

## THE TOOL

### Getting Started

1. Turn on your computer, disk drive, and monitor or TV.
2. If you are working in the C-128, you first need to access the 64 mode. Type GO64. Answer by typing Y and press RETURN.

A full text of instructions is included on the disk. We recommend that you print out the full set of instructions before proceeding. If you are working with a non-Commodore printer, be sure to set your printer interface to Commodore 1525 emulate mode. Follow the procedure below.

1. Type `LOAD ``README``,8`.
2. When the READY message appears, type RUN.
3. The printing module will load. The instruction files, beginning with the file named ```manual```, will print until complete.

Now that you have full instructions printed, you are ready to load the program itself.

1. Type `LOAD ``boot.tool``,8`.
2. When the ready prompt appears, type RUN.

In future sessions with the program, you can type `LOAD ``1``,8`.

### The Screen Generator

Designed to ease data acquisition on the screen, the screen generator is more reliable than BASIC and results in faster execution. Display commands help you to draw lines, columns, to scroll on the screen, etc. Acquisition commands define acquisition zones and associated controls, to allocate zone contents to BASIC variables, complete printusing, etc. Screen-page management commands include save and load screen pages from disk.

The variable names chosen in the instructions are arbitrary. Any BASIC variable would suit, aside from reserved variables utilized by the TOOL: ```zo``` and ```ok```.

The screen is defined as a drawing board with the origin in the upper left-hand corner. The origin coordinates are 1,1. Any point on the screen is defined by: l = line number, c = column number, ln = horizontal line (number of columns), lg = vertical line (number of lines).

The acquisition instructions have been designed to be used instead of INPUT and GET. A zone is defined by certain parameters such as starting point, size, allowed type of characters, possibilities of exit, and an identification number.

With TOOL, you can use similar frames or screen pages over and over again. If during an acquisition, the control command is forgotten, you can easily exchange pages for a HELP page. Simply save the actual page on disk, then load the HELP page. When the verification is done, reverse the operation, and you will find that the cursor reappears at the exact point it was before the page exchange.

### **Graphic Instructions**

This set of instructions allows charts to be drawn using the high-resolution screen with a definition of 320 by 200. The high-resolution screen is defined as a drawing board with its origin in the lower left corner. Any point in the high-res screen is defined by a horizontal and a vertical coordinate.

### **Programming Utility Aids**

These utilities enhance the capabilities of BASIC. You can autonumber or renumber programs, delete program lines, dump a listing of variables being used, locate errors in a BASIC program line, find any string of characters within a program, execute the program one instruction at a time through a trace command.

In addition, you can locate characters inside a string of characters, create and define a string of characters, use an IF THEN ELSE function, and reserve a joystick function. Lastly, the DOS support provided simplifies the operations of the disk.

target a\$ [ ,l,c]	- "clever" GET of a pre-defined char.3-16
clear l,c,lg,ln	- erases ln characters (origin point :l,c) 3-5
clear2 n	- erases the nth zone.3-15
decz n,l,c,ln[ ,ty][ ,f\$]	- declares the nth zone.3-10
inz n,a\$	- transfers the content of the nth zone into a\$3-13
:	
out a\$,l,c	- displays a\$ (origin point :l,c)3-6
outz n,a\$	- displays a\$ into the nth zone and controls a\$3-14
reqz n	- requires an answer in the nth zone3-12
rev l,c,lg,ln[ ,co]	- reverses the video mode on part of the screen3-7
sclear	- clears the screen3-20
screen sc,br[ ,cr]	- modifies the colors of the screen, the borders and the characters 3-21
scroll l,c,ln,lg,ty	- scrolls a part of the screen3-8
sload 8,"name"	- loads a screen page in memory (screen)3 -19
ssave 8,"name"[ ,n][ ,n2ton3]	- saves a screen page on disk with the zones3-18
tcol ln,l,c	- draws a line of length ln (origin l,c)3
-4 tline ln,l,c	- draws a col. of length ln (origin l,c)3
-3	

### Reserved Variables

zo	- ASCII code for the return key in
request	mode (decz)
ok	- pointer in string.

### Not-Reserved Variables (Programmer's Choice)

a\$	- string - transfer
c	- column number (x-coordinates)
ca	- ASCII value of a character
co	- colour (for the rev instruction)
f\$	- print-using format
l	- line number (y-coordinates)
lg	- width(or height) of a window(zone)-(
number	of lines).
ln	- length of a window (zone) - number of columns
	- length of a string (creatst)
n1,n2,n3,n	- zone identification number
ty	- formatting type (decz)
	- scrolling type (scroll)-(u,d,l,r)

auto n	- numbers the lines automatically	5-2
creatst a\$,ln[,ca\$]	- creates any string of characters.	6-3
color [-]l,c,lg,ln,co	- color like rev but in graphic mode	4-8
delete l1-l2	- deletes program lines (like list)	5-2
display a\$, l,c	- displays a screen on the graphic screen	4-6
draw x,y,ty	- draws a vector on the gaphic screen	4-3
dump	- gives the list of the variables used by a program and their values	5-4
error	- locates an error within a BASIC line	5-5
find (deli)(string) (deli)	- finds string of characters within a program	5-6
graphic	- swtchs the screen in GRAPHIC mode	4-1
hcopy	- copies a screen on a printer automati- cally	6-5
hunt[-],ca\$,a\$,pn	- searches a character within a string	6-2
if (c) then (i1) : else (i2)	- structured programming instruction	6-4
move x,y	- moves the graphic cursor	4-2
off	- end of step-by-step execution mode	5-7
plot x,y,ty	- draws (or erases) a point on the graphic screen.	4-4
point x,y,ty	- tests if a point is on the graph.	screen 4-5
renu l1,l2,l3	- rennumbers a program automatically	5-3
sound a1,a2,a3,a4,v	- creates a sound (not implemented yet)	6-6
text	- switches the screen in TEXT mode	4-7
trace	- step-by-step execution mode	5-7

### Reserved functions

JOY(1)	- value given by the joystick	6-7
JOY(2)	- value given by a second joystick	6-7

### Not Reserved Variables [Programmer's Choice]

a\$	- any string of characters
br	- border color
c	- origin point (column number)
ca	- any ASCII value
ca\$	- character to be searched
co	- color option
cr	- character color
(c)	- condition (ex: A = B)
(deli)	- delimiter.(any character but")
(i1) (i2)	- instructions
l	- origin point (line number)
lg	- height / (number of lines)
ln	- length / (number of columns)
l1,l2,l3	- program lines numbers
n	- increment for the auto numbering
pn	- position pointer for a character
research	
(string)	- characters to be searched in a program
sc	- screen color
ty	- any numeric variable/1 - draws, 2 - erases
x,y	- coordonates of a point to draw

List of the error messages generated by THE TOOL with an explanation of the problem

BAD\_FORMAT

decz using the p-type :  
the print-using defined does not correspond to  
the  
variables used - length problems (see 3-10)

NO\_COLOR

A color parameter is missing or the value of a  
parameter is not allowed. (see instruction 3-21)

NO\_ZONE

A z-type instruction has been performed on an  
un-declared zone (see 3-9 to 3-15)

OUT\_OF\_PAGE

coordinates of the origin point or length or height  
are incorrect in the screen generator instructions.