

APT Partition Editor and Utilities

for Atari 8-bit Computers

User's Manual

7th Edition

Contents

Introduction	3
Overview and Features	3
FDISK: Initialisation and Disk Partitioning.....	4
General Usage Notes	16
Card Preparation in macOS.....	18
Tools.....	19
MOUNT	19
LSDEV	19
LSDSK.....	19
ATRMNT	20
IDEDIAG.....	20
RESETDSK	20
Drivers.....	21
SDX HDD Drivers	21
RAPIDISK.SYS and RAPIDSKR.SYS	22
ULTD.SYS	22
Benchmarking and Stability Testing.....	23
Customising the System.....	23
Credits	24
Feedback	24
Finally	24

Introduction

The Atari Partition Table (APT) is a hard disk partitioning scheme (designed by KMK) intended to provide media compatibility among compliant Atari hard disk host adapters. Any host adapter whose firmware or soft-driver follows the APT specification will be compatible with the APT partition editor (FDISK) and APT utilities (such as those provided on the APT and IDE Plus 2.0 toolkit disks).

Compatible PBI adapters include IDE Plus 2.0, Ultimate 1MB/SIDE, XEL-CF, Incognito, and IDEa (when used with its APT BIOS). Compatible soft-drivers (which require the use of SpartaDOS X) are available for MyIDE, MyIDE/Flash, MyIDE II, SIDE, SIDE2, and Incognito (when in “Colleen” mode).

The APT partition editor – FDISK – is compatible with all APT device firmware and drivers.

Overview and Features

While the IDE Plus 2.0 adapter has its own toolkit (written by KMK) whose tools are generally compatible with all APT implementations, I will focus here not on the IDE Plus 2.0 toolkit disk, but on the APT tools distributed with the U1MB, Incognito, SIDE, MyIDE, and IDEa APT drivers. Of course the majority of these tools may also be used with IDE Plus 2.0 if desired, and indeed there is no alternative means of partitioning media in APT format on the IDE Plus.

A general synopsis of the features of current APT implementations is as follows:

- Support for a potentially unlimited number of dynamically mountable hard disk partitions (dynamic mounting not supported in the soft-drivers, nor currently in the IDEa or IDE Plus 2.0 APT implementations)
- Support for 15 drives (ATRs and/or partitions)
- Support for sector sizes of 128, 256 and 512 bytes
- Partition locking and deactivation
- Partition naming
- “External” (commonly FAT) partitions accessible by both the PC and the Atari
- ATR disk image mounting (Ultimate, Incognito and IDE Plus 2.0 only)

FDISK: Initialisation and Disk Partitioning

Before using a disk, it must first be initialised. I assume you already have the host adapter attached to the Atari, and whichever BIOS or driver installed and working. Power on the machine and boot whichever version of DOS you prefer to use (SpartaDOS X is not a requirement, although its use is optimal). If you are using a disk of Compact Flash card without any APT partitioning information on it, the driver may report “No partitions”, or simply respond “Device does not respond” when attempts are made to access drives other than D1:. This is normal, since we have not yet partitioned the disk.

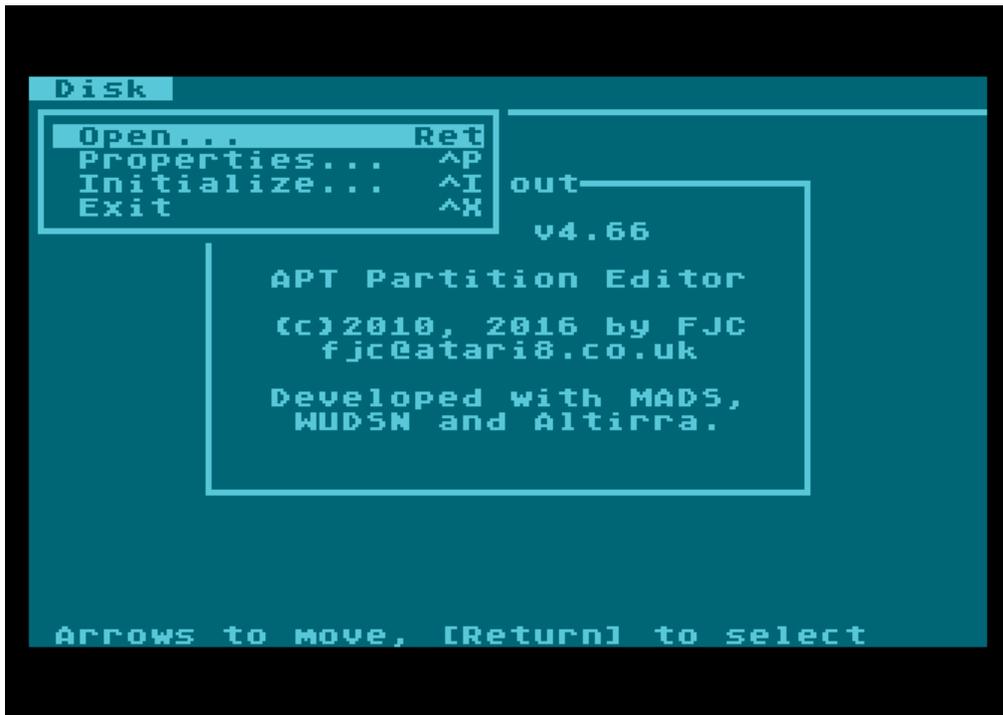
```

APT Partition View Help
-----
Num-Ptn  Start      Size-Drv-BPS-Attr-Partition Name
-----
1 Ext     63         1660865 --- 512 A--- [Unnamed]
2 Ext    1660928    1978368 --- 512 A--- [Unnamed]
3 D05    3646757    65535   D3: 512 A-BM BOOT
4 D05    3712293    65535   D4: 512 A--M 130XE ASSEMBLER
5 D05    3777828    65535   D5: 256 A--- [Unnamed]
6 D05    3843363    65535   D0: 128 A--- [Unnamed]
7 Ext    4707108    1542177 --- 512 A--- [Unnamed]
8 Ext    6249348    1574307 --- 512 A--- [Unnamed]
9 -----
10 -----
11 -----
12 -----
13 -----
14 -----
15 -----
16 -----
17 -----
-----
Partitions: 8                               798147 Free secs
Press [Esc] for MENU
    
```

If you already flashed the appropriate APT-ready SDX build to your hardware, and you are using SDX, you may simply type FDISK at the DOS prompt to run the partition editor, since the application is already on CAR: (user ROM space permitting). Otherwise, mount the APT toolkit ATR on drive 1 and run FDISK by typing its name (under SpartaDOS), or by using your DOS’s binary load command. You may also run FDISK from the SIDE loader if you prefer, if using SIDE/Ultimeate/Incognito.

FDISK has an intuitive GUI-style interface which is compatible with the SpartaDOS X S_VBXE.SYS driver (which provides the application with an 80 column display). There’s no need to disable the cartridge ROM (i.e. by using the “X” command under SpartaDOS X) unless you use RC_GR8.SYS to obtain an 80 column display, in which case BASIC or the SDX library must be disabled to make room for the display buffer.

Once FDISK loads, it will display a version notice which will disappear when you open the menus or press a shortcut key. Open the menu with “Esc” as directed, then press return, and you will see the “disk” menu:



To prepare a new disk or CF card, select “Initialise...” from the Disk menu. If you already prepared the disk, you can select “Open...” to open the APT for editing (if no APT is found, you will be directed to the disk initialisation dialog anyway). You will then be presented by the device selector:

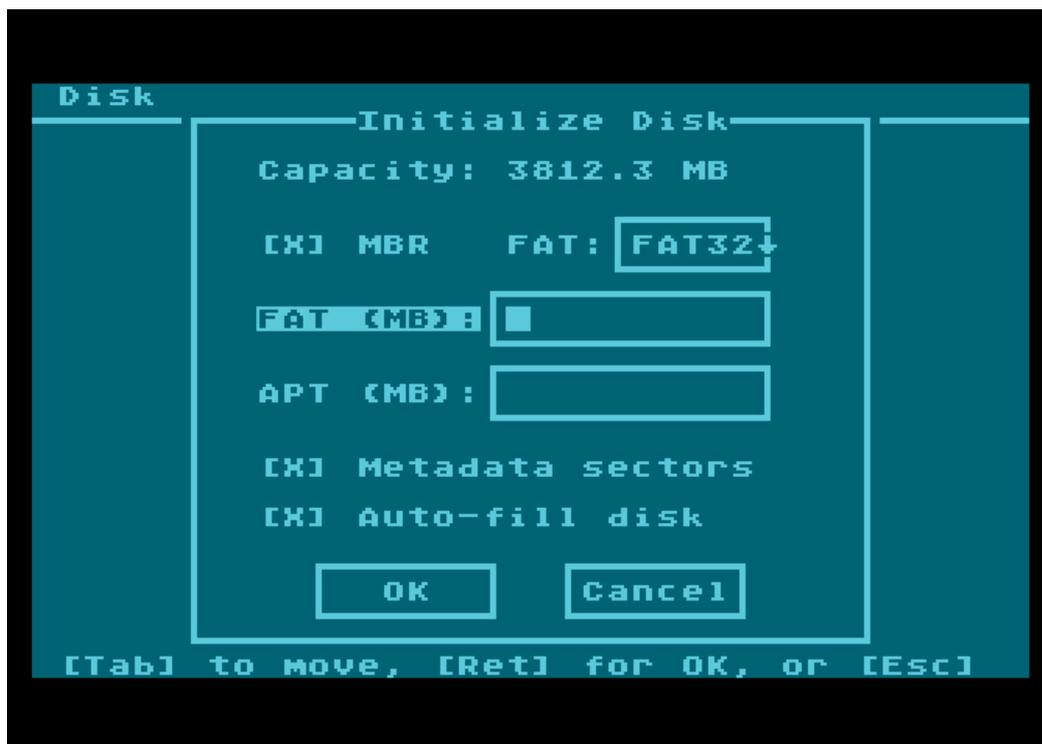


The device selector lists every single APT block device on the system, whether a PBI controller or LSIO-attached device driver. It’s therefore possible to run different physical devices via different adapters at the same time. Most commonly, though, you will see only one entry here: PBI, controller

0, master disk, which translates to 1.0.1. If you have two physical disks attached to the same adapter, you may also see 1.0.2.

The name of the APT host adapter or driver is also shown, so you can be in absolutely no doubt which device you are working on.

So – having selected the device you wish to initialise by pressing Return, you will may see some warnings if any content is discovered on the disk. If you want to initialise anyway, just “OK” through the alert boxes and you will eventually see the disk initialisation dialog:



Of course, the reported disk geometry will entirely depend on the chosen media. You have several options here, and some decisions to make. Now that FDISK handles “external” partitions, you may wish to have a FAT16 partition at the front of the disk for use with KMK’s FAT drivers for SpartaDOS X. Or you may wish devote half the disk to the SIDE loader (to contain XEX and ATR files), and half the disk for regular APT hard disk partitions. This is the time to decide how you want to arrange the disk.

For the sake of this walk-through, we’ll assume you want the following structure: a FAT16 partition to use with KMK’s FAT driver, and some APT space (which is mandatory) for regular hard disk partitions you can format and use with the DOS of your choice (in this case assumed to be SDX). “MBR” is already checked by default, and this makes sense since the MBR – even when FAT partitions are not required – serves as a protective mechanism to prevent a PC from obliterating the APT. So we’ll leave MBR enabled (to toggle it or any check box, incidentally, press space), and press shift-tab (assuming the “FAT (MB)” text box still has the focus) to move back up to the “FAT” pull-

down list. Once that is highlighted, press the down cursor key (with or without control), move the highlight to “FAT16” with the arrow keys, and then press enter to close the list.

Now press TAB to move back to “FAT (MB)”. Here’s where you tell FDISK how much disk space you want to use for the FAT16 partition. Notice (providing “auto-fill disk” is enabled) that the size of the APT automatically changes to fill the available free space when you TAB out of the APT (MB) field. You can turn auto-fill off if you want, perhaps if you wish to leave empty space on the disk for additional FATs created later on with some PC-based partitioning tool (since FDISK will only put a single FAT and an APT on the disk for you).

We’ll also leave “Metadata sectors” checked since this is where we can store names for our disk partitions. So, let’s move on by pressing Return. We’ll then be offered the chance to re-evaluate the chosen partitioning scheme and write it to disk if we are absolutely sure (which we need to be, since once the MBR is written to disk, it will be difficult to recover any information which was previously on the disk).

All being well, the next thing you’ll see is the APT editor:



This empty table is where you define the APT partitions which you’ll later format with DOS and copy programs and information to. The table has seven columns:

- Num: the partition sequence number
- Ptn: the partition type; commonly DOS, Ext, or F/W (firmware). The default is “DOS”
- Start: the start position of the partition
- Size: the size of the partition

- Drv: the partition's drive number, if assigned
- BPS: the partition density (128, 256 or 512 bytes)
- Attr: the partition attributes

With the highlight on the first entry, simply press Return and type the size in sectors of the first partition you want to create. 65,535 sectors is a common size and the maximum which MyDOS, SpartaDOS and SDX's SPARTA.SYS driver can currently handle. If you prefer to work in megabytes, select View->Megabytes from the drop-down menus and all numerical values will be shown in MB instead of sectors. Note that the "magic number" is 31.9995MB and no more if you want the partition to be compatible with currently available DOSes.

Once you have entered the partition size and pressed Return, cursor back up the entry and select Partition->Set Drive to give the partition a drive number. It's recommended you start numbering hard disk partitions from D2: or D3: upwards, but it really depends on how you want to configure the system. Leaving D1: free will allow you to boot from a floppy whenever you need to. In any case, you should be careful to leave at least one drive number unassigned, in case you need access to external SIO storage when the hard disk has every drive number assigned to partitions.

Next, select Partition->Set BPS and change the density of the partition if 512 bytes isn't the size you want. You can choose between 128, 256 and 512 bytes per sector. 256 is the most common size for pre-SDX versions of DOS. Note you can use the shortcut keys listed in the menus to access these options direct from the editor without actually opening the menus at all.

OK – so what if you want eight identical partitions and don't want to type in the details for each one? Well, the latest version of FDISK has a facility which makes this very easy indeed: "Fill/Divide". Select Partition->Fill/Divide and type in the details of how you want the program to divide and fill the space in the APT.



If you want eight partitions all of equal size, simply enter “8” in the “Partitions” field, set “First drive” to the number you want assigned to the first partition (the rest will follow in series; I’d recommend D3: here), then press Return. FDISK will create eight 512bps partitions of equal size, with drive numbers increasing by one, as near as possible filling all the available space. But if the APT space is large, the partitions may end up more than 65,535 sectors long, which at the moment we don’t want.

The solution, then, is to simply specify the size in sectors as well. This will limit the size of the created partitions. Likewise, you can omit “Partitions” entirely and just type 65535 into the “Size” field. In that case, FDISK will create as many 65,535 sector partitions as possible to fill the entire APT space. Note that you can also change the sector size here, if for example you want a number of identical 256bps partitions.

Since partitions are commonly created in identical sizes, “Fill/Divide” is probably the most convenient way to build a partition table. In any case, when you’ve finished creating the partition table, you should set the boot partition. This will tell the BIOS or drive which partition to boot from when the computer is switched on. Just cursor up to the partition you wish to assign as the boot partition, and press “B”, or select Partition-Bootable. You’ll see the “B” flag appear next to the partition, under “Attr” (Attributes).

Note the other flags too: A means “active”, and simply means that the partition will be auto-mounted when the partition table is read when the machine is switched on. Clearing the active flag won’t delete the partition, but will make it inaccessible. Removing the drive number effectively has

the same effect as clearing the active flag. Next is space for the “L” flag, which means “locked”. A locked partition is read-only, and cannot be formatted or written to.

Next we have the “B” (boot) flag as already discussed, and finally the “M” flag, which stands for “Metadata”. Metadata is contained in a special reserved sector, typically right in front of the partition on disk (watch the start sector numbers when editing the partition table when metadata is enabled, and you may notice space being reserved for metadata, as well as for the APT partition table chain itself if many partitions are created). You can turn metadata off so that new partitions do not have metadata by default: do this via APT->Properties (since the setting is too destructive to access direct from the editor). But note that without metadata, partitions cannot be named. In addition, enabling or disabling metadata later on a per-partition basis (which can be done via Partition->Properties) will *move* the partitions and make any data in them invalid. Partitions will also move when partitions ahead of them are added or deleted, so before formatting anything and copying information into the partitions, it’s wise to get decisions like metadata and partitions size out of the way at the first instance.

My advice is to leave Metadata globally enabled (the default state).

You should now have a partition table which resembles this:

```

APT Partition View Help
-----
Num Ptn Start Size Drv BPS Attr
-----
 1 DOS 3646757 65535 D3: 512 A--K
 2 DOS 3712293 65535 D4: 512 A--M
 3 DOS 3777829 65535 D5: 512 A--M
 4 DOS 3843365 65535 D6: 512 A--M
 5 DOS 3908901 65535 D7: 512 A--M
 6 DOS 3974437 65535 D8: 512 A--M
 7 DOS 4039973 65535 D9: 512 A--M
 8 DOS 4105509 65535 DJ: 512 A--M
 9 ---
10 ---
11 ---
12 ---
13 ---
14 ---
15 ---
-----
-Partition name-----
[Unnamed]
-----
Partitions: 8          536001 Free secs
-----
Press [Esc] for menu
    
```

In the example above, we have eight equally sized 512bps partitions, numbered D3: through DJ: (3-10), and D3: set as the boot drive. But what of the FAT partition we created earlier? We want that to be readable in SpartaDOS X...

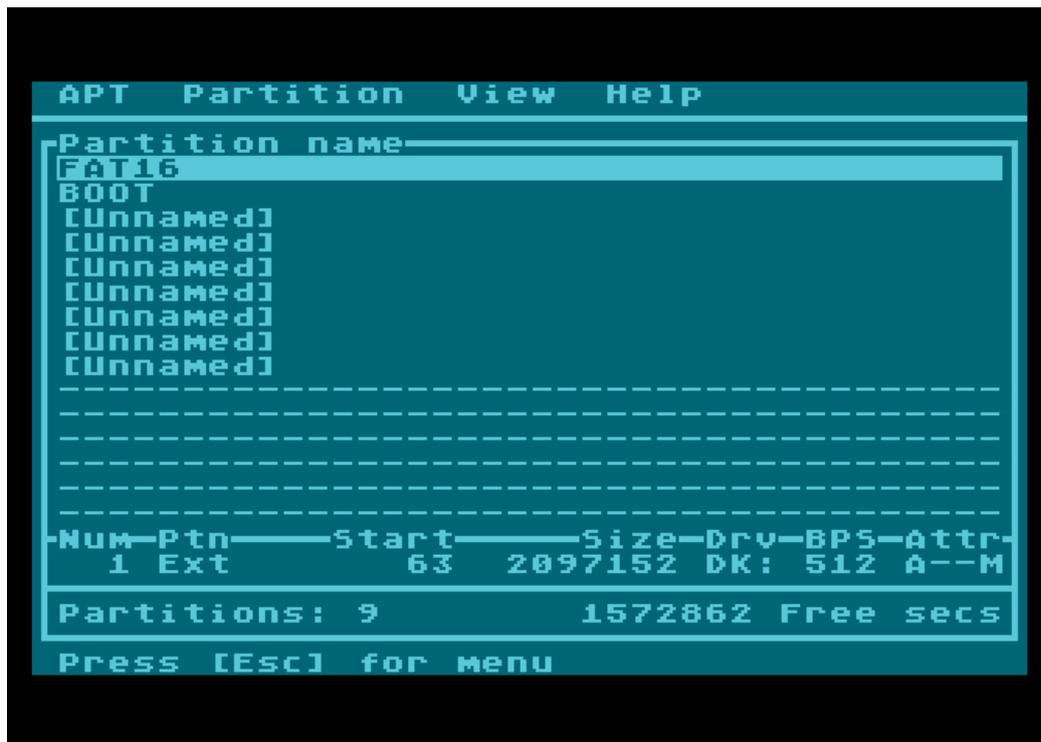
To get access to the FAT16 partition from SDX, we need to add one last partition: an external partition. Choose Partition-> External, and you will see a list of all the FAT partitions (which is to say, those which belong to the MBR partition table, and are *outside* of the APT area):



We can see the FAT16 partition we created earlier (although it's currently unformatted – we'll do that on the PC later), so hit return and an entry pointing to that FAT will be added to the APT table. The external partition is denoted by a left-pointing wedge at the left side of the entry. All that remains is to assign the external partition a drive number.



Note that, as previously mentioned, we can name partitions. Names are stored in the metadata sectors. To edit a partition's name, just press "N", or choose Partition->Edit Name. Type the name in the space at the foot of the screen and press enter. The names of partitions are displayed here as you scroll through the list. When using an 80 column display, the partition names are displayed to the right of the partition details, and are editable *in situ*. Without an 80 column display, you can still get an overview of named partitions by toggling View->Names:



This displays the partition table in name view, with the details displayed where the names are normally shown. You can still edit the everything in this view; it's just that names are shown in the list and details shown where the names would normally be. Go back to normal view by choosing View->Names again.

When you're finally happy with the partition table, write it to disk with APT-Write, or by pressing Ctrl+W.



You get two chances to confirm this operation. If everything went according to plan, the partition table has now been written to disk, and made active. The next thing you need to do is format the partitions. So, select APT->Exit to get back to DOS. Once there, use your DOS's formatter (in SDX, type "FORMAT") to write directory structures to all the partitions you just created except the FAT partition. There's no point in formatting the FAT partition, since we want a FAT file system there, and we'll do that by using the PC's card reader.

Once you have formatted all the other partitions, you should be able to list the empty root directories in DOS. If everything looks right, remove the disk or CF card from the adapter (if necessary, power down first), and connect to the PC. The PC's operating system should recognize the MBR partition table, and say that the FAT partition requires formatting. Go ahead and format the partition, being sure (since we chose FAT16 when setting it up) that "FAT" is used (not "FAT32" in this particular example). Once that's done, copy some files into the newly formatted FAT16 partition on the PC, making sure they have 8.3 names (no long filenames for the moment). Now, remove the disk or card from the PC, reconnect to the Atari, and power-on again.

At the SpartaDOS X prompt, type DIR drive:, where drive is the letter of the external FAT partition you created earlier. If all went well, SDX should report "Unrecognized disk format". That's as we expect, since we need to install FATFS.SYS first, which may be found in the 'drivers' folder of the APT Toolkit ATR. If you intend to use FAT partitions all the time, you should copy this file to your boot partition and add the line "DEVICE FATFS" to CONFIG.SYS.

With the driver installed, you should now have read-only access to the external FAT partition. Well done! The implications of this should be obvious: the FAT partition offers a very convenient way to

move files and ATRs in bulk from the PC to the Atari, without the need for specialised tools or file system drivers on the PC side. Moreover, if you copy ATR disk images into the FAT partition (providing you are using Ultimate or Incognito), you can mount these disk images directly, with read/write access using the ATRMNT tool which you can find on the APT Toolkit ATR.

With the FAT partition mounted, just type `ATRMNT A: ATRNAME` to mount an ATR on the specified (unused) drive letter. You can then read from and write to the ATR, just as if it were a regular disk, and when done, copy the changed ATR back to the PC by simply dragging it or copying it from the FAT partition.

Note that IDE Plus 2.0 requires mounted ATRs to reside in an SDFS partition, so if using that adapter, you'll first need to copy the ATRs into an SDFS formatted partition, before mounting them with IDE Plus's proprietary mounting tools (`ATRMNT` and `MOUNT` are intended for use only with Ultimate and Incognito PBI BIOSes).

General Usage Notes

Of course, the foregoing guide just covers one typical scenario of disk partitioning. You may instead want to create a FAT32 partition alongside the APT for use with Candle's SIDE loader. You can even create multiple FAT partitions of different types by leaving sufficient space for them when initialising the media with `FDISK`, and adding more FAT partitions later on using a tool on the PC. In that case, it's possible to then add external entries for these FAT partitions just as we did for that single FAT16 partition. Once FAT32 file system drivers are available for SpartaDOS X, it will make sense to have multiple external FAT16 and FAT32 partitions.

As for `FDISK`, while there are facilities that we have not fully covered here, we have certainly covered basic operation, which is sufficient to get up and running with a usable hard disk. It is possible to resize and change the properties of already formatted and populated partitions, but be aware that since `FDISK` now insists on there being no unused disk space *between* partitions, removing metadata, inserting external partitions with metadata, deleting partitions, and inserting new ones is likely to invalidate the data on neighbouring partitions. So you need to bear this in mind when first planning your partitioning scheme.

The way that `FDISK` works now is completely different from the way it worked in the past, and for this reason – unless you have already been using recent beta versions of the application – you might find it best to simply partition your media from scratch. Things such as named external partitions have required changes to the APT specification (a new “external” partition type `0x03` was defined, and a global metadata flag implemented), so when using partition tables created with very old versions of the program, it's quite likely that you will run into problems with large or complex APT structures. The current version of `FDISK` (and the corresponding changes to the APT spec) are

intended to be forward-thinking, catering for forthcoming improvements to SpartaDOS X's file system capabilities with the minimum of fuss. Large FAT and SDFS partitions can already be created with FDISK, so once they have file system support, they should simply work.

Card Preparation in macOS

While the Windows OS allows for the formatting of the FAT partition created with the APT FDISK tool and the creation of extra FAT partitions (via the Disk Management Console), things are less easy in macOS owing to the fact Disk Utility does not recognize media with an unformatted FAT partition. As a result, the OS fails to recognise the FAT partition on a card initialised with FDISK, and will not allow it to be formatted using Disk Utility.

To fix this, we must resort to the macOS terminal and *diskutil*. After preparing a card using FDISK on the Atari (assuming the card has an MBR and one FAT partition and an APT container), connect it to the Mac using a card reader, open the terminal, and perform the following steps:

- Type: *diskutil list*. The card's MBR partition table should be displayed, showing the FAT and the (unrecognised) APT entry (type \$7F)
- Take a note of the volume ID of the FAT (e.g. disk2s1)
- Type (e.g.): *diskutil EraseVolume "ms-dos FAT16" FATVOL disk2s1* (be careful to replace *disk2s1* with the volume ID of the FAT on the card, and replace *FATVOL* with the desired volume name)
- The FAT volume will be formatted and will become accessible on the Mac

Be sure to use the logical volume ID of the FAT volume and not the low-level device ID (otherwise the MBR will be destroyed). You may now create additional FAT volumes using Disk Utility if you reserved free space on the card when preparing it with FDISK.

Tools

As previously mentioned, the APT toolkit disk includes a number of support utilities, which I will briefly explain below. You may also refer to the documentation files on the toolkit ATR.

MOUNT

MOUNT.COM is a tool for dynamically mounting and unmounting partitions. It is not intended for mounting ATR disk images, although it will unmount ATRs. It should be used only with the Ultimate 1MB and Incognito APT PBI BIOS. Requires SpartaDOS X.

Usage

```
MOUNT [dev] /N drive: part_num
MOUNT [dev] drive: part_name
MOUNT [dev] /U drive:
MOUNT [dev] /L
```

Dev: [[BusID.]CntrlrID.]DeviceID]

```
    BusID: 1=PBI, 2=LSIO
CntrlrID: 0-7
DeviceID: 1=Primary, 2=Secondary
```

Switches:

```
L : List partitions
N : Mount by partition number
U : Unmount specified drive
```

LSDEV

LSDEV.COM simply lists all APT devices on every APT controller found on the system. Unlike (X)DEVINFO, LSDEV will also report LSIO and PBI controllers when both are present. Compatible with all APT implementations, and all DOSes.

Usage

```
LSDEV
```

LSDSK

LSDSK.COM lists every online APT partition on every device on the system. Works with any DOS.

Usage

```
LSDSK
```

ATRMNT

ATRMNT.COM runs with SpartaDOS X only and is an ATR mounting tool for FAT partitions. It should be used in conjunction with the FATFS.SYS driver.

Usage

```
ATRMNT [/U] D: [ATRNAME[.ATR]]
```

```
/U: Unmount drive
```

IDEDIAG

IDEDIAG.COM is a tool which runs some basic stability tests on U1MB and Incognito PBI-hosted hard disk controllers (including the XEL-CF on the 1088XEL and 1088XLD machines). The tool takes no command line arguments. It may prove useful when trying to diagnose communication problems between the computer and the hard disk.

RESETDSK

RESETDSK.COM simply issues an 'all reset' command on any APT hard disk host adapter. This causes all physical disks to undergo a hardware reset, and have their PIO configuration reset to initialisation defaults. This is the exact same process which normally happens when the user presses system reset.

Drivers

The following drivers are provided on the toolkit disk (and many are provided pre-installed on the CAR: volume of the custom SDX ROMs):

- SIDE.SYS (SIDE/SIDE2 HDD driver for SDX)
- COLLEEN.SYS (Incognito 'Colleen' mode HDD driver for SDX)
- MYIDE.SYS (MYIDE HDD driver for SDX)
- MYIDEI.SYS (Internal MYIDE HDD driver for SDX)
- MYIDE2.SYS (MYIDE2 HDD driver for SDX)
- XEL_CF.SYS (XEL-CF HDD driver for SDX)
- RAPIDISK.SYS (HDD IO accelerator for Rapidus)
- RAPIDSKR.SYS (reboot-proof HDD IO accelerator for Rapidus)
- ULTD.SYS (U1MB/Incognito RTC driver for SpartaDOS 3.x, Bewe-DOS, RealDOS, etc)

SDX HDD Drivers

The 'HDD Drivers' for SDX are commonly pre-installed on the CAR: volume and automatically loaded by CONFIG.SYS. These drivers are required when there is no PBI controller hardware; for example, on a SIDE cartridge with no U1MB PBI support, or when Incognito is running in 'Colleen' (800) mode with no OS support for the PBI BIOS. The drivers patch into the SDX LSIO table (much like RAMdisk drivers) and therefore provide DOS level (rather than OS level) access to the hard disk.

Drivers are installed with DEVICE *driver_name* in CONFIG.SYS (e.g. DEVICE SIDE). Some option switches are also supported:

/N	Emit SIO-style audio beeps during HDD transfers
/Q	Install quietly (i.e. do not display the controller name, list of drive numbers, etc)
/S	Enable secondary physical disk (XEL_CF.SYS only)

RAPIDISK.SYS and RAPIDSKR.SYS

RAPIDISK.SYS and RAPIDSKR.SYS are two drivers which patch into the Ultimate 1MB/Incognito PBI BIOS and offer greatly increased HDD transfer rates on machines equipped with the Rapidus 65C816 accelerator board. The driver requires an OS which runs in 65C816 'native' mode (i.e. 'DRACOS').

Both drivers install in high linear memory, but while RAPIDISK.SYS is un-loaded on an OS restart, RAPIDSKR.SYS will remain present even after an OS restart. This allows the driver to be used when running – for example – the SIDE Loader; mounted ATRs may therefore also be accelerated. The drivers can even be installed directly from the loader if required, although more commonly they would be referenced from CONFIG.SYS under SpartaDOS X:

```
DEVICE RAPIDISK
```

If the computer is switched out of 'native' 65C816 mode, or the current OS is no longer capable of running native 65C816 code, the driver will become unavailable and the standard ROM-based PBI IO code will be used.

ULTD.SYS

ULTD.SYS is an U1MB/Incognito Real-time clock driver intended for use with SpartaDOS 3.x, RealDOS and BeweDOS (BW-DOS). It replaces the built-in RTime-8 clock driver and provides time-keeping services based on the SD1305 RTC chip built into the U1MB and Incognito. It is also fully compatible with the TDLINe on-screen time/date display.

Benchmarking and Stability Testing

Excellent utilities exist for the purpose of benchmarking disk transfer speeds and verifying the accuracy of data transfer: RWTEST and RWCRC respectively. These tools are written by Drac030 and are available from his website:

<http://drac030.atari8.info>

RWTEST will establish the transfer speed of the hard disk interface, and RWCRC will perform a cyclic redundancy check on a large file in order to ascertain that I/O on the interface is free from errors. A “pass” using RWCRC usually means a stable hard disk interface, and is a highly recommended test when first setting up your IDE host adapter.

You should obtain the latest versions of RWCRC and RWTEST from the “Other Stuff” page of KMK’s website.

Customising the System

If you wish to edit the default CONFIG.SYS on the CAR: device (necessary when altering the configuration when soft-drivers are used), you may use the SDX Imaging utility available from the SpartaDOS X Upgrade site (<http://sdx.atari8.info/>) for this purpose. Note, you should always include DEVICE [driver name] in CONFIG.SYS, otherwise you will lose access to the hard disk.

The above does not apply to PBI host adapter solutions, such as Ultimate, Incognito, IDE Plus, and IDEa.

Credits

Thanks to the following:

- Candle for SIDE, SIDE2, Ultimate 1MB and Incognito
- Mytek for the 1088XEL, 1088XLD and XEL-CF designs
- Lotharek for providing hardware
- KMK and Trub for SpartaDOS X
- Phaeron for the fantastic Altirra emulator and the Altirra Hardware Reference Manual
- Lewis for beta testing the old SIDE drivers
- ProWizard for his tremendous and dedicated assistance in troubleshooting the PBI BIOS and for bringing to my attention bugs everyone else – including me – missed
- Hias for his help on the old MyIDE partition table design
- Everyone else who helped during these past three years... there are many of you.

Feedback

If you discover any bugs or experience any issues using the software, or perhaps you simply have some suggestions about improvements which could be made, please contact me via the AtariAge forums, or by email at:

fic@atari8.co.uk

Finally

If you find the tools and drivers useful, and would like to offer appreciation for the many hundreds of hours in total which went into developing them, please feel free to make a donation at www.atari8.co.uk/donate/. I enjoy no licensing agreements with OEMs, and therefore, while providing software without which the corresponding hardware would be useless, my only rewards are the pleasure I gain from programming my Atari, feedback from users, and any gestures of acknowledgement received by way of donations.

Thank you in advance. I hope you find the APT tools useful.

Jonathan Halliday (FJC)

April 2020