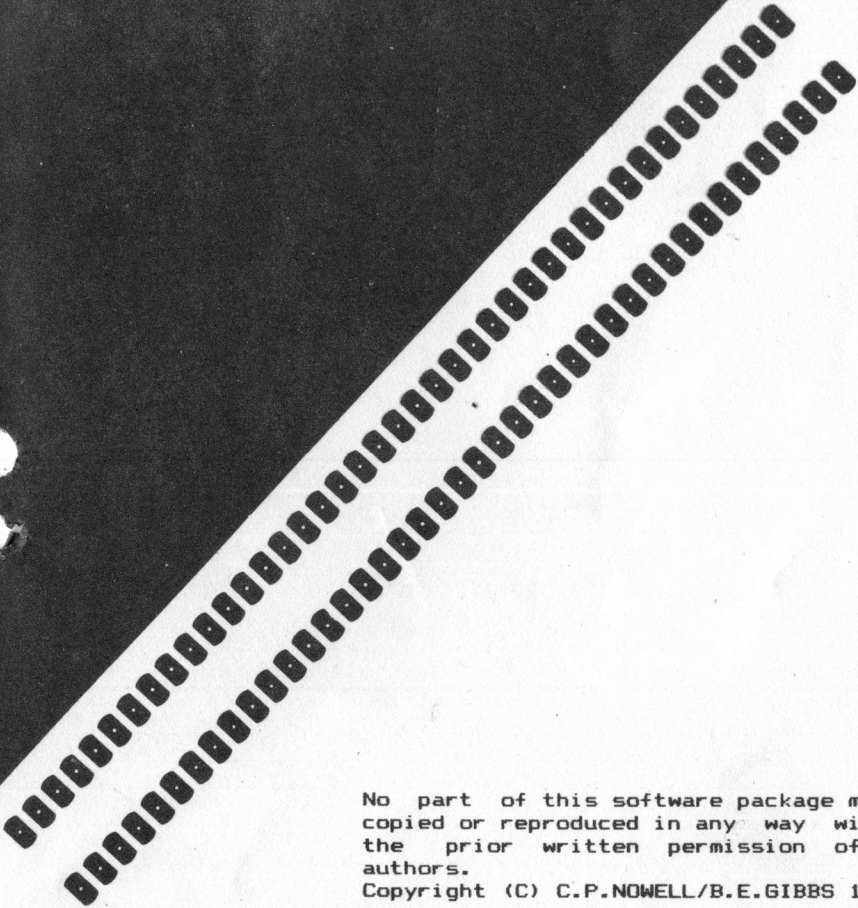


VDU TOOLKIT

(PTL)



No part of this software package may be copied or reproduced in any way without the prior written permission of the authors.

Copyright (C) C.P.NOWELL/B.E.GIBBS 1984

VIDEO TOOLKIT

Copyright Colin Nowell/Brian Gibbs 1984

INTRODUCTION :

This version has been written for use with the 80/82 Video card, incorporating the 8k Operating System VBUG 6(PTL)

RESIDENT ADDRESS :

\$E800 - \$EFFF

NOTE :

Certain alterations have been made to the original commands in version V1.0, thus programs written using this version will require editing i.e:-

#WIDTH is now #SIZE

#WINDOW is now #HEIGHT

#RENUMBER has been deleted due to lack of space.

The new #WIDTH is now used to set the window width.

BASIC EDITOR for 80/82 VDU :

It is assumed that this Toolkit is residing in Ram, if not the Video Card Basic Editor will not function as it requires a operating buffer, which is located within the Toolkit. If RAM not present then Microtan Screen must be used for editing.

On entering the Edit Mode "n" CNTL E where n = line number, the toolkit automatically sets character size 1 (80 column mode).

If using the Microtan screen CNTL U & CNTL D are no longer required as CNTL L & CNTL R follow straight through.

All other commands remain the same.

THE COMMANDS :

All commands are prefixed by a #

GRAPHICS LINEAR.

Parameters = $0 \leq x \leq 511, 0 \leq y \leq 255$

SET x,y - set a point

RESET x,y - reset a point

INSET x,y - invert a point

DRAW x,y - draw to new co-ordinates

UNDRAW x,y - erase to new co-ordinates

INDRAW x,y - invert line to new co-ordinates

MOVE x,y - move to new co-ordinates

TEST x,y - return status of point (1 = set, 0 = reset/off)

MODE x - change the plotting mode 0 = ABS, 1 = REL (although this command is implemented, entering -ve parameters is not possible yet in the above commands).

GRAPHICS ANGULAR.

ARC a,b,c,d - draw angular line

UNARC a,b,c,d - undraw angular line

Parameters for ARC and UNARC are:-

$0 \leq a = \text{start angle} \leq 360$

$0 \leq b = \text{radius} \leq 255$

$0 \leq c = \text{end angle} \leq 360$

$0 \leq d = \text{aspect ratio} \leq 255$ (12 = True)

TILT a - set origin for ARC, UNARC

$0 \leq a \leq 360$ (0 on power-up)

INC a - set step for ARC drawing

$0 \leq a \leq 360$ (8 on power-up = circle)

45 = hexagon, 90 = square, 120 = triangle etc.

CURSOR CONTROL.

(1) No Parameters.

HOME - home cursor inside window

LEFT - move one column left

RIGHT - move one column right

DOWN - move one line down

UP - move one line up

OFF - turn cursor off

FLASH - make cursor flash

ST - make cursor steady

SWAP - move text cursor to graphics cursor position (bottom left of cursor cell = graphics position)

(2) With Parameters.

CURS r,c - move text cursor to row, column

$0 \leq r \leq 24, 0 \leq c \leq 84$ (maximums depend on character size and window width)

CHARACTER CONTROL.

(1) No Parameters.

SPS - superscript = raise next character ONLY 1/2 screen line.

SBS - subscript = lower as in SPS

LU - underline control = toggle alternately on/off.
(soft toggle in \$23,35)

(2) With Parameters.

SIZE n - change character width $1 \leq n \leq 5$

CHAR u(d1,d2,d3,d4,d5,d6,d7,d8,d9,d10) = define user character
where $128 \leq u \leq 152$ & $0 \leq d1 \dots d10 \leq 255$ note the brackets syntax
(d1 = top line of character cell, d10 = bottom).

See Fig. 1.

SCREEN CONTROL.

(1) No Parameters.

CLG - clear all pixels, cursor position and window parameters secure

CLS - clear all pixels within current window and home the cursor (top left corner of window)

RVS - background colour - toggle between white on black/black on white (soft toggle in \$22,34.10).
(If #RVS:#CLS is issued, then the whole of the current window reverses image).

DLIN - delete from current cursor position to end of line

DSCR - delete from current cursor position to end of screen (current window)

(2) With Parameters.

HEIGHT rf,r1 - set height of window to between 1st. row (rf) & last row (r1) then home the cursor ($0 < rf < 23$, $1 < r1 < 24$)

Note: For cursor positioning rf now becomes 0, until a new window is set

WIDTH cf,c1 - set window width to between 1st. column (cf) & last column (c1) then home the cursor ($0 < cf < 84$, $1 < c1 < 85$)

Note: The cursor positioning applies the same as width, but maximum column will be determined by current character size.

VBUG CONTROL:

PUSH - save all VBUG'S parameters on it's internal 6502 stack.

POP - restore the above.

SPECIAL COMMANDS.

REV - reverse the printing direction to right to left/left to right.

Note: This is a toggle type command under VBUG control.

PAGE - toggle between page or scroll mode, scroll is normal on power up.

When in page mode, next line is cleared before printing on it and the cursor homes after use of the bottom line. (VBUG controlled toggle)

N.B. Both commands valid INSIDE current window.

UTILITIES.

(1) COMMAND.

SYS n - replaces Z=USR(I) but no parameter passing (0 <= n <= 65535)

(2) FUNCTIONS.

DEC(n\$) - returns decimal value of the hex STRING n\$.
 n\$ may be assigned or in the form "FC00".
 If n\$! "7FFF" it will be returned in TWO'S COMPLEMENT form
 (Blame BASIC, not me !!!!)

HEX\$(n) - returns a hex string of the value of n. Notes as
 for DEC but the range is true (16 bit). (0 <= n <= 65535)

Both these functions MUST be preceded by "PRINT" i.e
 PRINTHEX\$(2348) or PRINTDEC("FC00")

FIG. 1.

