and SCREENDN will display the next 48-line window in the file above and below the present location respectively. LINEUP and LINEDN move the display window one line up and down, respectively. One can use the repeat key to scroll the text in either direction quickly. This program enables one to scroll through a file very quickly. Spaces and tabs are converted to special characters to reduce flicker in the display.

GLANCE Filter Program

The file scanner just described is not useful when the file output must be processed before being displayed. A GLANCE filter program has been written to paginate the output so that only a single "page" (66 lines) is displayed on the screen at one time. The NEXT, DONE and STARE control buttons are designated at the lower edge of the display. The "next" page of the output may be displayed by pushing the NEXT button below the bottom of the screen. The DONE button will terminate the program. A hard copy of the "page" may be obtained by pushing the STARE button. This hard-copy feature is discussed in the next section. The GLANCE filter program may be invoked to list one's files either as:

gl [-l nlines] [-c cset] ls -l

or as:

```
ls -l | gl [-l nlines] [-c cset].
```

The optional parameters which may be specified are "nlines" (number of lines in a page) and "cset" (a user-defined character set). The default values of these parameters are nlines = 66 and cset = "/etc/cset". One may filter the output of any program through this filter. Some other typical uses of the filter program are:

```
cat file | gl
pr file | gl
od /dev/rp0 25000 | gl -l 40
```

Perhaps the most most useful application of the GLANCE "filter" is that discussed in the next section of this document. Again flicker is reduced in the display by converting tabs and spaces to special characters.

Document Preparation

Documents are typically produced under the UNIX time-sharing system using the nroff program (11). Changes can be made to the source of the document by using the editor "ed" available under UNIX. Producing the output requires running nroff, usually on a hard-copy terminal. The time taken to realize that there is an error in the document may be substantial. Sometimes a special terminal may be required to produce the output, e.g. a Teletype 37 for Greek characters and half-line motions. The GLANCE filter program was written to simulate these special characters and the Greek character set. This enables one to obtain quickly a display of any page of the document. Errors may be corrected immediately, and the document displayed again. The turn-around time for making a correction or addition to a document is reduced substantially.

Documents to be prepared using the GLANCE filter program are prepared in the same format as those to be produced for final copy on the Teletype 37 terminal or any other suitable hard-copy terminal. Half-line motions are obtained by typing "escape sequences", i.e. an ESCAPE character followed by a digit. The following three escape sequences are recognized by the program:

```
ESCAPE7  -  produces a full space up
ESCAPE8  -  produces a half space up
ESCAPE9  -  produces a half space down.
```

Greek characters are produced, as on the Teletype 37 terminal, by the "shift out" character (CONTROLn) which enters the special character mode, and the "shift in" character (CONTROLo) which leaves it. Only the 32 characters which are produced on the Teletype 37 terminal are currently defined in the file "/etc/greek", which must be loaded by the user of the GLANCE filter program if Greek characters are to be displayed. Some examples of the use of this filter program are:

```
nroff doc | gl
nroff +6 doc.? | gl
neqn doc.? | nroff -ms | gl -c /etc/greek
```

Errors in this document were quickly corrected by "glancing" through the document on a GLANCE terminal using the GLANCE filter program. An example output is attached as Figure 1 showing half-line motions and some Greek characters.

Plotting Facilities

The fact that the GLANCE terminals have vector generation as well as character generation capabilities enables a user to plot arbitrary data sets in an interactive manner on the GLANCE display screen. A general-purpose plotting routine, GPLOT, has been written for the GLANCE terminal to allow a user to easily plot his data sets interactively, debug the plot, i.e. change his data set, and then plot the data again. When he is satisfied with the end result, he may obtain a hard-copy by means of a connection to the STARE facility on the HIS 6070 computer.

In general, the same features are provided in GPLOT as are provided by the TPLOT package (12) on the HIS 6070 computer. However, some new features are provided by GPLOT, since the user has control over the character generator in GLANCE. The features provided by GPLOT include:

- choice of linear or log scales for x and/or y
- automatic or specified selection of x and y scale factors
- automatic selection of grid lines
- choice of grid lines or grid ticks
- ability to inhibit grid lines and specify one's own
- automatic labeling of grid lines
- choice of rectangular frame or polar graph frame
- controllable plot size (both horizontal and vertical)
- control over directions of positive X and positive Y
- automatic centering of titles and labels
- plotting of multiple curves with identification
- character or dashed-line drawing
- ability to plot an arbitrary object at any data point
- ability to place a legend below graph
- ability to place labels at any point on graph.

The user's data sets must be in the following format in the input file for GPLOT:

```
LLX LLY NGL ... ( up to 20 option parameters )
TTL (Title)
STL (Sub-title)
YLB (Y-axis label)
XLB (X-axis label)
NCURVES (Number of curves)
CHAR (optional character)
NPTS (Number of data pairs)
Data set consisting of NPTS (X,Y) pairs.
   ...
```

The first line of input data contains a list of up to 20 option parameters which the user may specify to control his plot. The names of these parameters are identical to those used by TPLOT where applicable. The parameters (LLX LLY NGL ARX ARY YREF LABOP) specify the grid selection options. The parameters (KO XMIN XMAX YMIN YMAX) specify the scaling options. The parameters (KX KY ITYCUR NTT) specify the curve drawing options. The parameters (USEROPT CFL URSUBR LFL) are options which may be used with GPLOT that provide enhanced plotting capabilities generally not provided by TPLOT. The meanings of these parameters are listed below for reference:

LLX	linear/log scale option X direction
LLY	linear/log scale option Y direction
NGL	grid line option, lines or ticks
ARX	X direction aspect ratio
ARY	Y direction aspect ratio
YREF	vertical reference point
LABOP	grid line labelling control
KO	scaling option, automatic or specified
XMIN	minimum X value
XMAX	maximum X value
YMIN	minimum Y value
YMAX	maximum Y value
KX	X or |X| or previous X array
KY	Y or |Y| or previous Y array
ITYCUR	connection option for data points
NTT	number of times curve is traced
USEROPT	user options

CFL   character file
URSUBR user subroutine
LFL   label file.

For complete details on the various possible values of the above listed options the reader is referred to a previous document which discusses the implementation of GPLOT on the 516 TSS time sharing system (5),(13) and to the original TPLOT manual (11).

The GPLOT program is invoked by:

gplot gdata [+goutput]

The "gdata" file contains an arbitrary number of user data sets. Upon displaying each plot on the screen, five control buttons below the screen are labelled as:

DEBUG REWIND NEXT DONE STARE

The DEBUG button initiates a display of all of the user's specified options and other parameters to enable him to "debug" his plot, i.e to determine why his plot did not turn out the way he expected it to. By pushing the NEXT button, the next graph will be displayed if he has more data in his input file. If not, the GPLOT program will terminate. Pushing the DONE button will automatically terminate the plotting of the user's data sets. By pushing the REWIND button, the first plot of the user's data will be displayed again. To obtain a hardcopy of any one of his plots, the user may push the STARE button. This will cause the contents of both GLANCE memories to be transmitted to the HIS 6070 computer to be output on the STARE facility using a program to translate from GLANCE code to STARE code.

An optional output file "goutput" may be specified if the user wishes to write the generated display list into a file as well as in the GLANCE display list memories. This allows one to come back later and glance at the output plots by means of:

glance goutput

without having to go through the calculations to produce the display list again. The "glance" program displays one plot at a time and gives the user the same control as in GPLOT by means of the labelled buttons on the bottom of the scope screen. Here display of the next plot is almost instantaneous. One can "glance" at plots quickly to observe differences between adjacent plots in the "goutput" file. Again one may push the STARE button to obtain selective hard copies. An example of such a hard copy is attached as Figure 2.

It is possible to generate displays on the scope screen other than those produced by GPLOT. In order to obtain hard copies of these pictures, one may invoke:

stare [-i]

from any terminal to produce a hard copy on the central STARE facility. Here the optional parameter "i" specifies which GLANCE terminal is being referred to. For instance, one may wish to generate arbitrarily complex pictures using the character generator, pictures which cannot easily be produced by TPLOT. This allows one to get hard copies of any display on a GLANCE scope screen.

Utility Programs

Some other general utility programs which have been written for the GLANCE terminals include:

dglm1 [-i] [address]

dglm2 [-i] [address]

glrst [-i]

glrstr

The programs "dglm1" and "dglm2" dump the contents of the GLANCE memories 1 and 2, respectively, starting at the octal word address specified (0 if no address is given). The option "-i" specifies the GLANCE terminal (user's terminal is default). The dump is accomplished by copying the contents of the corresponding memory into a file "glmem.1" or "glmem.2", resetting the terminal to "tty" mode and then proceeding to print out its octal contents. The user may then examine the contents of "glmem.1" or of "glmem.2" selectively using the "od" command in UNIX The program "glrst [-i]" will re-enable the corresponding terminal if power has been turned off or it has been otherwise disabled. The program "glrstr" restores both memories of the terminal from which the program is invoked and leaves the terminal in "tty" mode. This program is typically used after a user has redefined the character set and then wishes to return to normal "tty" mode operations using the standard character set.

Conclusions

We have shown how the GLANCE terminal is being utilized under the UNIX time-sharing system as a general "login" device and a general-purpose graphics terminal. It has proven to be a very useful terminal both as a general-purpose "tty" terminal and as a graphics terminal. Currently the I/O loop on the PDP-11/45 computer in Dept. 1352 supports 2 such terminals as well as 4 PDP-11 mini-computer terminals. More GLANCE terminals are planned in the near future. However, it has become clear that these terminals present a substantial CPU load on the PDP-11/45. Therefore the UNIBUS LINK is currently being programmed to interface a PDP-11/20 computer to the PDP-11/45 to take much of the load off of the PDP-11/45 computer and transfer it to the PDP-11/20.

Further enhancements to GPLOT are planned in the near future. With the bulk of the I/O ring and GLANCE terminal driver code in the PDP-11/20, it will now be possible to program a display text editor in the PDP-11/20 with few restrictions on size of the program and its execution speed.

Acknowledgements

The bulk of the code for GPLOT has been written by a part-time employee, Jon Sieber. The author is grateful to A. D. Hause and C. Christensen for sharing their experience with him in programming the GLANCE terminal under the 516 TSS system. The standard character set was provided by K. Thompson. D.L. Bayer is currently writing the UNIBUS LINK driver connecting the PDP-11/20 and PDP-11/45 computers. A. D. Hause has provided valuable assistance in preparing this document.

MH-1352-HL-JER H. Lycklama

Att.
References
Figures

REFERENCES

1. J.V. Camlet and D.R. Weller, "A High Speed Microprogrammed List Processor for Computer Graphics", Society for Information Display, International Symposium, June 1972.

2. D.R. Weller, "A High-Speed I/O Loop Communication System for the DEC PDP-11 Computer", TM 73-1356-8.

3. C. Christensen and A.D. Hause, "A Multiprogramming, Virtual Memory System for a Small Computer", AFIPS - Conference Proceedings, Volume 36, 1970 SJCC.

4. A.D. Hause, "A Text Editor for a CRT Computer Terminal", TM 73-1352-2.

5. H. Lycklama, "Plotting Facilities for Mini-Computer Systems", TM 74-1352-7.

6. D.R. Weller, "A Loop Communication System for I/O to a Small Multi-User Computer", MM 70-1384-1.

7. DEC DR11-C General Device Manual.

8. E.J. Sitar and L.I. Stukas, "Two-Bit Incremental GLANCE Display Software Package", MM 71-8231-6 & MM 71-1383-4.

9. J.V. Camlet, "Coding for Glance", 516-TSS Document 516-48.

10. K. Thompson and D. M. Ritchie, "UNIX User's Manual - Fifth Edition, 1974".

11. J.F. Ossanna, "NROFF User's Manual", TM 73-1271-2.

12. J.F. Kaiser and E.J. Sitar, "A Versatile Subroutine, TPLOT, for Automatically Generating Complete Graphs", MM 67-1383-1 & MM 1386-2.

13. J.E. Steinhart, "GLANCE Plotting Routines", 516-TSS Document 516-77.

class 2 ——> $[Co(III)Et_8P]^{+2}\cdot 2ClO_4^-$

From the spectra in Figure 3a one can see that class I ($^2A_{1u}$) is characterized by a major absorption peak near 700 nm with a high energy shoulder while class 2 ($^2A_{2u}$) has a nearly featureless spectrum in the 500-700 nm region.

By examination of the optical spectra of Co(II)Et_8P (see Figure 4) as it undergoes the electron oxidation, one can see that there is an initial 1e$^-$ oxidation of the red Co(II)Et_8P to $[Co(III)Et_8P]^+Br$ (red). Here there is some change in the spectrum. However, upon the further oxidation of $[Co(III)Et_8P]^+$ to $[Co(III)Et_8P]^{+2}\cdot$ (green), a porphyrin π-cation radical, there is significant change in the spectrum. This two e$^-$ oxidized species has a spectrum that resembles the Zn tetraphenylporphyrin (6), a class 2 compound. The π-cation radical species of $[Co(III)Et_8P]^{+2}\cdot$ was also confirmed by esr measurements.

The information gathered on these Co(II)Et_8P compounds was used to demonstrate that the two-electron oxidized green compounds Catalase I and horseradish peroxidase (HRP I) were in fact π-cation radicals (see Figure 4b) (6). The spectrum of HRP I is similar to $[Co(III)Et_8P]^{+2}\cdot 2Br^-$ ($^2A_{1u}$ - class I).

These cobalt compounds are of interest because the information obtained can be used to understand vitamin B_{12}. The absorption spectra of vitamin B_{12} (cobalamins) and co-

Figure 1

NEXT DONE STAPE

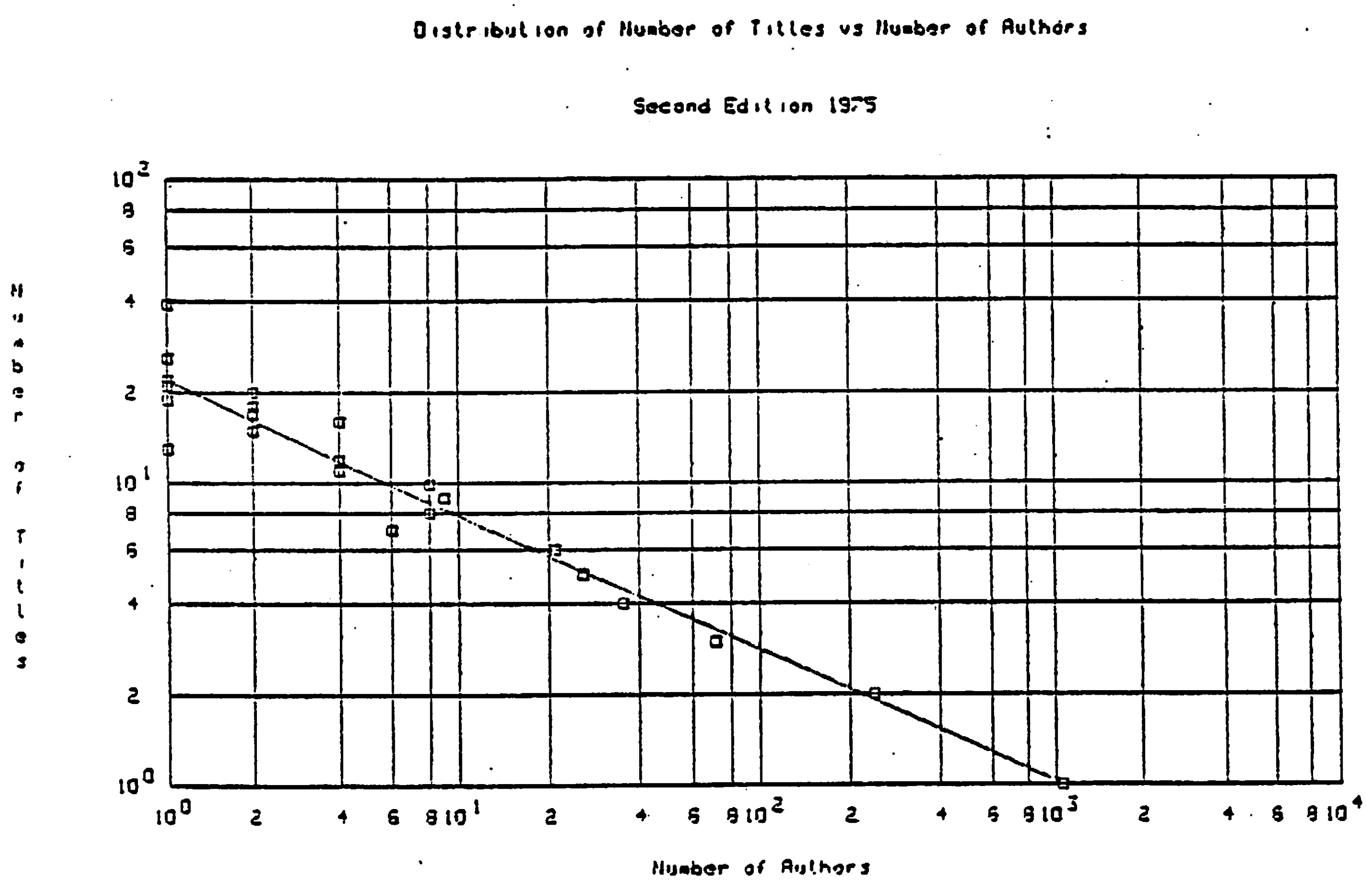
Distribution of Number of Titles vs Number of Authors
Second Edition 1975
Number of Titles
Number of Authors

Figure 2