

HI-RES for C64 BASIC Syntax

Following is the syntax for all commands accepted by [HI-RES](#) software extending Commodore BASIC:

HIRES 1 – activate primary hi-res screen

HIRES 1 CLR – activate primary hi-res screen and clear hi-res screen

HIRES 1 GET – copy/get primary hi-res screen from alternate hi-res screen

HIRES 1 PUT – copy/put primary hi-res screen to alternate hi-res screen

HIRES 1 SWAP – swap primary hi-res screen and alternate hi-res screen.

HIRES 0 – deactivate hi-res screen, switching back to text screen

HIRES 0 PLOT – copy text screen to hi-res screen

HIRES 1 G1 [,C1] [CLR] [PLOT] – activate hi-res screen at graphics address G1, with color table at address C1, optionally clear hi-res screen, and optionally just set as plot screen without activating/switching screens.

HIRES 1 G1[, C1] SWAP G2[, C2] – define primary and alternate hi-res screen addresses.

Optionally include definition of color table address(es) for primary and/or alternate hi-res screens. No screen is activated with this syntax.

COLOR [0|1], [FG][, BG], BD – set text (0) or hi-res (1) colors for foreground, background, and/or border. Colors are numbers 0 to 15. All numbers may be omitted, defaults to text screen if not specified. Except if a comma is specified, there must be a value 0-15 after the last comma.

COLOR [0|1], [FG][,BG] @ X1,Y1 [TO X2, Y2] – change foreground and/or background (hi-res (1) screen only, otherwise ignored for text (0) screen) for a single cell or range of cells. For the text (0) screen, these are row (0-24) and column (0-39) coordinates. For the hi-res (1) screen, these are pixel x (0-319) and y (0-199) coordinates, and affect an entire 8x8 pixel cell area.

PLOT 0|1, @ X1, Y1 [TO X2, Y2][...] – clear (0) or plot (1) a point or line or set of lines (multiple TO phrases allowed) on hi-res screen.

PLOT 0|1, "ABC", X, Y – clear (0) or plot (1) text on hi-res screen.

PLOT COLOR [FG][,BG] – set the foreground and/or background colors for the next hi-res operation. Allows use of all 16 colors including different backgrounds on hi-res screen (unlike standard text screen where background color is set for entire screen)

RECT 0|1 @ X1, Y1 TO X2, Y2 – clear (0) or plot (1) a rectangle on hi-res screen

PATTERN ADDR @ X1, Y1 TO X2, Y2 – fill rectangular area of hi-res screen from 8x8 pixel pattern defined at address.

SHAPE GET|PUT|OR|XOR|AND|NOT ADDR, X1,Y1 TO X2, Y2 – retrieve (GET) or place (others) a shape to/from address from/to hi-res screen. PUT stores the image to the hi-res screen. OR/XOR/AND/NOT performs a binary combination of shape and hi-res screen contents.

When alternate screen is logical (color table not a valid physical color table), then byte swapping is used to switch contents between screens, but when alternate screen is physical with a valid physical color table, then the switch is instantaneous and swaps the definition of primary and alternate screens.

Color table addresses only need to be defined once corresponding to the graphics address, thus are optional for future calls. Using the address syntax of switching or plotting to a hi-res screen is only necessary if changing color table, or changing to a different graphics address than that of the primary or alternate hi-res screen – otherwise simpler syntax (no address specified) is recommended when not necessary to specify the address – the exception is when it is ambiguous which graphics address is primary or alternate and you desire to activate a specific one. If the primary graphics address changes, then the former primary graphics/color addresses become the alternate addresses.

Graphics addresses must be multiple of 8192. Color addresses must be multiple of 1024. Primary color table must be in same 16K block as graphics address. Alternate hi-res color table address can satisfy requirements for a physical color table for fast screen switching, or can be elsewhere to be a logical alternate hi-res screen requiring byte swapping between physical and alternate screens (a bit noticeably slower, but has advantages for memory management). Once defined, color table addresses are remembered corresponding to that 8K graphics address.

Single Line Examples:

```
HIRES 1, 57344,56320 SWAP 40960,35840: REM define primary $e000/$dc00, alt $a000,$8c00
HIRES 1, 40960, 35840 CLR: REM activate hi-res primary screen $A000 (color $8C00) and clear
HIRES 1, 10*4096, 8*4096+12*256: REM $a000, $8c00
HIRES 1, 40960 : REM activate primary $a000
HIRES 1, 40960 CLR: REM activate and clear $a000
HIRES 1, 40960 CLR PLOT: REM clear graphics, but plot means define address(es) without activating
COLOR 0, 14, 6, 14 : REM text colors lt blue on blue with lt blue border
COLOR 1, 15, 0, 14 : REM graphics colors lt gray on black with lt blue border
PLOT COLOR 1, RND(1)*15+1 : REM randomly set next used graphic foreground color other than black
PLOT RND(1)*2 @ RND(1)*320, RND(1)*200 : REM plot or clear random point
PLOT 1 @0,0 TO 319,199 : REM diagonal across screen
PLOT 1 @100,100 TO 220,100 TO 160,20 TO 100,100 : REM plot triangle
RECT 1 @5,5 to 315,195 : REM large rectangle
FOR I=0 TO 7 STEP 2:POKE 700+I,85:POKE 701+I,170:NEXT:PATTERN 700 @ 0,0 TO 319,199: REM checkerboard
SHAPE GET 40960, 0,0 TO 159,99: REM save quarter of screen
SHAPE PUT 40960, 160,100 TO 319,199: REM restore quarter of screen
POKE 780,0:SYS 49152: REM reinit hi-res software, e.g. after warm reset (SYS 64738 or hw equivalent)
```

Colors:

| | | | |
|---------|----------|-----------|------------|
| 0 Black | 4 Purple | 8 Orange | 12 MdGray |
| 1 White | 5 Green | 9 Brown | 13 LtGreen |
| 2 Red | 6 Blue | 10 LtRed | 14 LtBlue |
| 3 Cyan | 7 Yellow | 11 DkGray | 15 LtGray |

C64 Memory Map with HI-RES

```
0000-03FF Zero Page and Low Memory
0400-07FF Text Screen Codes Memory (24 bytes extra)
0800-9FFF BASIC RAM
0800-0CFF (Optional) Scrolling BASIC Editor Software will limit BASIC RAM to 0D00-9FFF
```

A000-BFFF BASIC ROM
A000-BFFF HI-RES Alternate Graphics Screen
C000-CFFF HI-RES Machine Code (RAM)
D000-D423 HI-RES Machine Code (RAM under I/O Space)
D424-D7FF Available RAM Under I/O Space
DC00-DFFF HI-RES Color Table (RAM under I/O Space)
D000-D7FF I/O Space
D800-DBFF VIC-II Color RAM Nybbles (Text Color RAM is only 4-bit)
D800-DBFF HI-RES Alternate Virtual Screen Color Table (RAM under I/O Space - NOTE: 8-bit RAM)
DC00-DFFF HI-RES Primary Color Table (RAM under I/O Space - NOTE: 8-bit RAM)
E000-FFFF KERNAL ROM
E000-FFFF HI-RES Primary Graphics Screen (RAM under ROM)

Possible Graphics Screens

1st 16K RAM BANK

0000-1FFF Not Available due to various conflicts

2000-3FFF BASIC RAM reserved, possible Graphics Screen

2nd 16K RAM BANK

4000-5FFF BASIC RAM reserved, possible Graphics Screen

6000-7FFF BASIC RAM reserved, possible Graphics Screen

3rd 16K RAM BANK

8000-9FFF BASIC RAM reserved, possible for Virtual Graphics Screen or Color Table(s) only due to Character ROM conflict

A000-BFFF Graphics Screen available

4th 16K RAM BANK

C000-DFFF **Not Available due to HI-RES Software located here**

E000-FFFF Graphics Screen available

* To take RAM away from BASIC, typically you would reduce the RAM reserved for BASIC by changing memory location 56 (normally 160 for \$a000 limit), and then NEW to clear variables and reset BASIC pointers. Similarly memory location 44 reserves the lower limit of BASIC RAM.

* Note: physical graphics screens require color table to be in same 16K RAM bank, and at 1K offset.