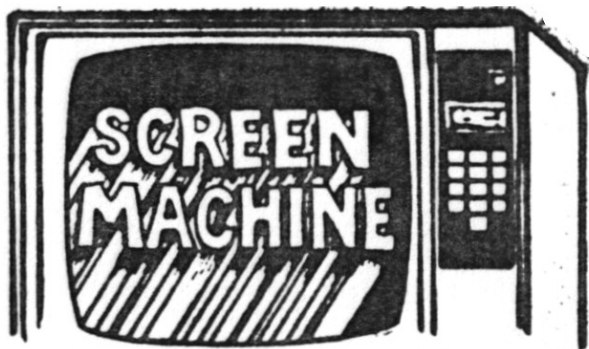*RAINBOW CONNECTION SOFTWARE*
presents . . .

SUPER
SCREEN MACHINE

Revolutionary — heralded as the most useful, powerful and versatile state-of-the-art utility, ever developed for the Color Computer!

RAINBOW CONNECTION SOFTWARE

3514 6th Place NW
Rochester, MN 55901

32K recommended
16K Extended Basic Require
$44.95 Cass - $47.95 Disk.

SCREEN MACHINE
by
Dan Larson and Paul Penrose
(c) 1983 Rainbow Connection Software
SPECIFICATIONS

PRELOADER

Language: ML.
Memory Consumption: 0 bytes net (memory reverts back to BASIC or ML
  program RAM).
Purpose: Loads Screen Machine in upper memory of 16K, 32K, or 64K for
  optimum memory efficiency. The 64K option also transfers all ROM to
  RAM, including cartridge, and stays configured even after reset.

SCREEN MACHINE

Language: ML (relocatable).
Memory Consumption: Approximately 4.9K, 5.3K for Super Screen Machine,
  plus up to 6K of graphics pages (PCLEAR 4).
Features:
-- Four standard and four double-width character densities in PMODE 4.
-- Twelve character densities available from 64x24 to 16x8.
-- Two distinct character sets automatically switch for sharpest
   lettering in all sizes.
-- Functions in all PMODEs (0-4) and pages (1-8).
-- Colored characters available in PMODEs 1 and 3, plus artifacted
   colors in PMODE 4.
-- Easy 2-letter commands implemented by simple PRINT CHR$(27)"__."
   commands, or via a shift/clear control key for easy keyboard entry.
-- Help screen can be accessed to display all Screen Machine command
   syntax and current status.
-- Print speed approximately 200 characters/second at 64 cpl.
-- Scroll speed from 1.5 lines/second (smooth scroll) to 12 lines/second.
-- Added commands include underline, subscript, superscript, top and
   bottom scroll protect, artifacts, and double width.
-- Super Screen Machine adds these commands: variable smooth scroll,
   Superpatch disk EDTASM+ command, dynamic screen dump interface command,
   variable key click, break disable, 10 user commands, and EDTASM+ command.
-- BASIC interface of all commands, including PRINT @, TAB, and comma
   fields. Keyboard interface of all keys.
-- Error checking of all SM commands and appropriate error messages.
-- Full ASCII 224-character set, including true lowercase with
   descenders, improved cursor, slashed zero, Greek math symbols, lunar
   landers, stick figures, tanks, cars, planes, card suits, electronic
   symbols, etc.

CHARACTER GENERATOR (CHARGEN)

Language: BASIC.
Memory Requirement: 32K.
Purpose: Create custom character sets for use with Screen Machine.
Operation: Uses arrow keys and 1-character commands to change, save, and
load custom character sets.

DEMO (DEMO)

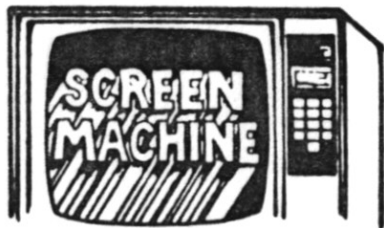Language: BASIC.
Memory Requirement: 32K.
Purpose: Demonstrate the many advanced features of Screen Machine and
  potential of character-graphics programming techniques.
Operation: Self-executing, menu-driven.

SCREEN MACHINE

Table of Contents

SCREEN MACHINE
by
Dan Larson and Paul Penrose
(c) 1983 Rainbow Connection Software

INTRODUCTION

Congratulations! You have just purchased the ultimate in hi-res graphics
text display utilities. Screen Machine overcomes the greatest shortcoming
of the Color Computer--namely, the plain black on green, low-density (32x16)
text display that can't even be combined with hi-res graphics! In addition,
many truly useful screen text commands have been added along with a set of
128 useful character-graphics figures. Screen Machine is relocatable and
resides in upper memory, providing high-density screen displays for a variety
of uses or to incorporate into your own programs for an added dimension in
professional screen displays. Screen Machine is so versatile and yet so
advanced that its applications are limitless. But perhaps the most important
feature of all is its ease of use.

If you can't wait to start using Screen Machine's advanced features, just
proceed to the Demo Program section and enjoy the show. Then when you're
ready, continue reading the detailed instructions and view the listing of
the Demo Program. Combined with the aid of the help screen, you'll be an
expert in no time at all creating complex screen displays never before
possible.

LOADING THE PROGRAM

Diskette

Insert the diskette and type: Run "SM".

Cassette Tape

Insert the cassette into your recorder and type: CLOADM. If the first copy
fails to load, try the second later on the tape.

Disclaimer

Reproduction of this program or documentation in any form other than owner's
backup copy is strictly forbidden. Rainbow Connection Software is not
responsible for any loss or damage incurred through the use of its products
and offers no warranty, either expressed or implied, beyond immediate
replacement of defective materials.

Replacement and Upgrade

You may replace your manual and program or upgrade to Super Screen Machine
at any time. See our ads in leading Color Computer magazines for informatic

PRELOADER

After Screen Machine has loaded, the Preloader selection will be displayed.
Normally, you will want to select the choice corresponding to your machine'
memory so that Screen Machine will reside out of the way at the top of
memory. There are times, however, when manually selecting another
configuration is advantageous.

The 64K selection will also load all ROM (including cartridge and Basic)
into RAM memory to allow full use of the lower 32K of memory. Pressing
reset will not "crash" Screen Machine and will retain the 64K configuration
by using the last 8 bytes of 32K. Although Screen Machine resides at the
top of 64K, you will still have only the lower 32K for your Basic program
use as before, which can be verified with the PRINT MEM command.

SCREEN MACHINE OPERATION

Upon exiting the Preloader, the Screen Machine title screen will be displayed. All the marvelous features of Screen Machine are now at your disposal. It is important at this point that you notice that you are now back in Basic's immediate command mode and you can type in any Basic command or type in a program.

All of the added Screen Machine commands are invoked by one of two methods. The first method is by pressing the two keys "SHIFT" and "CLEAR" at the same time. Try it. The text screen will be displayed and you will see a "?" in the upper left-hand corner along with a flashing cursor. Now just type in your Screen Machine commands, separated by either commas or spaces, up to 31 characters total. To execute these commands, press either the "." or "ENTER" key. Now your Screen Machine commands will be executed and you will return to what you were doing before you pressed "SHIFT"/"CLEAR." In fact, you can execute Screen Machine commands this way just about anytime, even when a Basic program is running!

The second method for executing Screen Machine commands is via a simple PRINT statement. The format is: PRINT CHR$(27);"sm commands." The period is optional only if a carriage return comes right after the command string. An example is:

     PRINT "THIS IS ";CHR$(27);"ULN.UNDERLINED";CHR$(27);"ULF"

The first Screen Machine command is "ULN" which turns the underlining on. Notice that the word "UNDERLINED" is put inside the same string as "ULN," but since the period is between them, there is no confusion to Screen Machine. The second Screen Machine command does not need a period at the end of it because when Basic hits the last quotation mark, it will generate a carriage return since no semicolon or comma follows.

You may wish to replace CHR$(27) with a predefined string variable (eg, A$=CHR$(27)) for ease of use, or you may wish to replace entire Screen Machine command statements with predefined string variables (eg, B$=CHR$(27)+"DWN,ACN,ULF,DN50,BS8."). Such predefinition not only simplifies later program statements, but can save memory as well.

SCREEN MACHINE ERROR DETECTION

Screen Machine contains its own built-in error detection and messages.

SM SYNTAX ERROR indicates an improperly phrased Screen Machine command.

SM LENGTH ERROR occurs when more than 31 characters are contained inside a stacked Screen Machine command.

SM RANGE ERROR indicates an illegal value in a Screen Machine command. Overscroll protection will cause this error to occur.

SM MISSING ERROR occurs when the parameter at the end of a Screen Machine command is missing such as N (on) or F (off).

SM SWITCH ERROR results from using an illegal switch in a Screen Machine command. Legal switches are N, F, N, R (N and R are legal only for the SD--screen dump--command).

The following is a list of Screen Machine added commands. Replace $ with N for on, or F for off. Replace # with a legal number.

HP   Prints a Help screen showing the present status of all SM commands.

DN## Determines density--32, 42, 50 (power-up), or 64 characters per line. This command will also move the cursor back to the beginning of the first line ("Homing" the cursor), and will turn off the top and bottom scroll protects.

TS# Top scroll protect. Defines the number of lines (0-14 or 22) you desire to be protected from scrolling at the top of the screen.

BS# Bottom scroll protect. Defines the number of lines (0-14 or 22) you desire to be protected from scrolling at the bottom of the screen, providing you are not printing there already. At least 2 lines of unprotected screen must exist between the top and bottom settings. If any errors are produced while setting either the top or bottom scroll protect parameters, both of them will be turned off.

DW$ Double width printing. This command works only in PMODE 4.

UL$ Automatic underlining until turned off.

SB$ Automatic subscript. Subscripts on the bottom line of the display will be ignored. Turning subscript on will automatically turn off superscript. If a carriage return is printed while subscript is on, it will automatically be turned off.

SP$ Automatic superscript. Superscripts on the top line of the display will be ignored. Turning superscript on will automatically turn off subscript. If a carriage return is printed while superscript is on, it will automatically be turned off.

AC$ Alternate color set. When this feature is turned on, the color of the characters is no longer controlled by Basic's COLOR command, but by a SM command. This allows the programmer to use Basic's hi-res commands with on color pair for graphics and have SM's characters displayed in another color very easily. This also allows you to assign colors to SM that Basic's COLOR command will not allow in certain PMODEs (such as PMODE 4). See the next command for more information on this.

CL#,# Once turned on with the AC command, the alternate color set colors are determined by the RW command "C#,#" where the #s are replaced with foreground and background color codes (0, 1, 2, or 3) similar to Basic's COLOR command. This command will allow you to enter any color code for character colors, even double width in highest resolution PMODE 4! For instance, CL2,3 will give you either brilliant blue or orange (unpredictabl upon power-up or reset) on buff (eg, PMODE 4:CLS 3:SCREEN 1,1:PRINT CHR$(27);"DWN,ACN,CL2,3."). Basic's CLS command followed by a 1 or 2 will also be enhanced to provide brilliantly colored backgrounds in PMODE 4.

These artifacted or pseudocolors can be created on most color TVs by plotting every other pixel or column when drawing a line or filling in an area in PMODE 4. You may also PAINT in PMODE 4 with the Basic command POKE 178,2 or POKE 178,1 and omit the foreground color (but not the comma) in your PAINT or LINE command.

SUPER SCREEN MACHINE COMMANDS

NOTE This section is only for those people who own SUPER Screen Machine.

As well as the preceding Screen Machine commands, Super Screen Machine has the following added commands. All commands are recognized (though not functional) and standardized on all Screen Machines, enabling software authors to create Super Screen Machine software that is totally compatible (will not crash) on all present and future Screen Machines.

SR# Scroll Rate. Defines rate of scrolling (1-4). A value of 1 will produce the "smoothest" and slowest scroll, while a value of 4 will give the "jerkiest" and fastest scroll. The program powers up with a rate of 2, which is a good, all-around scroll rate. Rates 3 and 4 are best for listing programs, where 1 and 2 can really enhance any video recorder title screens and other visually oriented programs.

KC# Key click volume control. Defines relative volume (0-4). The value you give it is how many quarters volume the keys will click when you press them. **NOTE** The click will come from your TV, so adjusting the volume control on it can control the overall volume of the clicks. Use a value of 0 to turn this function off.

BK$ Break key enable/disable. Turning this function off will result in the break key NOT being recognized by Basic. (It will also cause the shift/clear to not be recognized.) When the break key is turned off, the INKEY$ function in Basic will treat the break and shift/clear keys just like any other key on the keyboard (a character code of 3 will be returned if the break key is pressed). Turn the break key back on using the command PRINT CHR$(27);"BKN.". The Basic command POKE 113,0 will disable the reset button by "crashing" the entire system when pressed.

SD# Screen dump command. Replace # with either "N" for Normal print or "R" for Reverse print. Just load in any one of Custom Software Engineering's Graphic Screen Print programs (except Version 1.0s) into memory after Screen Machine has been loaded and executed. The screen dump program MUST have been the LAST machine language program loaded, must not occupy the same portion of memory as Screen Machine or any other program, and it must be the correct version designed for your printer in order for this command to work. You will find that reverse print in PMODE 4 will produce black characters on white as you probably desire.

We recommend you use the following decimal values when loading, according to your present Screen Machine configuration:

```
--   16K:  CLEAR200,10336:CLOADM"",8800
--   32K:  CLEAR200,26720:CLOADM"",25184
--   64K:  -------------- CLOADM"",55808
```

This command is great for getting "snapshots" of the screen even while Basic programs are running. Just press shift/clear and enter the screen dump command! Wow, dynamic screen dumps!

US# User command. This command will activate any one of 10 (0 through 9) special machine language drivers. These drivers may be created by the user or supplied by RCS or other publishers. Technical information on how to "hook" your own drivers into Screen Machine is provided in the "Advanced Topics" section later in this manual.

E+ This will properly execute your EDTASM+* cartridge, assuming you have one plugged in. The procedure is as follows. Power up into EDTASM+* then enter the "Q" command to go to Basic. In Basic, load in and execute Screen Machine. Now set up the density, PMODE, and colors you wish before entering the "E+" command and EDTASM+* will come back up using Screen Machine. If you have a 64K machine and you set up Screen Machine for 64K, doing an "E+" command will also patch EDTASM+* to use Screen Machine's keyboard routines, shift/clear, flashing cursor, and all!

S+ This will properly patch and execute Super Patched EDTASM+* disk (as detailed in September 1983 Rainbow magaine). The procedure is as follows. Load in and execute Screen Machine, setting it up for 16K by typing "1" when the loader menu comes up. Then load in your Super Patched EDTASM+* file using an OFFSET of 3200 hex (&H3200). Change the PMODE, COLOR, or SCREEN at this point before entering the Screen Machine command "S+" to enter EDTASM+*. You will have key click, shift/clear, and all of Screen Machine's power and commands at your control.

WARNING: Avoid typing Screen Machine commands after executing the S+ or E+ (64K) commands, as any Screen Machine errors will cause unpredictable and unreliable operation.

---

*EDTASM+ is a registered trademark of Tandy Corporation.

INTERFACED BASIC COMMANDS

CLS#

By experimenting with Basic's CLS# command and Screen Machine's alternate color set command (CL#,#), you can create many different color combinations in the normally drab PMODE 4 screen. CLS without a parameter will clear the screen to the current background color.

PMODE

Screen Machine functions perfectly in all PMODEs (resolutions). PMODEs 0,2 and standard PMODE 4 (power-up) offer only two colors at a time, while PMODEs 1 and 3 offer comparable resolution with four colors simultaneously. Lower PMODE numbers offer larger size characters, while higher numbered PMODEs provide more characters on the screen. All print locations are automatically adjusted. Always specify the PMODE to eliminate any source of confusion between Basic and Screen Machine. The current PMODE is displayed on the Help screen. When you change into different PMODEs, anything left on the screen will turn into meaningless garbage, so you will want to do a CLS immediately afterward. By specifying different page numbers with your PMODE commands, you can do "page flipping." For example, you might want to PCLEAR eight pages instead of just four, then print a menu on Pages 1 through 4. After the user chooses an option, you would do a PMODE 4,5 to flip to Pages 5 through 8 and not mess up your menu. When you want to go back to the menu, all you have to do is PMODE 4,1. A PMODE command always homes the cursor and turns off both scroll protects.

SCREEN

Screen Machine will write to any graphics page in either color set. Normally, SCREEN 1,0 (power-up) will yield a green background ideal for high-density text on a color TV, and SCREEN 1,1 will produce a white (buff) background with purer colors. SCREEN 1,1 works best for everything on a black and white TV. The SCREEN command should be your last Basic graphic command before printing or drawing.

COLOR

Basic's COLOR command (power-up COLOR 0,1) will set the foreground and background colors for Screen Machine as well. The current foreground and background colors are displayed on the Help screen. Also, in parentheses are the foreground and background colors of the alternate color set. Transposing foreground and background colors in the COLOR command will result in INVERSE characters (reverse video) being printed.

PRINT TAB ( )

Screen Machine automatically matches tabbing to the current character density.

PRINT @

True compatibility is maintained with all densities fully supported.
See table on back cover.

COMMA FIELDS

Standard 16-character spacing is maintained in 32-CPL and 64-CPL
densities, while 12-character spacing is employed in the 42-CPL density
and 25-character spacing in the 50-CPL density.

LOWERCASE

Lowercase letters are invoked from Basic by pressing SHIFT and Ø
at the same time. Screen Machine's lowercase with descenders add
legibility especially in higher densities. All Basic commands (even
editing) must be in uppercase.

CHARACTER GRAPHICS

The Screen Machine graphics characters (128 to 255) can be printed on
the screen by using Basic's command PRINT CHR$( ) as detailed in your
computer manual.

These characters, however, are far more useful and defined and can
even be used to easily create animation. Load and run the accompanying
Chargen program to view and modify these characters.

Basic's remaining hi-res graphics commands are fully interfaced and
may warrant your review. They are CIRCLE, DRAW, LINE, PAINT, PCLEAR,
PCLS, PPOINT, PSET, and PRESET.

CHARACTER GENERATOR (CHARGEN)

The character generator program is provided to allow you to make and save customized character sets. This complex program is written in Basic and machine language and requires 32K. DO NOT BREAK and rerun the program. On startup, the entire 224-character set will be displayed in 32 and 42 spaces (characters per line) size. (More on this later.) A list of all 1-key commands will be displayed in the upper right-hand corner of the screen. The commands are as follows:

Arrow Keys (not in command list) will move the flashing cursor over the selected character in the display area. The CHR$ code is also displayed for your convenience. Full wraparound of the cursor is provided for faster selection of the character.

A - Alter character. Enters the alter character mode, which is covered in another section.

S - Save character set. Will save the entire character set to a disk or cassette file. The file that is saved may be loaded back only with Screen Machine in the same memory configuration that it was saved under, or unpredictable results will be obtained.

L - Load character set. Will load the entire character set from a disk or cassette file. The file loaded must be loaded back with Screen Machine in the same memory configuration that it was saved under.

3 - Set display area to 32/42 size. The characters in the display area are in the 32-space and 42-space character size style.

6 - Set display area to 50/64 size. The characters in the display area are printed in the 50/64 size style.

The program may be aborted by pressing the break key.

Alter Mode

The alter mode has an entirely different screen display. A list of all 1-key commands for the alter mode are displayed in the upper left-hand corner of the screen. On the right side of the display are two large block character representations of the 50/64 and 32/42 styles. Two separate styles are needed to ensure legibility in the higher two densities and "looks" in the lower two densities. The higher density style is obtained by removing two columns from the 32/42 size data. The sixth column (usually blank) is removed automatically, and the other column to be removed is defined by the user. This column is called the DELETE COLUMN. The delete column can be changed by the user while in the alter mode. The column which is deleted is displayed under the correct position in the 32/42-block letter display.

Commands

A = Abort changes. Will abort any changes made to the character and return to the main screen.

S = Save changes. Will save the changes made to the character and return from the alter mode.

P = Set the current pixel.

@ = Reset the current pixel.

E = Erase entire character.

Z = Move the DELETE COLUMN left.

X = Move the DELETE COLUMN right.

C = Redraw the 50/64 size block character.

Arrow Keys

The arrow keys move the pixel cursor over the selected pixels with wraparound.

Notes

The 50/64 size style is a 4x8 matrix formed by removing two columns from the 32/42 size style which is a 6x8 matrix. In all character densities, the bottom row of pixels is most often used as spacing between lines and may be used for descenders, etc. In the 64-character size, there is no space printed between characters. In the 50-character size, a single row of pixels placed between the characters by the computer cannot be changed. In the 42-character size, the right-hand column of the character data is usually the space between the characters and may be changed by the user. Finally, in the 32-character size, the rightmost column is a space and the computer also adds two columns of space on the leftmost side of the character.

DEMO PROGRAM (DEMO)

A complete, comprehensive, and colorful demo is provided for your convenience. It is written in Basic and requires 32K. Simply load and run the demo after Screen Machine has been executed. The demo is well structured and documented with remarks for ease of comprehension. The few minutes spent running through the selections and examining the listing of this self-executing, menu-driven demo will save you hours in learning how to apply Screen Machine's many advanced features. A demo is worth a thousand bytes!

This section was written for those who wish to use Screen Machine with machine language programs or just want some more technical information on Screen Machine.

Screen Machine is written in almost 100% position-independent code so that the same core routines can be used for all memory configurations. It is not 100% position independent because it must be loaded on 256-byte boundaries (with respect to the end of the program). This is because Screen Machine reserves one page (256 bytes) in its program area for variable storage and uses direct page addressing to get at these variables. This method was chosen because it is the most memory efficient and fastest addressing mode available on the 6809 (except Inherent addressing, of course). For these same reasons, Basic also uses this method, so the machine language programmer must be very careful in his treatment of the DP register. Basic can crash quite easily if you are not careful with the DP register. Screen Machine, on the other hand, is very safe from this kind of tampering, and only in one situation could you crash it by changing the DP register (more on that later).

After you specify a memory size at the loader menu, Screen Machine proceeds to move itself to the top of memory and patch into Basic. When it is done, it is actually a part of Basic and will stay that way until Basic cold-starts again. There are several ways that Screen Machine patches into Basic, the first of which is the "RAM hook."

The RAM hook is something that Basic uses so that other programs (EXTENDED BASIC and DISK BASIC included) modify the way certain operations are handled. Basic places the hooks in RAM when it cold-starts and does not modify them after that. Each hook is 3 bytes long and is either a RTS instruction followed by two zeros or a JMP back into Basic. At certain places in Basic, you will find subroutine calls (JSRs) to these hooks; normally, this will cause an immediate return or a jump to some routine in Basic. Screen Machine, however, modifies some of these hooks so that it can grab control of certain key operations. The hooks it modifies are listed below.

$167: CHROUT hook. Used to intercept characters sent to the text screen. The intercepted characters are sent to Screen Machine instead (if DEVNUM ($006F) is zero) and are printed on the graphics screen.

$168A: GETCHR hook. Normally, this hook jumps to a Basic routine that scans the keyboard, flashes the cursor, and keeps doing this until a key is pressed. Screen Machine modifies this hook so that it jumps to a routine that scans the keyboard, flashes Screen Machine's cursor, looks for a shift/clear or clear key, and keeps doing this until a key is pressed.

$164: COMMA field hook. Provides Screen Machine with a way to modify the comma field widths for the different densities and PMODEs.

$191: INTERPRET loop hook. Screen Machine modifies this hook so that Screen Machine's version of the Interpret loop can be run, which provides for break key control as well as shift/clear control during program runs.

Another way Screen Machine patches into Basic is by modifying COLOR BASIC's jump table. The memory location $123 contains the address of COLOR BASIC's jump table in ROM, and $120 contains the number of entries in the table. This table is a list of addresses for the routines that execute all of COLOR BASIC's commands. By changing the address in memory location $123, Screen Machine forces COLOR BASIC to use Screen Machine's jump table instead. This jump table is identical to COLOR BASIC's table, except the addresses for the PRINT and CLS commands have been modified to point to Screen Machine's routines instead.

The last way that Screen Machine patches into Basic is by changing the pointer to EXTENDED BASIC's command processor routine. This pointer. is located at $12D and points to the routine in ROM that EXTENDED BASIC uses to process the different EXTENDED BASIC tokens. Screen Machine changes this pointer so that the PMODE and COLOR commands can be trapped to change Screen Machine's parameters as well as Basic's.

Using Screen Machine from a machine language program is almost as easy as using it from Basic. To print a character, just use the CHROUT routine as specified in the Basic and EDTASM+ manuals. To get a character from the keyboard, just do a JSR $A171. This routine will flash the cursor and return when a key is pressed. The character will be in the A register. Sending a character 27 followed by a Screen Machine command will do the same thing as it will in Basic.

Since machine language programmers do not have easy access to CLS, PMODE, COLOR, etc, a special routine has been provided to help out. The routine is called SMCOM and is called by doing a JSR ($3FA4) on a 16K configuration, a JSR ($7FA4) on a 32K configuration, and a JSR ($FEA4) on a 64K configuration. On entry, the A register is used to tell SMCOM routine you wish to execute:

0 = GCOLOR. Call this routine whenever you change the foreground or background color.

1 = MODE. Call this routine whenever you switch screens (as in SCREEN1,1 or SCREEN0,0).

2 = PCLS. Clears the graphics screen to the current background color.

3 = GPARAM. Call this routine whenever you change graphics modes.

The only register that is preserved is the DP register. You may also need to access some of Screen Machine's variables, which are stored on the last page at the top of your memory ($3F00 for 16K, $7F00 for 32K, and $FE00 for 64K). The following is a partial map of this area for your convenience.

-- $BE = cursor location. This is a 2-byte number which is the PRINT@ location of the cursor, so a 0000 means the cursor is homed.

--   E3 = screen length. This is a 2-byte value that tells you how many character positions there are on the current screen. Do NOT modify this location!

-- $E1 = line length. This is a 1-byte value that tells how many characters will fit on one line. Do NOT modify this location!

-- $E6 = lines per screen. This is a 1-byte value that tells how many lines will fit on one screen. Do NOT modify this location!

-- $C7 = text graphics flag. Placing a 00 value in here and calling MODE will flip to the Basic's normal text screen, but any characters sent will still go to the graphics screen. A value of $FF will flip back to the graphics screen.

-- $A2 = the address of the USer table, which will be explained a little later.

The following are locations of Basic memory which affect the operation of Screen Machine. These locations reside on absolute page zero.

-- $00B6 = PMODE. This is a 1-byte value from 0 to 4.

-- $00B2 = foreground color. This is a value from 0 to 3.

-- $00B3 = background color. This is a value from 0 to 3.

-- $00C1 = color set. A value of 0 is like a SCREEN #,0 and a value of 8 is like a SCREEN #,1.

USer Command

The US command was provided so that machine language programmers can make routines that are triggered via the shift/clear or PRINT CHR$(27) procedures. US0 is reserved for Screen Machine; however, US1 through US9 are free for use by the user. To patch in your own routines, you must first find the base address of the user table (see limited Screen Machine memory map) and then put the address of your routine in the table. The example below should help.

```
LDD  #ROUTINE  Now D has the addr. of our routine
LDX  $7FA2     Get base address of table for 32K STD
STD  4,X       Put addr. in US3 slot
```

When control is passed to your routine, these registers will have the following values:

-- DP = page number of variable page. Do not modify this register or you could CRASH Screen Machine!

-- X = pointer to next parameter in command line.

Your routine will be expected to advance the pointer 1 byte past its last parameters (if it has any) and to do its own error checking and reporting. If you do not want any more commands processed after your routine is done (like after an error is reported), do a LEAS 2,S before you return (using a RTS).

FINAL COMMENTS

Screen Machine can be used in games, word processors, utilities, etc. In addition, the custom graphics characters can be used to develop easy, effective hi-res character-graphics programs. The potential is truly unlimited.

Screen Machine is fully interfaced with all keys and commands. Although some Basic programming knowledge is recommended, just a few minutes spent studying and referencing your computer's Basic manuals will turn you on to the power of computing with Screen Machine.

Screen Machine can be used to directly create video recorder title screens or large lettering for children or the visually impaired simply by typing:

<SHIFT/CLEAR> DN32 <ENTER> PMODE1:SCREEN1,1:CLS <ENTER>

after execution and typing COLOR0,1 or COLOR2,1 or COLOR3,1 for red, green, or blue lettering. PMODE3 will cut the height of the lettering in half. Simple Basic programs can be created to produce animation, scrolling, and other effects.

Screen Machine can also be used to run many Model I and Model III Basic programs on the Color Computer, since all PRINT@ locations are functionally compatible in the 64-cpl density. In addition, you may re-create the special graphics characters using the supplied character generator program.

Software authors wishing to use Screen Machine in their marketed programs should contact Rainbow Connection Software regarding licensing rights and details of custom files.

Programs may be marketed that require the end user to first load and execute his Screen Machine utility before loading the author's program. We ask only that you first send us a copy so we may assist you with any technicalities and marketing information. We would, of course, be interested in publishing and marketing high-quality Screen Machine compatible software.

Screen Machine, like its predecessor Rainbow-Writer, is fast becoming the industry standard, and many more published programs will be Screen Machine compatible.
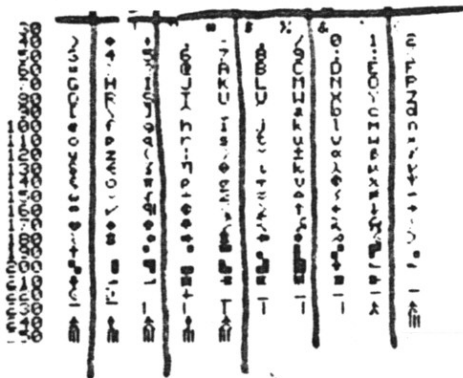
Ukr Characters   Jan 7/83

(1) CLOADM → SM

2) CLOAD → ⎰
              ⎱ CHARGEN
     RUN

3) Don't run DEMO

4) Run own program eg TANYA math game

            ‾ ✓

which is saved as UKRTANYA — on that tape

PRINT @

Screen Machine maintains standard Color Computer Print @ format for ease of operation.

To determine the proper Print @ location for any given row on the screen, just read the corresponding left margin location for your PMODE and character density.

Left Margin Print @ Screen Locations

ROW OR LINE #	STANDARD PMODE 4				PMODE 2+3 AND DW 4				PMODE 6			
	64	50	42	32	64	50	42	32	64	50	42	32
1	0	0	0	0	0	0	0	0	0	0	0	0
2	64	50	42	32	32	25	21	16	32	25	21	16
3	128	100	84	64	64	50	42	32	64	50	42	32
4	192	150	126	96	96	75	63	48	96	75	63	48
5	256	200	168	128	128	100	84	64	128	100	84	64
6	320	250	210	160	160	125	105	80	160	125	105	80
7	384	300	252	192	192	150	126	96	192	150	126	96
8	448	350	294	224	224	175	147	112	224	175	147	112
9	512	400	336	256	256	200	168	128	256	200	168	
10	576	450	378	288	288	225	189	144	288	225	189	
11	640	500	420	320	320	250	210	160	320	250	210	
12	704	550	462	352	352	275	231	176	352	275	231	
13	768	600	504	384	384	300	252	192				
14	832	650	546	416	416	325	273	208				
15	896	700	588	448	448	350	294	224				
16	960	750	630	480	480	375	315	240				
17	1024	800	672		512	400	336					
18	1088	850	714		544	425	357					
19	1152	900	756		576	450	378					
20	1216	950	798		608	475	399					
21	1280	1000	840		640	500	420					
22	1344	1050	882		672	525	441					
23	1408	1100	924		704	550	462					
24	1472	1150	966		736	575	483					

Notice the 32-character density contains only 16 rows in PMODE 4 for direct compatibility with the standard Color Computer text screen. Certain modes of 64-character density offer near compatibility with 24-line capability.

SUPER
SCREEN MACHINE

The Rolls Royce of graphics/text screen enhancers — more features than all others combined!

- Add these features to your computer/program: ML extension of Basic loads on top of 16, 32, or 64K machines to enable easy mixture of hi-res graphics and text in your programs. Dense text or large lettering for children, visually inspired or VCR title screens with no programming!

- User definable 224 character set featuring lower case descenders, Greek, cars, tanks, planes, etc., completely interfaced with all keys, commands, and PMODES. 12 sizes (most colored) from 16x8 to 64x24.

- 2 distinct character sets automatically switch for sharpest lettering featuring underline, subscript, superscript, reverse video, top and bottom scroll protect, double width, colored characters in PMODE 4, and help screen.

- Simple 2-letter abbreviated commands inside your program or control key entry from keyboard, even during program execution!

- Includes demo program, character generator program and manual.

- Super Screen Machine adds SMOOTH Scroll, Key Click, Break Disable, Screen Dump command and more.