

Radio Shack TRS-80 emulator for Raspberry Pi

User Guide:

=====

For writing image to micro sd card, I recommend Etcher, with a very user friendly interface.

Here are several version of Etcher for download, depending on the OS version you are using:

<https://www.balena.io/etcher/>

<https://www.filehippofilex.com/etcher/>

Its even support zip packed files, so you dont need to unpack your img file at all.

Just drag and drop zip image file to Etcher, click on Flash button and wait for it to finish.

Its very important that you have a quality micro sd card, which leaves as much free space as described.

Cheap cards often leaves slightly less space than the declared and then there may not be enough space to write a disc image.

In this case, a quality 4 Gb micro sd card (genuine SanDisk or Kingston) or larger, will be quite enough.

This image was designed to work on any LCD TV full HD or HD ready, with HDMI connectivity.

Emulator controls:

=====

F7 - Main Menu

F8 - Exit emulator

F10 - Soft Reset, SHIFT/F10 - Hard Reset

ESC - Break

Example of loading programs:

=====

To view content of e.g Drive 1, type DIR 1 and press Enter.

To start a game, just type its name (without extension after / sign) and press Enter.

For example, type meteor and you will start "Meteor Mission 2" game from disk 1.

Similarly, you can start programs from other disk drives.

Use the same procedure if you want to load programs from other drives.

To change default model at boot, press F7 to access Menu, go to "TRS-80 Model Selection" menu and choose model you want to emulate.

Go back to the Menu, choose "Configuration/State File Management Menu" and then "Write Configuration File".

Type "sdltrs" (without quotes) and press Enter. From now on your custom configuration was saved and will take a effect after next reboot.

Don't forget to change the coresponded version of the DOS floppy image in drive 0, according to the TRS-80 model you have selected.

It is highly recommended that exit from the emulator be done regularly with F8 from the Main Menu, for the complete shutdown process.

Main Menu usage:

=====

The main menu is the entry point for the Text GUI. You may move up and down through the menu items using the up and down arrow keys. A menu item may be selected by pressing Enter when the item is highlighted. Pressing ESC moves up a menu level (or exits the GUI if you are at the top level). Besides entering the GUI using F7, submenus of the GUI may be entered by using the key shortcuts show on the first three menu items.

Floppy Disk Management Menu:

=====

The floppy disk management menu allows you to insert and remove floppy images into the emulated drives, as well as save and load disk sets, create new disk images, and set emulated disk drive options.

For the first 8 options in the menu, there are three options:

Hitting Enter while the item is selected, will move to the image selection screen, which is explained below.

Hitting Backspace will remove a disk image if there is one in the drive.

Hitting Space will toggle the write protect for the drive. The indicator for write protect is an asterisk "*" in front of the "Disk" in the menu item.

Save Disk Set

This ote, allows the user to save the names of the disk images that are currently in the drives to a file, to be loaded later. The file that the set is save in has an extension of ".set", and it is a human readable text file, containing the paths of the image files, or a blank line for a drive for a drive that has no disk.

Load Disk Set:

This button allows the user to load a disk set saved earlier with the Save Disk Set Button. The user is allowed to browse for the set file, displaying files ending in ".set". The emulator then loads the disk images specified in the file into the corresponding drives. If a drive in the set file is a blank line, no changes are made to the drive. This allows the user to load multiple disk sets sequentially.

When the user presses Enter in on a Drive item in the Floppy Disk Management screen, the file open menu is displayed. The up and down arrows may be used to scroll through the list of files. Pressing Enter on the <..> item will take you to the parent directory. On Windows, the Drive letters are listed as the last items in the menu, and you can change drives by pressing Enter on those items. You may also go to a particular section of the file list my pressing the first letter of the file name.

In the Floppy Creation screen, you may choose a format of a disk. It may be created in either the JV1, JV3, or DMK disk image format (See the Features page for info on the disk formats).

If the "Insert Created Disk Into This Drive" is set to something besides none, the created image will be mounted on the specified floppy

drive.

Doubler Type:

Specify what type of double density adaptor to emulate (Model I mode only). The type may be percom, radioshack (or tandy), both, or none. The type may be abbreviated to one character. The default is both, which causes the double density adaptor emulation to respond to the special commands of both the Percom and Radio Shack cards.

True DAM Emulation:

Turn off the single density data address mark remapping kludges described in the "Emulated floppy disks" and "Real floppy disks" sections above. With this option given, the distinction between F8 and FA data address marks is strictly observed on both writing and reading. This option is probably not useful unless you need to deal with Model I disks that use the distinction as part of a copy-protection scheme. See also "Common File Formats for Emulated TRS-80 Floppy Disks", available at <http://www.tim-> Finally, the last menu item allows you to set the size of each disk drive, as either a 5" or 8" drive.

Hard Disk Management Menu:

=====

The hard disk management menu allows you to insert and remove floppy images into the emulated drives, as well as save and load disk sets and create new disk images.

For the first 4 options in the menu, there are three options:
Hitting Enter while the item is selected, will move to the image selection screen, which is explained above in the floppy disk section.
Hitting Backspace will remove a disk image if there is one in the drive.
Hitting Space will toggle the write protect for the drive. The indicator for write protect is an asterisk "*" in front of the "Disk" in the menu item.
Save Disk Set

This option, allows the user to save the names of the disk images that are currently in the drives to a file, to be loaded later. The file that the set is save in has an extension of ".set", and it is a human readable text file, containing the paths of the image files, or a blank line for a drive for a drive that has no disk.

Load Disk Set:

This button allows the user to load a disk set saved earlier with the Save Disk Set Button. The user is allowed to browse for the set file, displaying files ending in ".set". The emulator then loads the disk images specified in the file into the corresponding drives. If a drive in the set file is a blank line, no changes are made to the drive. This allows the user to load multiple disk sets sequentially.

In the Create Hard Disk Image window, you may choose a format of a disk. (See the Features page for info on the hard disk formats).

If the "Insert Created Disk Into This Drive" is set to something besides none, the created image will be mounted on the specified hard drive.

Cassette Management Menu:

=====

The cassette management menu allows you to insert and remove cassette images into the emulated drive, as well as create new cassette images.

For the first options in the menu, there are two options: Hitting Enter while the item is selected, will move to the image selection screen, which is explained above in the floppy disk section. Hitting Backspace will remove a cassette image if there is one in the drive.

The second option allows you to set the position of the cassette tape by changing the Position field. Valid values are 0 (fully rewound) through the number of bytes on the tape (at end of tape).

The final option allows you to create a blank cassette image.

In the Create Cassette Image window, you may choose a format of a cassette (.cas or .wav).

If the "Insert Created Disk Into This Drive" is set to Yes, the created image will be mounted on the specified hard drive.

TRS-80 Model Selection Menu:

=====

TRS-80 Model:

Specifies what TRS-80 model to emulate.

Hi-Res Graphics:

In Model III or 4/4P mode select which hi-res graphics card to emulate, the Micro Labs Grafyx Solution hi-res graphics card or Radio Shack hi-res card. In Model I mode, the HRG1B 384x192 hi-res graphics card is emulated. Note that the Model III and Model 4/4P cards from Micro Labs are very different from one another.

Configuration/State File Management Menu:

=====

Save State (alt-S or cmd-S)

This menu item allows you to save the state of the emulator, so that it may be loaded later with the Load State command. Disk images inserted into the computer are stored in the state file.

Load State (alt-L or cmd-L)

This menu item allows you load a state file (.t8s) previous saved with the Save State command.

Write Configuration (alt-W or cmd-W)

This menu item allows you to save the configuration of the emulator, so that it may be loaded later with the Load Configuration command. Configuration files (.t8c) are discussed further on the Features page.

Read Configuration (alt-R or cmd-R)

This menu item allows you load a Configuration file (.t8c) previous saved with the Save Configuration command.

Printer Management Menu:

=====

The Printer Management menu controls the type of printer emulation that is used for outputting to the printer TRS-80.

There are three choices for printer emulation, None, Text Printer, and Epson FX-80. The Text printer simply sends the printer output to a text file with optional script processing. The Epson option is only available on the Macintosh, and it emulates a legacy printer, and allow the user to save the printer output in a PDF file.

Close and Reopen Printer Output File:

Selecting this option will cause the printer output file to be closed, and the printer command to be executed using the file name.

Print Command:

The only option for the text printer type is the command used to print. Within the command, %s is used to represent the file name of the temporary file generated during printing. On Linux, this defaults to "lpr %s", on Windows "notepad %s", and on the Mac "open %s". The %s in the printer command will be replaced with the name of the printer output file. (On the Mac, this is for the text printer only, not the Epson emulation).

Default Directories Menu:

=====

This menu allows the user to specify the default directories used for loading/saving various types of file used by the emulator. To choose the directory, press the Enter when the appropriate entry is highlighted. That will bring up the Directory select screen. The up and down arrows may be used to scroll through the list of directories. Pressing Enter on the <..> item will take you to the parent directory. On Windows, the Drive letters are listed as the last items in the menu, and you can change drives by pressing Enter on those items. The directory is selected by pressing the Tab key.

ROM Files Menu:

=====

This menu allows you to chose the ROM files used in the emulation of the Atari. These files are not provided with the emulator. To select the file, press Enter when the appropriate item is highlighted, and the file selection menu will appear. It's operation is explained in the Floppy Disk section above.

Display Settings Menu:

=====

Emulator Colors:

Specifies the foreground and background color of the sdltrs window. Default is black (0x000000) for foreground and white (0xFFFFFFFF) for background.

Fullscreen GUI Colors:

Specifies the foreground and background color of the fullscreen Text GUI. Default is green (0x008010) for foreground and white (0xFFFFFFFF) for background.

Character Sets:

Select among several sets of built-in character bitmaps. The three different options allow you to specify the character set used for each model, should you change models during sdltrs execution using the GUI. In Model I mode , five sets are available. The default, wider, is a modified Model III set with characters 8 pixels wide; it looks better on a modern computer screen with square pixels than the real Model I fonts, which were 6 pixels wide. lcm0d is the character set in the replacement character generator that was supplied with the Radio Shack lower case modification. (It was reconstructed partly from memory and may have some minor bit errors.) stock is the character set in the stock character generator supplied with most upper case only machines. Since sdltrs currently always emulates the extra bit of display memory needed to support lower case, this character set gives you the authentic, unpleasant effect that real Model I users saw when they tried to do homebrew lower case modifications without replacing the character generator: lower case letters appear at an inconsistent height, and if you are using the Level II BASIC ROM display driver, upper case letters are replaced by meaningless symbols. early is the same as stock, but with the standard ASCII characters [, \,], and ^ in the positions where most Model I's had directional arrows. This was the default programming in the Motorola character generator ROM that Radio Shack used, and a few early machines were actually shipped with this ROM. Finally, german or genie gives an approximate emulation of the GENIE, a German Model I clone. Characters are 8 pixels wide, and double width is supported even though later GENIE models did not include it.

In Model III , 4 and 4P (modes, three sets are available: katakana (the default for Model III) is the original Model III set with Japanese Katakana characters in the alternate character positions. This set was also used in early Model 4's. international (the default for

Model 4 and 4P) is a later Model 4 set with accented Roman letters in the alternate positions.

bold is a bold set from a character generator ROM found in one Model III, origin uncertain.

Window Border Width:

Put a border of width pixels around the TRS-80 display. The default is 2.

Resize Window on Model 3 Mode Change:

In Model 3 mode, resize the window whenever the emulated display mode changes between 64x16 text (or 512x192 graphics) and 80x24 text (or 640x240 graphics).

Resize Window on Model 4 Mode Change:

In Model 4/4P mode, resize the window whenever the emulated display mode changes between 64x16 text (or 512x192 graphics) and 80x24 text (or 640x240 graphics).

Disk Leds:

This option allows the user to chose if disk activity indicators are shown in the TRS-80 Emulator window (lower edge).

Joystick Settings Menu:

=====

Use Numeric Keypad for Joystick

This item determines if the numeric keypad is used for emulation of a joystick.

USB Joystick/Gamepad

This item allows the user to select which USB joystick, if any, is used for emulation of the TRS-80 Joystick. It will bring up a list of the USB joysticks that SDL is aware of as shown below:

Miscellaneous Settings Menu:

=====

Shift Bracket Emulation:

Chose Shift Bracket Emulation on or off. If on, emulate [, \,], ^, and _ as shifted keys, and {, |, }, and ~ as unshifted. This is the default in Model 4 and 4P modes, and it works well with the keyboard driver in Model 4 TRSDOS/LS-DOS 6. If off, emulate [, \,], ^, and _ as unshifted keys, and {, |, }, and ~ as shifted. This is the default in Model I and III modes, and it works well with many TRS-80 keyboard drivers. With some keyboard drivers these keys do not work at all, however.

Turbo Mode/b>:

This will turn Turbo mode on or off. When turbo mode is on, the emulator will run faster than a normal TRS80, by a factor of Turbo Rate, which is explained below, and also set on this page. The F11 key may be used to switch Turbo mode on or off while running the emulator.

Turbo Speed:

Set the number of times normal TRS80 speed that the emulator runs when in "Turbo" mode. Turbo mode can be used to speed compute intensive applications, compiling source code for example, on the emulator. The default setting is 5x. Above around 8x you may experience problems with runaway keyboard repeat on the emulator, so use higher values with caution.

Keystretch Value:

Fine-tune the keyboard behavior. To prevent keystrokes from being lost, sdltrs "stretches" the intervals between key transitions, so that the Z-80 program has time to see each transition before the next one occurs. Whenever the Z-80 program reads the keyboard matrix and sees an emulated key go up or down, sdltrs waits cycles Z-80 clock cycles (T-states) before it allows the program to see another key transition. Key transitions that are received during the waiting period or when the Z-80 program is not reading the keyboard are held in a queue. The default stretch value is 4000 cycles; it should seldom if ever be necessary to change it.

Emts safe

Turn on/off ability for emts (Emulation traps) to write to unexpected places in the host filesystem.

Serial Port Switches:

Set the sense switches on the Model I serial port card. This option is meaningful only in Model I mode, and only when the Serial Port Name is not set to "". The default value is 0x6f, which Radio Shack software conventionally interprets as 9600 bps, 8 bits/word, no parity, 1 stop bit.

Serial Port Name:

Set the tty device to be used for I/O to the TRS-80's serial port. The default is "". Setting the name to be empty emulates having no serial port.

Emulator overview:

=====

Sdltrs is built on top of a Z-80 emulator, with added routines to support keyboard and video I/O through a SDL interface. The hardware emulation can operate as a TRS-80 Model I, Model III, Model 4, or Model

4P.

sdltrs supports 48K of RAM in Model I or Model III mode, 128K in Model 4 or Model 4P mode. Floppy disks and hard disks are emulated using files to store the data; or under Linux only, real floppy drives can be used.

A printer is emulated by sending its output to a text file. A serial port is emulated using a host serial port. Cassette I/O is emulated using files to store the cassette data. Game sound and music output are also supported. Though the cassette port, through the Model 4 sound option, and through the optional Orchestra-85/90 music synthesizer card are all emulated. In Model I mode, the HRG1B graphics card is emulated. In Model III and 4/4P mode, you can select whether the Radio Shack graphics card or Micro Labs Grafyx Solution is emulated. There is also a mouse driver for model 4/4P mode. Several common time-of-day clock cards are emulated on all models. The Alpha Products joystick is emulated using the numeric keypad, or through a SDL compatible joystick.

Because sdltrs emulates the hardware, all known TRS-80 Model I/III/4/4P operating systems should run on it, including all flavors of TRSDOS, LDOS/LS-DOS, NEWDOS, DOSPLUS, MultiDOS, and TRS-80 CP/M. However, the emulator also includes some extensions to the standard hardware, and the special drivers, utilities, and instructions needed for these are not provided for all operating systems.

Special support in the emulator allows the program to block when waiting for information from the keyboard. This will work only for programs that wait for keyboard input using the standard Model I/III ROM call; the emulator decides whether to block the Z-80 program when it tries to read from the keyboard memory by pattern-matching its stack.

Keys:
====

The following keys have special meanings to sdltrs:

F1 - TRS-80 Model 4/4P F1 key (address bit 7, data bits 4), TRS-80 Model 1 Electric Pencil control key
F2 - TRS-80 Model 4/4P F2 key (address bit 7, data bits 5)
F3 - TRS-80 Model 4/4P F3 key (address bit 7, data bits 6)
F4 - TRS-80 Model 4 Caps Lock key (address bit 7, data bit 3)
F5 or ScrollLock - TRS-80 @ key
F6 - TRS-80 '0' key (so that a shifted 0 can be obtained)
F7 - Enter Text GUI
F8 - Exit SDLTRS
F10 - Warm Reset
Shift F10 - Power on / Hard Reset (reboot)
ESC - Break

Left Arrow or Backspace or Delete - TRS-80 left arrow key
Right Arrow or Tab - TRS-80 right arrow key
Up Arrow - TRS-80 up arrow key (caret for exponent)
Down Arrow - TRS-80 down arrow key

Home or Clear - TRS-80 clear key
 Control - TRS-80 Model 4 Ctrl key (address bit 7, data bit 2)
 RightAlt - TRS-80 shifted down arrow key (used as a control key with some TRS-80 software)
 Page Up - TRS-80 Left Shift key (address bit 7, data bit 0)
 Page Down - TRS-80 Right Shift key (address bit 7, data bit 1)
 End Key - TRS-80 Unused key (address bit 7, data bit 7)
 Insert Key - TRS-80 Underscore key (address bit 3, data bit 7)
 Shift UP Arrow - TRS-80 ESC key
 Ctrl + c - Copy from TRS-80 to host
 Ctrl + v - Paste from TRS-80 to host
 Ctrl + a - Select All on TRS-80 screen
 LeftAlt + '=' (think +) (or Left Cmd + '=') - Set windowed mode scaling to next higher size. (1x, 2x, 3x, and then wraps).
 LeftAlt + '-' - Set windowed mode scaling to next lower size. (3x, 2x, 1x and then wraps).
 LeftAlt + Enter - Switch from Windowed to Fullscreen mode and back.
 Alt + d - Enter Floppy Disk Management GUI
 Shift + Alt + d - Enter Hard Disk Management GUI
 Alt + t - Enter Cassette ('t' for Tape) Management GUI
 Alt + s - Save Emulator State
 Alt + l - Load Emulator State
 Alt + w - Write Configuration File
 Alt + r - Read Configuration File
 Alt + p - Pause Emulator (and then unpause)
 Alt + 1 (or 2,3,4,5,6,7,8) - Insert Floppy disk into drive N
 Shift + Alt + 1 (or 2,3,4,5,6,7,8) - Remove Floppy disk from drive N
 In Model III, 4, and 4P modes, the left and right shift keys are distinct; in Model I mode, they are the same. The keys [, \,], ^, _, {, |, }, and ~ also activate unused positions in the keyboard matrix (address bit 3, data bits 3-7). With many TRS-80 keyboard drivers, these keys map to the corresponding ASCII characters; with others, they do nothing. In some cases you may find the shift state is reversed from what it should be; if you press [but { is displayed instead (etc.), see the -shiftbracket and -noshift-bracket options below to correct the problem. The Insert key maps to the same position as underscore (address bit 3, data bit 7), so that this key can be used both with and without shift pressed; with many TRS-80 keyboard drivers one of these maps to ASCII code 0x7f.

Pressing a key on a numeric keypad with NumLock disengaged emulate the Alpha Products joystick. Keys 2, 4, 6, 8 (KP_Down, KP_Left, KP_Right, KP_Up) are the main directions; keys 1, 3, 7, and 9 (KP_End, KP_Page_Down, KP_Home, KP_Page_Up) work as diagonal directions by activating two main directions at once; and key 0 (KP_Insert) or 5 (KP_Begin) is the fire button.

Emulated cassette:
 =====

To control the emulated cassette, a file for the loaded tape is controlled by the -cassette option or the cassette option in the menu system. There is also a menu option to control the cassette position in the file.

Emulated 5-inch floppy disks:

=====

In Model I mode, sdltrs emulates a Radio Shack Expansion Interface with the Percom Doubler or Radio Shack Doubler installed. The Doubler provides double-density disk access by allowing either the stock WD1771 FDC chip or a WD1791 chip to be selected under program control. At powerup the 1771 is selected, so operating systems with no Doubler driver see a stock system. By default, the emulator pretends to be both a Percom and Radio Shack Doubler at the same time -- it responds to the special commands of both -- so a driver for either should work. Under LDOS, use the command "FDUBL" (newer versions of LDOS), or "PDUBL" or "RDUBL" (older versions) to install the driver. Software that tries to detect which doubler you have (such as Super Utility) may be confused by the emulation of both at once, so you can choose to emulate only one with a command line option or text GUI control.

In Model III, 4, or 4P mode, sdltrs emulates the stock floppy controller, which uses a WD1793 chip (software-compatible with the WD1791) to provide both single and double density.

Four 5.25-inch floppy drives are emulated. Disk image files may be inserted into the emulated drives with command line options, text GUI controls. If the user does not have write permission for a floppy file, and/or the file has an internal write protect flag set, a write-protect tab is emulated. Use the mkdisk(1) program to turn the write protect flag on or off.

If you try to boot an emulated Model I, III, or 4 with no emulated disk (that is, no disk in drive 0), sdltrs emulates having no floppy disk controller. The behavior of a real machine with a disk controller in this case didn't seem useful to emulate faithfully: A real Model I hangs with a screen full of garbage; a real Model III or 4 goes into a retry loop printing "Diskette?" on the screen and rechecking if you've inserted one. A real Model 4P always has a floppy controller, however, so sdltrs always emulates one.

Due to a limitation of the original Model I hardware, drive :3 cannot be double-sided in Model I mode. In the original Model I, you could not have a drive :3 at all if any drive in the system was double-sided, but the emulator is able to be more forgiving.

Emulated floppy image files can be of any of three types: JV1, compatible with Jeff Vavasour's popular freeware Model I emulator for MS-DOS; JV3, a compatible extension of a format first used in Vavasour's commercial Model III/4 emulator; or DMK, compatible with David Keil's Model 4 emulator. All threetypes work in sdltrs regardless of what model it is emulating. A heuristic is used to decide which type of image is in a drive, as none of the types has a magic number or signature.

JV1 supports only single density, single sided, with directory on track 17. Sectors must be 256 bytes long. Use FORMAT (DIR=17) if you want to format JV1 disks with more (or less) than 35 tracks under LDOS.

JV3 is much more flexible, though it still does not support everything the real controllers could do. It is probably best to use JV3 for all the disk images you create, since it is the most widely implemented by

other emulators, unless you have a special reason to use one of the others. A JV3 disk can be formatted with 128, 256, 512, or 1024-byte sectors, 1 or 2 sides, single or double density, with either FB (normal) or F8 (deleted) data address mark on any sector. In single density the nonstandard data address marks FA and F9 are also available. You cannot format a sector with an incorrect track number or head number. You can format a sector with an intentional CRC error in the data field. sdltrs supports at most 5802 total sectors on a JV3 image.

The original Vavasour JV3 format supported only 256-byte sectors, and had a limit of 2901 total sectors. If you use sector sizes other than 256 bytes or format more than 2901 sectors on a disk image, emulators other than sdltrs may be unable to read it. Note that an 80 track, double-sided, double-density (18 sector) 5.25-inch floppy will fit within the original 2901 sector limit; the extension to 5802 is primarily for emulation of 8-inch drives (discussed below).

The DMK format is the most flexible. It supports essentially every thing that the original hardware could do, including all "protected" disk formats. However, a few protected disks still may not work with sdltrs due to limitations in sdltrs's floppy disk controller emulation rather than limitations of the DMK format; see the LIMITATIONS section below.

Blank disks may be created from the Text GUI.

Early Model I operating systems used an FA data address mark for the directory on single density disks, while later ones wrote F8 but would accept either upon reading. The change was needed because FA is a nonstandard DAM that is fully supported only by the WD1771 floppy disk controller used in the Model I; the controllers in the Model III and 4 cannot distinguish between FA and FB (which is used for non-directory sectors) upon reading, and cannot write FA. To deal nicely with this problem, sdltrs implements the following kludge. On writing in single density, an F8 data address mark is recorded as FA. On reading with an emulated WD1771 (available in Model I mode only), FA is returned as FA; on reading with a WD179x, FA is returned as F8. This trick makes the different operating systems perfectly compatible with each other, which is better than on a real Model I! You can use the -truedam flag to turn off this kludge if you need to; in that case the original hardware is emulated exactly.

TRS-80 programs that attempt to measure the rotational speed of their floppy disk drives using timing loops will get the answers they expect, even when sdltrs does not emulate instructions at the same speed as the original machines. This works because sdltrs keeps a virtual clock (technically, a T-state counter), which measures how much time it should have taken to execute the instruction stream on a real machine, and it ties the emulation of floppy disk index holes to this clock, not to real time.

Emulated 8-inch floppy disks:

=====

In addition to the four standard 5.25-inch drives, sdltrs also emulates four 8-inch floppy drives. There is no widely-accepted standard

hardware interface for 8-inch floppies on the TRS-80, so sdltrs emulates a pseudo-hardware interface of its own and provides an LDOS/LS-DOS driver for it.

The only difference between 5.25-inch and 8-inch emulated drives is that the emulator allows you to format more bytes per track in the latter. A new JV3 floppy can be formatted as either 5.25-inch or 8-inch depending on whether you initially put it into a 5.25-inch or 8-inch emulated drive. A new DMK floppy, however, must be created as an 8 inch image in order to be large enough for use in an 8-inch emulated drive. JV1 floppies cannot be used in 8-inch drives. Be careful not to put an emulated floppy into a 5.25-inch emulated drive after it has been formatted in an 8-inch emulated drive or vice versa; the results are likely to be confusing. Consider using different file extensions for the two types; say, .dsk for 5.25-inch and .8in for 8-inch.

To use the emulated 8-inch drives, you'll need a driver. Under LDOS or LS-DOS, use the program XTRS8/DCT supplied on the emulated floppy utility.dsk. This driver is a very simple wrapper around the native LDOS/LS-DOS floppy driver. Here are detailed instructions.

First, make sure an appropriate version of LDOS is in emulated floppy drive 0, and the supplied file utility.dsk is in another emulated floppy drive. Boot LDOS. If you are using Model I LDOS, be sure FDUBL is running.

Second, type the following commands. Here d is the LDOS drive number you want to use for the 8-inch drive and u is the unit number you chose when naming the file. Most likely you will choose d and u to be equal to reduce confusion.

```
SYSTEM (DRIVE=d,DRIVER="XTRS8",ENABLE)
Enter unit number ([4]-7): u
```

You can repeat these steps with different values of d and u to have more than one 8-inch drive. You might want to repeat four times using 4, 5, 6, and 7, or you might want to save some drive numbers for hard drives (see below).

Finally, it's a good idea to give the SYSTEM (SYSGEN) command (Model I/III) or SYSGEN command (Model 4/4P). This command saves the SYSTEM settings, so the 8-inch drives will be available again the next time you reboot or restart the emulator. If you need to access an 8-inch drive after booting from a disk that hasn't been SYSGENed, simply use the same SYSTEM command again.

In case you want to write your own driver for another TRS-80 operating system, here are details on the emulated pseudo-hardware. The 8-inch drives are accessed through the normal floppy disk controller, exactly like 5.25-inch drives. The four 5.25-inch drives have hardware select codes 1, 2, 4, and 8. The four 8-inch drives have hardware select codes 3, 5, 6, and 7, corresponding respectively to files. (See also the -sizemap option below, however.)

Emulated hard disks:
=====

sdltrs can emulate a hard disk in a file in one of two ways: it can use a special, sdltrs-specific LDOS driver called XTRSHARD/DCT, or it can emulate the Radio Shack hard drive controller (based on the Western Digital WD1010) and use the native drivers for the original hardware. Using XTRSHARD/DCT

The XTRSHARD/DCT driver has been tested and works under both LDOS 5.3.1 for Model I or III and TRSDOS/LS-DOS 6.3.1 for Model 4/4P. It may or may not work under earlier LDOS versions. It definitely will not work under other TRS-80 operating systems or with emulators other than sdltrs or xtrs. The hard disk format was designed by Matthew Reed for his Model I/III and Model 4 emulators; sdltrs duplicates the format so that users can exchange hard drive images across the emulators.

To use XTRSHARD/DCT, first create a blank hard drive file using the Text GUI.

Second, load the file into one of the emulated hard drives using a command line option, the Text GUI.

Third, make sure an appropriate version of LDOS is in emulated floppy drive 0, and the supplied file utility.dsk is in another emulated floppy drive. Boot LDOS. If you are using Model I LDOS 5.3.1, patch a bug in the FORMAT command by typing PATCH FORMAT/CMD.UTILITY M1FOR-MAT/FIX. You need to apply this patch only once. It must not be applied to Model III or Model 4/4P LDOS.

Fourth, type the following commands. Here d is the LDOS drive number you want to use for the hard drive (a typical choice would be 4) and u is the unit number you chose when naming the file (most likely 0).

```
SYSTEM (DRIVE=d,DRIVER="XTRSHARD",ENABLE)
Enter unit number ([0]-7): u
FORMAT d (DIR=1)
```

Answer the questions asked by FORMAT as you prefer. The DIR=1 parameter to FORMAT is optional; it causes the hard drive's directory to be on track 1, making the initial size of the image smaller. You can repeat these steps with different values of d and u to have more than one hard drive.

Finally, it's a good idea to give the SYSTEM (SYSGEN) command (Model I/III) or SYSGEN command (Model 4/4P). This command saves the SYSTEM settings, so the drive will be available again the next time you reboot or restart the emulator. If you need to access the hard disk file after booting from a floppy that hasn't been SYSGENed, simply use the same SYSTEM command(s) again, but don't FORMAT. You can freely use a different drive number or (if you renamed the hard disk file) a different unit number.

Technical note: XTRSHARD/DCT is a small Z-80 program that implements all the required functions of an LDOS disk driver. Instead of talking to a real (or emulated) hard disk controller, however, it uses special support in sdltrs that allows Z-80 programs to open, close, read, and write host files directly. This support is described further in the "Data import and export" section below.

Processor speed selection:

=====

A standard Model 4 has a software-controlled switch to select operation at either 4.05504 MHz (with heartbeat clock at 60 Hz) or 2.02752 MHz (with heartbeat clock at 30 Hz). sdltrs emulates this feature.

Model I's were often modified to operate at higher speeds than the standard 1.77408 MHz. With one common modification, writing a 1 to port 0xFE would double the speed to 3.54816 MHz, while writing a 0 would set the speed back to normal. The heartbeat clock runs at 40 Hz in either case. sdltrs emulates this feature as well.

Sound:

=====

Sound support uses the sound capabilities of libsnd which Open Sound System /dev/dsp device, standard on Linux and available on many other Unix versions as well.

The Orchestra-85 music synthesis software will run under sdltrs's Model I emulation, and the Orchestra-90 software will run with Model III operating systems under sdltrs's Model III, 4, or 4P emulation. For best results, use Orchestra-90 and the Model 4 emulation, as this lets the software run at the highest emulated clock rate (4 MHz) and thus generate the best sound. If you want to run Orchestra-85 instead, you can tell it that you have a 3.5 MHz clock speedup with enable sequence 3E01D3FE and disable sequence 3E00D3FE; this will let the software run twice as fast as on an unmodified Model I and generate better sound. There is no need to use sdltrs's -autodelay flag when running Orchestra-85/90, but you might want to specify a small fixed delay to keep from getting excessive key repeat.

Mouse:

=====

A few Model 4 programs could use a mouse, such as the shareware hi-res drawing program MDRAW-II. The program XTRSMOUS/CMD on the utility disk (utility.dsk) is a mouse driver for Model 4/4P mode that should work with most such programs. sdltrs does not emulate the actual mouse hardware (a serial mouse plugged into the Model 4 RS-232 port), so the original mouse drivers will not work under sdltrs. Instead, XTRSMOUS accesses the host mouse pointer using an emulator trap. XTRSMOUS implements the same TRSDOS/LS-DOS 6 SVC interface as the David Goben and Matthew Reed mouse drivers. (It does not implement the interface of the older Scott McBurney mouse driver, which may be required by some programs.)

By default XTRSMOUS installs itself in high memory. This is done because MDRAW-II tests for the presence of a mouse by looking to see whether the mouse SVC is vectored to high memory. If the driver is installed in low memory, MDRAW thinks it is not there at all. If you use mouse-aware programs that don't have this bug, or if you edit the first line of MDRAW to remove the test, you can install XTRSMOUS in low memory using the syntax "XTRSMOUS (LOW)".

Time of day clock:

=====

Several battery-backed time of day clocks were sold for the various TRS-80 models, including the TimeDate80, TChron1, TRSWatch, and T-Timer. They are essentially all the same hardware, but reside at a few different port ranges. sdltrs currently emulates them at port ranges 0x70-0x7C and 0xB0-0xBC. The T-Timer port range at 0xC0-0xCC conflicts with the Radio Shack hard drive controller and is not emulated.

These clocks return only a 2-digit year, and it is unknown what their driver software will do in the year 2000 and beyond. If you have software that works with one of them, please send email to report what happens when it is used with sdltrs.

Also see SETTIME/CMD in the "Data import and export" section above for another way to get the correct time into a Z-80 operating system running under sdltrs.

Finally, you might notice that LDOS/LS-DOS always magically knows the correct date when you boot it (but not the time). When you first power up the emulated TRS-80, sdltrs dumps the date into the places in memory where LDOS and LS-DOS normally save it across reboots, so it looks to the operating system as if you rebooted after setting the date.

Joystick:
=====

Pressing a key on a PC numeric keypad with NumLock disengaged emulates the Alpha Products joystick. See the Keys section above for details. The emulated joystick is mapped only at port 0, to avoid conflicts with other devices. Standard USB joysticks and gamepads will work as well, and may be selected from the Text GUI.

Save and Load Emulator State:
=====

sdltrs allows you to save the state of the emulator, so that it may be loaded later with the Load State command. Location of Disk and cassette image and inserted into the computer are stored in the state file. The state is stored in a .t8s file that may be saved using the Alt-S key combo. The Alt-L key combo will allow you to load a state file that has been saved.

Disk LED Indicators:
=====

sdltrs provides optional Disk LED indicators at the bottom of the emulated screen. You can turn these off with the -showled and -hideled options, or through the Text GUI. The following picture illustrates the LED's:

The LED's for the 8 floppy drives are on the left bottom of the screen, and in this case the first drive is active, and the other seven are not. The LED's for the 4 hard drives are on the right bottom of the screen, and in this case none of the drives are active.