SystemDisc
Technical Note

Introduction

DiscTools are a family of applications written by Piccolo Systems comprising:
- CloneDisc (EA-944-5) is a general purpose tool for copying the entire contents of a disc
- SystemDisc (EA-944-6) for managing the system software on SD card booting systems
- FormatDisc (not yet available) is a universal disc formatter

This technical note concerns the second in the family, SystemDisc.

Background

Modern RISC OS systems, such as the Beagleboard, Pandaboard or Raspberry Pi, feature a built-in SD or micro SD card reader. You need to have a memory card inserted into this slot when you power up the board, because their CPUs expect to read files (often known as 'firmware' or 'bootloaders') from which will in turn load RISC OS itself. These expect the SD card to be formatted with a FAT format - a format whose roots can be traced to MS-DOS or even earlier.

The SD card interface is, however, normally the fastest available file storage interface available on these systems, out-performing any USB storage devices by a significant margin, so it also makes sense to use it for general-purpose RISC OS file storage as well.

If you have an SD card in FAT format, you can put a standard RISC OS disc structure (!Boot, Apps, Diversions etc) on it and access it from RISC OS using either the built-in DOSFS or the third-party FAT32FS, but there are significant disadvantages:
- Incompatibility with some application software
- Less efficient space allocation
- Less robust against write errors (for example due to power loss or removal of card during writes)
- Inability to store the full range of RISC OS file attributes
- Corruption of RISC OS filetypes if the card is used in a non-RISC OS system

You end up with ugly and system-critical bootloader files cluttering up your root directory, which many people prefer to be hidden.

Consequently, the recommended solution for Beagleboard and Pandaboard users has remained to be to use a USB memory stick or hard drive, formatted to native RISC OS (FileCore) format, to hold the RISC OS disc structure. This does still mean that you have extra clutter attached to your board, and you're not getting the best speed from it.

For the Raspberry Pi educational computer Piccolo Systems helped to define a disc layout which sidesteps these issues with a clever scheme by which the card is formatted in a way which appears to be either FAT format or FileCore format, depending upon what is looking at it.

This did come with its own drawback, though, in that it mandated a one-size-fits-all disc capacity. A size of just under 2GB was chosen. This fits on nearly all 2GB SD cards, which are available very cheaply, and can also be used on any card of larger capacity (though only the first 2GB is then usable). 2GB is considerably larger than the software included with RISC OS as standard, yet understandably some users would like to be able to use the whole of larger SD cards, and/or to use the same trick on non-Pi hardware. This is where SystemDisc comes to the rescue!

SystemDisc is a flexible tool for creating and working with these dual-format media, which for want of a better name, I'm going to call system discs. With SystemDisc, you can:
- Take a card, formatted to a size of your choice using HForm or FormatDisc, and create a FAT-formatted boot partition within it of a size of your choosing
- Check the health of the boot partition on an existing card
- Fix a number of common problems that can occur with boot partitions

Using SystemDisc
The main window you see when you start SystemDisc looks like that shown in Figure 1.
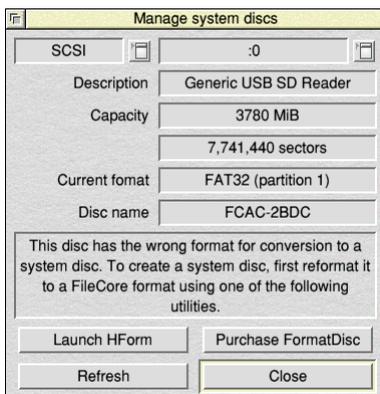
Figure 1 - SystemDisc's main window

When you buy a new SD card, plug it in (either the main SD slot or a USB card reader will do) and point SystemDisc at it, it will most likely look something like Figure 1.

You can use HForm to format the disc initially. HForm is is included as standard with every copy of RISC OS. HForm defaults to use the whole of the disc, unless you have previously formatted the disc with a FileCore format to use less that all of it, so you will probably want to accept the defaults. Make sure you answer "yes" when it asks if you want the disc to be bootable.

Once you have reformatted the disc, click **Refresh**, which should show something like Figure 2.
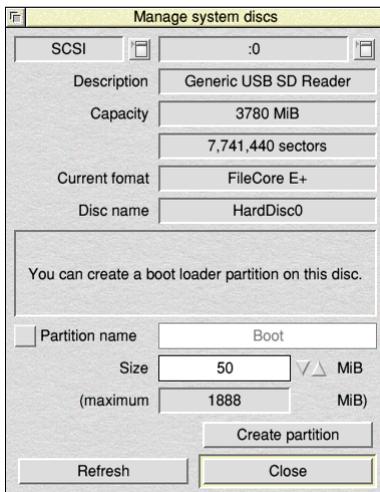
Figure 2 - Preparing to add the boot loader areas

Here you can customise the size of the boot partition. You can also give it a name, which will be how the card is identified when you insert it into non-RISC OS computers. Click **Create partition**.
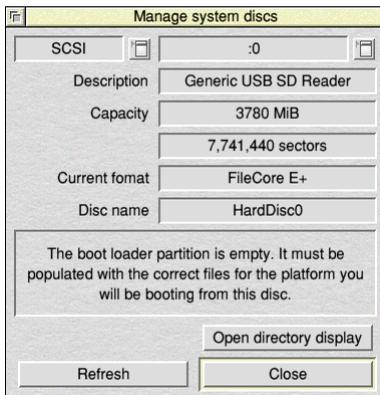
Figure 3 - Preparing to add the boot loader files

Now you need to install the correct ROM image and bootloaders for the board that you intend to use the card in. The exact set of required files varies from board to board, and sometimes will change over time as bootloaders are developed.

It is beyond the scope of this article to describe every possible combination, but the set of required files for some popular boards are as follows:

- Beagleboard: `BOOT/SCR, MLO, RISCOS, U-BOOT/BIN`
- Pandaboard: `MLO, PREENV/TXT, RISCOS, U-BOOT/BIN, UENV/TXT`
- Raspberry Pi: `BOOTCODE/BIN, CONFIG/TXT, FIXUP/DAT, RISCOS/IMG, START/ELF`
- Raspberry Pi 4: `CONFIG/TXT, FIXUP4/DAT, RISCOS/IMG, START4/ELF`

It's also possible to install these files using a different OS, in which case the '/' characters in the above filenames will appear as '.' characters instead. At this point, it's just as easy to use RISC OS, though, so click **Open directory display** and copy the files in. The filetypes don't matter, it is recommended to ensure that the filenames are all in block capitals as shown above before you copy the files in, as some of the early bootloaders can only recognise old style "8.3" filenames and not mixed case or lowercase ones.

Figure 4 - The loader files installed (Beagleboard shown)

Now click **Refresh**.

Figure 5 - Completed operation, awaiting !Boot application

This is the last step. Download one of the HardDisc4 archives from the miscellaneous downloads page of the RISC OS Open Limited website (click **Web site** to launch your web browser at the correct page - it's about half-way down) and extract it into the root directory of your new system disc. Don't worry about there being a !Boot directory already in the root directory in addition to the one in the archive - when you drag and drop the !Boot from the archive into the root directory, the Filer will take care of merging them together for you.

You should now have a fully functional system disc!

Frequently asked questions

The "Launch HForm" button is greyed out
This means that SystemDisc couldn't work out where your copy of HForm is. It knows to look in the place where it's installed by default, and from HForm 2.68 onwards, it will be able to find it wherever it is, so long as it has been "seen" by the filer. You either need to find and run HForm manually, or move your copy of HForm so you don't have this trouble again in future.

When I run HForm, it says "No ADFS module"
Chances are that you will be wanting to format an SD card either in the on-board SD card slot (using SDFS) or with a USB SD card reader (using SCSIFS). Versions of HForm prior to 2.60 didn't support these filing systems, and produce this error on any system that doesn't have an ATA controller (which is what ADFS is used for). Since you will need a recent "HardDisc4" archive to initialise your new system disc with a few steps down the line, you may wish to download this first and use the HForm from there instead.

I'm using an Iyonix, and when I run HForm, it only offers to format hard discs 4-7
As with the above, this is due to you having an old version of HForm installed.

I'm using RISC OS 5.18 or earlier with a USB card reader, and SystemDisc spends minutes hourglassing
Older versions of RISC OS could be very slow to mount (do an initial scan of) FileCore-formatted media, especially when using SCSIFS, and system discs need to use FileCore format. It is recommended that you upgrade to RISC OS 5.20 or later.