A Color Screen For The HP-87 Emulator

Philippe Bergheaud

1. Introduction

The HP-87 color ROMs control a software color screen that runs in parallel with Everett Kaser's emulator. The color screen typically duplicates the CRT in GRAPH or GRAPHALL modes, with extra colors. The color screen may also operate as secondary keyboard, multiplexed with the emulator keyboard.

There are now two executable files: the emulator `hp85.exe` and the color screen `sdlgraph.exe`. The emulator and the color screen processes can be lauched or terminated asynchronously at any time. Each process runs in its own window. They run as coprocesses and communicate via two OS files used as pipes.

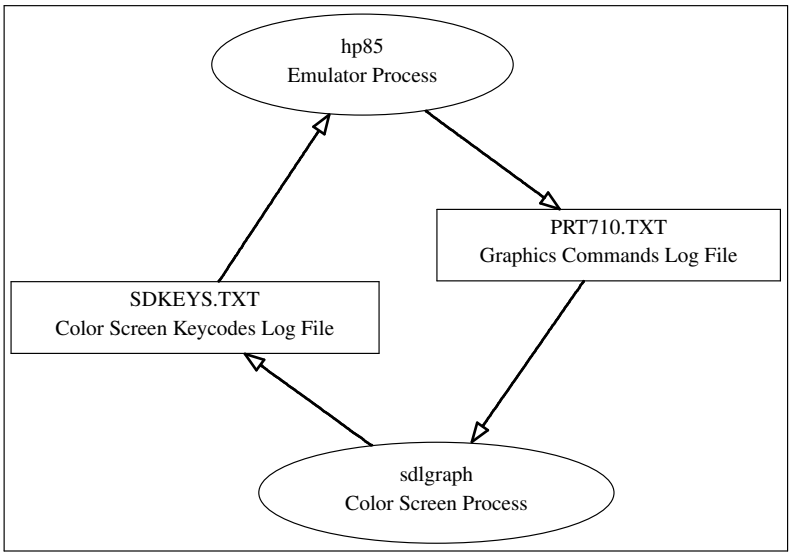

Figure 1. Coprocesses and OS Files

The emulator process `hp85` optionnaly logs CRT writes performed in GRAPH and GRAPHALL modes, or specific graphics commands, to the OS file `PRT710.TXT`. The color screen process `sdlgraph` reads `PRT710.TXT` and reproduces the graphics actions on the color screen.

The color screen process `sdlgraph` logs the keycodes entered in the color screen window to the OS file `SDKEYS.TXT`. The emulator process `hp85` optionnaly reads keycodes from `SDKEYS.TXT` and treats them as legitimate HP-87 emulated keyboard input.

2. Requirements

The HP-87 color extension runs on Everett Kaser's Series 80 Emulator 6.0.0, that provides a new HP-87 Color ROM (027), and four slightly modified system ROMs:

Filename	ROM Name	Status
`roms87\romsys1color`	System ROM 1	modified
`roms87\romsys2color`	System ROM 2	modified
`roms87\rom001color`	Graphics ROM (C)	modified
`roms87\rom027`	Color ROM	new
`roms87\rom320color`	Mass Storage	modified

Table 1.

The HP-87 color extension is not enabled by default. The color extension is controlled by the Color ROM (027) entry of the "Series 80 OPTIONS" menu. When the Color ROM is selected, the color versions of the system ROMs are automatically installed and selected, as well as the AUX 1 ROM (261).

When the Color ROM is deselected, the original versions of the system ROMs are restored.

3. Compatibilty

The differences introduced by the color version of the HP-87 system ROMs are the following:

1. The RESULT key duplicates the HOME CURSOR key. The result of the last CALC mode operation cannot be accessed any longer via the RESULT key.
2. The OS files `PRT710.TXT` and `SDKEYS.TXT` are truncated at POWER-ON. `SDKEYS.TXT` is left open with file handle `10`.
3. The Color ROM uses 93 bytes of RAM.

The emulator does not interact with the color screen by default. The Color ROM provides a set of new keywords to control the color screen and perform color screen operations.

4. The Color Screen Process -- Sdlgraph

The color screen process must be started separately from the emulator process. The color screen executable is available in two flavours: `sdlgraph.exe` and `sdlgraph_fr.exe`, respectively for US and French PC keyboards. They require the SDL library file `SDL2.dll`, which is also provided. The command line syntax is:

```
sdlgraph [-d delay] [filename]
```

- `delay` Delay (in milliseconds) between graphics commands (default: 0)
- `filename` Graphics commands input file (default: `PRT710.TXT`)

The color screen process `sdlgraph` opens an HP-87 GRAPH-sized window at startup, reads graphics commands from the input file, and reproduces them on the color screen. The input file may already contain graphics commands, and new commands may be appended to the input file at any time.

When the end of the input file is reached, `sdlgraph` monitors the input file length and waits for new graphics commands to arrive. If the input file is truncated by the emulator (this happens at POWER-ON), `sdlgraph` seeks to offset 0, and resumes reading commands from the beginning of the input file.

4.1 Color Screen Interactive Mode -- Sdlgraph

If `sdlgraph` is started without argument, then `sdlgraph` interprets the graphics commands in the file `PRT710.TXT`, and logs keystrokes entered in the color screen window to the file `SDKEYS.TXT`. This is the default mode, meant to interact with the emulator.

Double-clicking on `sdlgraph` enters the interactive mode.

4.2 Color Screen Replay Mode -- Sdlgraph Filename

If an input `filename` is specified, `sdlgraph` ignores keyboard input, and simply replays the graphics commands in the file. This is the non-interactive mode, intended to replay interesting graphics commands files previously recorded by the emulator, renamed, and saved.

Opening `filename` with `sdlgraph` enters the replay mode.

4.3 Color Screen Window Actions

The color screen window can be moved around, minimized, unminimized, or closed. It cannot be resized. Screen updates are suspended while the color screen window is minimized, and resumed when the color screen window is unminimized. Mouse interaction with the color screen window also updates the color screen in asynchronous mode (See `ASYNC` below).

Closing the color screen window terminates the color screen process.

5. Color ROM Keywords

5.1 COLOR

```
COLOR red,green,blue
COLOR CHR$(red)&CHR$(green)&CHR$(blue)
```

sets the current color for the color screen. The argument values (modulo 256) define the intensity of the primary colors.

Color is an attribute of the color screen process. Color is bright white (255,255,255) at POWER-ON.

5.2 COLOR$

```
COLOR$
```

returns the string `CHR$(red)&CHR$(green)&CHR$(blue)` defining the current color.

5.3 COLOR ON

```
COLOR ON
```

enters the emulator `COLOR ON` mode, in which all CRT graphics writes are duplicated on the color screen, in the current color.

`COLOR ON/OFF` is an attribute of the emulator process, set to `COLOR OFF` at POWER-ON.

5.4 COLOR OFF

```
COLOR OFF
```

disables the emulator `COLOR ON` mode. This is the default mode.

`COLOR ON/OFF` is an attribute of the emulator process, set to `COLOR OFF` at POWER-ON.

5.5 LOWRES

```
LOWRES
```

enters the color screen `LOWRES` (low resolution) mode, with one color per screen byte. This is the default mode.

`LOWRES/HIRES` is an attribute of the color screen process, set to `LOWRES` at startup.

WARNING: Writes aligned on a byte boundary always operate as expected, but misaligned writes may assign the current color to extra adjacent pixels by side effect.

5.6 HIRES

```
HIRES
```

enters the color screen `HIRES` (high resolution) mode, with one color per pixel. In `HIRES` mode, the color screen obeys the following rule: ***"While a pixel is present on the color screen, its color never changes."***

With this convention, it becomes possible to adapt the byte-oriented I/O architecture of the HP-87 to the pixel-oriented color screen, at the cost of repeated byte writes (at the same address).

1. Changing the color of on-screen pixels requires two writes. The selected pixels must first be erased (writing the byte with the selected pixel bits masked) and then plotted again in the new color (writing the byte again, this time with the selected pixel bits set).
2. Assigning different colors to different pixels of the same screen byte requires one write per color. For each color, the plotted byte must hold the union of the pixels already on screen (which color will not change) and the new pixels (that will be plotted in the current color).

`LOWRES/HIRES` is an attribute of the color screen process, set to `LOWRES` at startup.

5.7 REDRAW

```
REDRAW
```

enters the emulator `REDRAW` mode, in which CRT bytes are always replicated on the color screen in `COLOR ON` mode, regardless of their on-CRT status. This is the default mode.

`REDRAW/NOREDRAW` is an attribute of the emulator process, set to `REDRAW` at POWER-ON.

5.8 NOREDRAW

```
NOREDRAW
```

enters the emulator `NOREDRAW` mode, a submode of `COLOR ON + LOWRES`. In `NOREDRAW` mode, CRT bytes are not plotted twice at the same address with the same value. This reduces the amount of information exchanged between the emulator process and the color screen process, when the same CRT contents is plotted repeatedly.

In `COLOR ON + LOWRES + NOREDRAW` mode, the emulator enforces the following rule: ***"While a byte is present on the CRT, its color does not change."*** Therefore, changing the color of a on-CRT byte without changing its value requires two CRT writes. The byte must be first blanked (zeroed), and then plotted again with the new color.

`REDRAW/NOREDRAW` is an attribute of the emulator process, set to `REDRAW` at POWER-ON.

5.9 SYNC

```
SYNC
```

enters the color screen `SYNC` (synchronous) mode, in which the color screen is automatically updated after each write. This is the default mode.

`SYNC/ASYNC` is an attribute of the color screen process, set to `SYNC` at startup.

WARNING: Color screen updates are relatively slow. Updating the screen after each write may noticeably slow down the color screen process.

5.10 ASYNC

```
ASYNC
```

selects the color screen `ASYNC` (asynchronous) mode, in which the color screen is NOT automatically updated. In `ASYNC` mode, color screen updates must be explicitely requested with the keyword `PRESENT`. Mouse interaction with the color screen window also updates the color screen.

`SYNC/ASYNC` is an attribute of the color screen process, set to `SYNC` at startup.

5.11 PRESENT

```
PRESENT
```

updates the color screen (once) in `ASYNC` mode. All color screen writes accumulated since the last update are plotted simultaneously. Crossing the color screen window boundaries with the mouse pointer also updates the color screen in `ASYNC` mode.

5.12 DELAY

```
DELAY seconds
```

suspends color screen updates for 0 to 255 seconds. The countdown is performed by the color screen process. Minimizing the color screen window stops the countdown until the window is made visible again.

5.13 CREV

```
CREV
```

returns the revision number of the Color ROM.

5.14 CLEAR

```
CCLEAR [y1,[y2]]
```

clears a GRAPH-sized color screen, from line y1 to line y2. If PEN >= 0, then the lines are blanked. If PEN < 0, then the lines are filled with current color. The color screen process switches to a GRAPH-sized window if necessary. This keyword ignores COLOR OFF, HIRES and NOREDRAW.

The screen size is an attribute of the color screen process. The color screen opens a GRAPH-sized screen at startup.

- y1 = 0–239 Line number, top to bottom, default 0
- y2 = y1–239 Line number, top to bottom, default 239

5.15 CCLEARALL

```
CCLEARALL [y1,[y2]]
```

clears a GRAPHALL-sized color screen, from line y1 to line y2. Honors PEN value like CCLEAR. Switches to a GRAPHALL-sized window if necessary. This keyword ignores COLOR OFF, HIRES and NOREDRAW.

- y1 = 0–239 Line number, top to bottom, default 0
- y2 = y1–239 Line number, top to bottom, default 239

5.16 CLABEL

```
CLABEL x,y,text$[,target]
```

plots a byte-aligned text string on the emulator CRT and/or the color screen. This keyword ignores COLOR OFF, HIRES, NOREDRAW, and PEN value.

- x = 0–49 Byte offset, left to right, in GRAPH mode
- x = 0–68 Byte offset, left to right, in GRAPHALL mode
- y = 0–239 Line number, top to bottom
- text$ Text string
- target = 0 Plot to emulator CRT and color screen (default)
- target = 1 Plot to emulator CRT
- target = 2 Plot to color screen

WARNING: The color screen characters in the range 0-31 belong to a special (ad hoc) graphics character set, different from the same range in the HP-87 character set.

5.17 PIXEL

```
PIXEL x,y
```

plots a single pixel on the color screen. If PEN >= 0, then the pixel is plotted in current color. If PEN < 0, then the pixel is erased. This keyword ignores COLOR OFF, HIRES and NOREDRAW.

- x = 0–399 Pixel offset, left to right, in GRAPH mode
- x = 0–534 Pixel offset, left to right, in GRAPHALL mode
- y = 0–239 Pixel offset, top to bottom

5.18 CLOSE

```
CLOSE
```

closes the color screen window and terminates the color screen process. Closing the the color screen window with a mouse action also terminates the color screen process.

5.19 SDISOPEN

```
SDISOPEN file#
```

returns 1 if file handle `file#` is associated to an open file, else returns 0.

- `file# = 1-10` File handle

5.20 SDKBD

```
SDKBD file#
```

enables or disables keyboard input from an open OS file, with file handle `file#`. When `SDKBD` is enabled and the end of the file is reached, the emulator monitors the file length and waits for new keycodes to arrive.

— In HP-87 IDLE mode: the emulator reads and processes input typed into the emulator window, as well as input from the `SDKBD` file.

— In HP-87 RUN mode: keycodes must be processed by the running program, with the keyword `SDKEY` (See below).

- `file# = 1-10` Enables keyboard input from the file open with file handle `file#`
- `file# = 0` Disables keyboard input from the current `SDKBD` file

`SDKBD` is an attribute of the emulator process. The `SDKBD` keyboard is inactive at POWER-ON.

5.21 SDKEY

```
SDKEY
```

reads a keycode from the KBD buffer (if `KBDBUFFER` is on, see below), or from the `SDKBD` file (if `SDKBD` is enabled), and returns its value in the range 0-255.

`SDKEY` returns -1 if no keycode could be read.

5.22 KBDBUFFER

```
KBDBUFFER boolean
```

takes over keyboard and buffers up to 16 keys at a time if arg==non-zero, else turns OFF keyboard buffering and releases the keyboard. The runtime statement

```
SDKBD file# @ KBDBUFFER 1
```

multiplexes keyboard input from the emulator window and keyboard input coming from the OS file with file handle `file#`.

`KBDBUFFER` is an attribute of the emulator process. `KBDBUFFER` is turned off at POWER-ON.

6. Color Screen Keyboard -- SDKBD 10

The emulator process `hp85` opens and truncates the OS file `SDKEYS.TXT` at POWER-ON, and leaves it open with file handle 10. The color screen process `sdlgraph` logs the keycodes typed in the color screen window to this file. Therefore, the statement

```
SDKBD 10 [@ KBDBUFFER 1]
```

activates keyboard input from the color screen window.

— In HP-87 IDLE mode: the emulator accepts keyboard input from both emulator window and the color screen window.

— In HP-87 RUN mode: the keycodes typed in the emulator window are returned by the function `SDKEY`. If `KBDBUFFER` is on, then `SDKEY` also returns the keycodes typed in the emulator window.

6.1 PC Keys To HP-87 Keys Mapping

The color screen process `sdlgraph` maps the PC keys into HP-87 keys, exactly like the emulator does, with one addition: The End key duplicates Break/Pause (for PC keyboards without Break/Pause key).

HP-87 keycodes returned by `SDKEY` are given in decimal in the table below.

PC Key	**HP87 Keycode Key**	(with PC Key Shifted)
Backspace	153 BACKSPACE	155 FAST BACKSPACE
Break/Pause	142 PAUSE	144 STEP
Delete	136 -CHAR	157 -LINE
Down Arrow	164 DOWN CURSOR	164 DOWN CURSOR
End	142 PAUSE	144 STEP
Escape	150 KEY LABEL	150 KEY LABEL
Return	154 ENDLINE	154 ENDLINE
F1	128 K1	132 K8
F2	129 K2	133 K9
F3	130 K3	134 K10
F4	131 K4	135 K11
F5	161 K5	165 K12
F6	162 K6	172 K13
F7	156 K7	147 K14
F8	141 RUN	143 CONT
F9	148 LIST	149 PLIST
F10	168 A/G	168 A/G
Home	166 HOME CURSOR	137 CLEAR
Insert	158 I/R	158 I/R
Left Arrow	159 LEFT CURSOR	159 LEFT CURSOR
PageDn	169 ROLL v	169 ROLL v
PageUp	145 ROLL ^	145 ROLL ^
Right Arrow	170 RIGHT CURSOR	170 RIGHT CURSOR
Up Arrow	163 UP CURSOR	166 HOME CURSOR

Table 2.

7. Demo Disk

The disk `disks0\ColorROM87` contains various games and other demonstration programs for the HP-87 emulator with Color ROM, Advanced Programming ROMs, and 1024 Kbytes of extended RAM.

7.1 Games

`Asteroids`	Everett Kaser's ASTER game
`Bats + bats`	Everett Kaser's BATS game
`Bulls + bulls`	Everett Kaser's BULPEN game
`Fatcat + fatcat`	Randy Salo's FATCAT game
`Pacman + pacman`	John Maushammer's PACMAN game
`Dedor + dedor`	A strategy game of mine

All games take control of the keyboard and can be paused (or stopped) with the PC key Break/Pause, or the PAUSE key of the emulator screen. All games accept input from the emulator or color screen windows.

7.2 L-System Demos

`Dragon`	Dragon curve
`Kite+Dart`	Penrose's kite + dart tiling
`Penrose3`	Penrose's type 3 tiling
`Plant`	Fractal plant

7.3 Game Of Life Demos

`Atoll`	Random atolls (Walled Cities Life)
`Blob`	Random blob (Maze Life)
`Circles`	Random circles (Diamoeba Life)
`Glider`	Conway's life glider

8. Color ROM Rev 1 Public Constants, Variables and Entry Points

C.ROM#	EQU 000027	Color ROM number
C.REV#	EQU 000001	Color ROM revision number
CLEARHK	DAD 100070	CRT Clear hook (readonly data, 4 bytes)
WRITEHK	DAD 100074	CRT Write hook (readonly data, 4 bytes)
Plot	DAD 060645	Plot byte R32 to color screen at address R34-35
PlotBAD	DAD 060664	Plot byte R2 to color screen at address CRTBAD
Coloron	DAD 061662	Set `COLOR ON` mode (plot to CRT and color screen)
Colorof	DAD 061700	Set `COLOR OFF` mode (plot to CRT only)
Lowres	DAD 061725	Set `LOWRES` mode (one color per byte)
Hires	DAD 061743	Set `HIRES` mode (one color per pixel)
Present	DAD 061753	Update color screen once (in `ASYNC` mode)
Close	DAD 061763	Close color screen window
Sync	DAD 061772	Set `SYNC` mode (update color screen after write)
Async	DAD 062002	Set `ASYNC` mode (do not update color screen)
Redraw	DAD 062012	Set `REDRAW` or `NOREDRAW` mode with R2
Color?	DAD 062063	Return current color (R55,R56,R57)=(blue,green,red)
Color	DAD 062147	Set current color (blue,green,red)=(R55,R56,R57)
Pixel	DAD 062305	Plot pixel x=R56-57, y=R55 to color screen
Delay	DAD 062344	Delay color screen update for R55 seconds
Clear	DAD 062464	Clear color screen from y1=R32 to y2=R33
Sdkey	DAD 063337	Copy one byte from `SDKBD` file into KEYHIT
Sdeoj2	DAD 063471	Seek to end of `SDKBD` file

Table 3.

The byte at address WRITEHK contains `262` (octal) in `COLOR OFF` mode, `316` in `COLOR ON` mode.

CONTENTS

LIST OF FIGURES

LIST OF TABLES