

OS9 - System Utilities

S-Screen Control

M-Menuing system

P-Point and Shoot File Selection

**(c) 1987 r3 Systems
4072 E. 22nd St.
Suite 178**

**Tucson, Arizona 85711
1 (602) 745-2327**

S - SCREEN CONTROL UTILITY
by Rick R. Roth

(c) 1987 r3 Systems
4072 E. 22nd St.
Suite 178
Tucson, Arizona 85711
1 (602) 745-2327

PLEASE READ THIS

The author has taken due care in preparing this documentation and the programs and data on the electronic media accompanying this documentation including research, development and testing to ascertain their effectiveness.

The author makes no expressed or implied warranty of any kind with regard to these programs nor the supplemental documentation. In no event shall the author be liable for incidental or consequential damages in connection with or arising out of the furnishing, performance or use of this program.

Licence Statement

r3 Systems S - Screen Control Utilities, in all machine readable formats, and the documentation, both printed and on disk, accompanying them are copyrighted. The purchase of R3 Systems S - Screen Control Utilities conveys to the purchaser a license to use R3 Systems S - Screen Control Utilities for his/her own use, and not for sale or free distribution to others. No other license, expressed or implied is granted.

S and S09 copyright 1987 by r3 Systems. Distribution licenses are available for a nominal fee if you wish to include S09 in programs you wish to distribute.

OS9 and Basic09 are trademarks of Microware and Motorola.

S - SCREEN CONTROL UTILITY

INTRODUCTION

S is a small (509 byte) machine language program designed to be merged in with the shell. When S is merged in the shell file you gain almost complete control of your OS9 text screen. This is accomplished with 37 easy to remember mnemonic commands. Included with S is another small machine language program S09. S09 is a Basic09 interface that allows you to call S from within a Basic09 program without the overhead of the SHELL command. The instructions for S09 are given in a separate section.

S

Syntax: s opts [opts [...]]

Function: Lets you control your screen color and appearance.

Options:

The options or s commands listed below are divided into several related groups, each option must be separated from other options by a space.

Screen Size Options

40 Resets the screen to the 40 column mode.

80 Resets the screen to the 80 column mode.

These options restore the default color clear the screen, home the cursor, reset the foreground, background, and border colors to the preset S defaults, and converts the screen to either the 40 or 80 column mode.

Screen Color Options

- CF n Set the Foreground color to n.
CB n Set the Background color to n.
BC n Set the Border color to n.
DC Resets the palette to the Default colors.

The values for n are:

0	White	4	Red
1	Blue	5	Yellow
2	Black	6	Magenta
3	Green	7	Cyan

Screen Type Options

- MC Set the screen to Montype composite.
MR Set the screen to Montype RGB.
MM Set the screen to Montype Mono
chrome.

These options duplicate the Montype command.

Cursor Position Options

- HC Home Cursor
CR Cursor Right
BS Backspace or Cursor Left
CU Cursor Up
CD Cursor Down
LF Line feed moves the cursor one line down and to the
left edge of the screen.

AT n n Moves the cursor to the position specified by the numbers entered. Each number MUST consist of 2 digits ie. 02 not 2. The first number is the X coordinate, and the second the Y coordinate.

Cursor Display Options

- CI Cursor Invisible turns off the cursor.
CV Cursor Visible turns the cursor on.

Screen Clear Options

CS	Clear the screen and homes cursor.
CL	Clear to the end of line.
CE	Clear to the end of screen.
	CL and CE clear from the current cursor position.
DL	Deletes the current line.

Character Display Options - General

BB	Begin Blinking characters.
EB	End Blinking characters.
BR	Begin Reverse characters.
ER	End Reverse characters.
BU	Begin Underlining characters.
EU	End Underlining characters.
'text'	Echos the text string enclosed in single quotes to the screen.

Character Display Options - Graphic Screen

BO	Bold On Turns on Bold characters.
BF	Bold Off turns off Bold characters.
PO	Proportional ON.
PF	Proportional Off.

Window Options

EW	End current window.
EO	End overlay window.

Miscellaneous Options

RB	Ring terminal bell.
Sn	Sleep n x 100 ticks similar to the level 1 sleep command. S0 will cause the shell to sleep until a signal is received (BREAK or CONTROL-C). This will effectively lock the keyboard.

Notes:

S will accept as many options as your version of the shell will pass on the command line . The versions of shell I have used have a limit of approximately 200 characters.

Because of space limitations when coding this program error trapping is minimal, if S encounters a syntax error in the command line string, S will exit without an error at the point where the error is detected. Each option must be preceded by one and only one space and multiple options must be separated by one and only one space. Options can be entered in either upper or lower case but case should not be mixed in an option. ie. RB or rb will ring the bell, rB or Rb will be an error and S will exit at this point.

Examples:

S rb rb cf 4 cb 0 bc 0 cs bb at 20 10 'TTThat's all folks'

This will ring the bell twice, set the foreground to red, the background and border to white, clear the screen and print TTThat's all folks on the screen at a position 20 characters across and 10 lines down in blinking characters. Everything printed after this will also blink until a S eb is entered.

This includes the prompt.

S09

Syntax: RUN S09(string[,string])

Function: S09 will pass a string containing a valid S command line without the overhead of calling another shell with the SHELL keyword. If the second optional string is included S09 will read the keystrokes entered and store them in the specified string variable. If the string variable is dimensioned as a string of 1 character, S09 will read a single keystroke, and then return the key pressed converted to upper case in the variable specified.

Parameters:

string a string containing a valid S command line.
string1 a string variable.

Examples:

```
DIM S_out:STRING[255]
S_out="cf 4 cb 2 bc 2 cs"
RUN S09(S_out)
```

```
DIM S_out:STRING[255]; S_in:STRING[20]
S_out="cf 4 cb 2 bc 2 cs at 20 10 'Please enter your name: '"
RUN S09(S_out,S_in)
PRINT
PRINT "Hello ";S_in
```

```
DIM S_out:STRING[255];S_in:STRING[20]
S_out=chr$(1b)+chr$(20)+chr$(4)+chr$(1b)+
chr$(21)+chr$(2)+chr$(1b)+chr$(22)+ chr$(2)+chr$(0c)+
chr$(2)+ chr$(32+20)+ chr$(32+10)
PRINT S_out;"Please enter your name: '"
INPUT S_in
PRINT
PRINT "Hello ";S_in
```

Both these fragments will do the same thing, ie clear the screen to red characters on a black screen, prompt you to enter your name, accept the input and then print the greeting.

```
DIM S_in:STRING[1]
DIM S_out:STRING[255]
S_out="CS AT 20 10 BU 'MAIN MENU' EU AT 20 12 'A. FIRST
SELECTION' AT 20 14 'B. SECOND SELECTION' AT 20 16
'Please Select: '"
RUN S09(S_out,S_in)
```

This fragment will print a menu on a clear screen, and prompt for a selection and then convert the key pressed to upper case and return the value in the variable S_in.

S QUICK REFERENCE CHART

(c) 1987 r3 Systems

COMMAND	DESCRIPTION
=====	=====
CF #	Color Foreground
CB #	Color Background
BC #	Border Color
MC	Montype Composite
MR	Montype RGB
MM	Montype Monochrome
80	80 Column screen
40	40 Column screen
RB	Ring Bell
HC	Home Cursor
CL	Clear to end of line
CE	Clear to end ofscreen
CS	Clear Screen
DL	Delete Line
BB	Begin Blinking
EB	End Blinking
BR	Begin Reverse
ER	End Reverse
BU	Begin Underline
EU	End Underline
PO	Proportional On
PF	Proportional OFF
BO	Bold On
BF	Bold Off
EO	End Overlay
EW	End Window
DC	Default Color
CU	Cursor Up
CD	Cursor Down
CI	Cursor Invisible
CV	Cursor Visible
BS	Backspace
LF	Line Feed
S#	Sleep
AT nn nn	Move cursor to nn nn
'string'	Echo string

Color #	Color	Color #	Color
=====	=====	=====	=====
0	White	4	Red
1	Blue	5	Yellow
2	Black	6	Magenta
3	Green	7	Cyan

