Ultimate 1MB, Incognito, 1088XEL|U1MB, 1088XLD|U1MB SIDE, SIDE2 and SIDE3 Firmware

for Atari 8-bit Computers

User's Manual

Revised to cover firmware version 4.2.

Contents

Introduction

This document covers use of the 'alt' firmware (including BIOS setup utility and XEX loader) for Ultimate 1MB and Incognito upgrades. A version of the XEX loader is also provided for the SIDE/SIDE2/SIDE3 cartridges. The firmware is supplied either as three separate files for the Incognito (Main BIOS, PBI BIOS, and XEX loader) and a single 64KB firmware 'blob' for U1MB.

The alternative BIOS setup tool was created from scratch, initially to address certain limitations of the original firmware, to facilitate further development of the PBI BIOS, and allow more flexible use of the flash ROM space. A new XEX loader was subsequently developed and the PBI BIOS significantly reworked, resulting in an almost complete ground-up rewrite of all firmware components.

Enhancements provided by the new firmware include:

- The ability to set the date and time from the setup menu
- Four configuration profiles
- Modular menus covering system settings, hard disk, device setup, etc
- Super-fast XEX loader with speed-search facility and multiple FAT partition support
- ATR disk images, XEX files, APT partitions and BASIC programs all handled by the loader
- Scripting facility to automatically handle mounting of multi-disk titles
- Turbo-Freezer compatibility
- Customisable BIOS entry, reboot and partition table rescan shortcut keys
- OS-independent high-speed SIO for serial disk drives (US Doubler, Happy, XF551, etc)
- OS-independent SIO2BT operation (with optional high-speed support)
- 'Z:' RTC driver
- Optional BIOS logo screen and boot device selection menu
- Optional flash write protection (Ultimate 1MB only)
- System information page displaying CPU type, audio/video hardware, etc
- BIOS Plugin API for diverse hardware control (Ultimate 1MB only)
- Plugin modules supporting Stereo Pokey, Covox, Soundboard, Rapidus, and the 1088XEL
- Variable SpartaDOS X ROM sizes (192, 256 and 320KB), and the ability to boot the WIP Graphical OS (when 192KB SDX is used)

New to version 4.2

Version 4.2 of the firmware is generally a bug-fix release with a slightly redesigned UI.

Features:

- Redesigned setup menu and loader logo/icon bar
- Redesigned screensaver
- Fixed JED version reporting
- Improved DOS 2.5 compatibility in loader FMS

New to version 4.0

Version 4.0 of the firmware brings several enhancements over prior releases. Among the most notable:

BIOS

- Revised UI
- The ability to use the external SIDE/SIDE2 cartridge ROM alongside mounted ATRs and HDD partitions (a sample set of OSS productivity ROMs is provided)
- Compatibility with the SIDE3 cartridge (you can use the SIDE3's SD card as a bootable APT hard disk, just as you can with CF cards on the SIDE/SIDE2 cartridges)
- Optional suppression of IO noise when using the HSIO driver, and (optional) simulated IO noise when using mounted ATRs and HDD partitions
- The ability to display the BIOS splash screen only at cold power-on
- All hardware-specific hard disk options moved to plugins, allowing use of different PBI-controlled hardware without changes to the main BIOS
- Improved SD card and SD/CF adapter compatibility
- More robust SIO error checking and support for immediate mode IO

Loader

- Revised UI with sixteen-line filename display
- Automatic logging of the last accessed directory on the last accessed FAT partition
- Get current working directory, change working directory and path support in the CIO FAT filesystem driver (loaded XEX files may now access the entire directory tree via SDX-compatible CIO commands)
- Long filename and path support in MAP files, enabling MAP files to reside in different folders from the ATRs to which they refer
- Search facility now capable of scanning a virtually unlimited number of nested folders

- Search speed improved by fifty percent
- Directories containing more the 250 items may now be viewed in full by paging through them 250 entries at a time
- Reverse lookups are now possible on drive slots, enabling fast jumping to ATR files by drive number
- New shortcut keys for parent and home folders
- Keyboard and joystick auto-repeat acceleration for faster traversal of lengthy directories
- Improved CF/SD card and SD/CF adapter compatibility
- The ability to reboot with SDX enabled
- Selection and automatic booting of other cartridge ROMs adapted for use with SIDE/SIDE2
- Improved error detection, handling and reporting (error codes are produced)
- Several bug fixes user interface improvements

Acknowledgements

Writing replacement firmware for Ultimate 1MB from the ground up turned out to be an even bigger task than I expected it to be, and when the scope of the project broadened to encompass the 1088XEL Mini-ATX motherboard project in late 2017, the undertaking at times threatened to become completely overwhelming. Then, in 2020, I was tasked with writing the loader and PBI BIOS for the SIDE3 cartridge. Fortunately, I received plenty of expert help, advice and encouragement from the most knowledgeable figures in the Atari community. Sincerest thanks to:

- Hias for testing (with Turbo-Freezer and a scope), suggestions, technical assistance, and especially for his fantastic high-speed SIO code which he kindly allowed me to incorporate into the PBI BIOS

- Phaeron for his encyclopaedic knowledge, technical insights, the fabulous Altirra Hardware Manual, and most of all for the incredible Altirra Atari 8-bit emulator

- Marius Diepenhorst for his enthusiastic and meticulous testing, inventive ideas, encouragement and friendship

- MrFish for suggestions and feedback on the interface design

- Lotharek for kind provision of Ultimate 1MB, VBXE and Rapidus hardware

- Kuba and Pin for late-night stress testing of XEX loader

- TheMontezuma for SIO2BT hardware and SIO patch testing

- Drac030 for APT and for his guidance regarding CPU frequency measurement

- Candle for designing all the hardware and for offering me the opportunity to write the old PBI BIOS for U1MB/Incognito

- Michael St. Pierre (mytek) for his support, donations, and for placing the alt-BIOS powered U1MB at the heart of his 1088XEL and 1088XLD motherboard projects

- Eric Bacher for patching the OSS programming language ROMs for use with SIDE/SIDE2

- Marcel Moelands for his enthusiastic testing of the Sophia 2 plugin, for his insightful suggestions for improvements, and for writing the description of the Sophia 2 plugin contained in this manual

Special thanks also to all the members of the Atari 8-bit community who take the time to report bugs, lodge feature requests, and make donations in recognition of the enormous amount of time invested into the development and maintenance of this firmware.

Installation

The procedures for updating from the original (Candle) firmware and for applying updates to equipment already running the new firmware are broadly the same, since the flashing software (UFLASH) is capable of handling the transition from old to new.

Using SIO2SD or SIO2PC and a Peripheral Emulator

Ultimate 1MB

1. Remove any cartridges (including SIDE/SIDE2) or PBI peripherals.
2. Mount the supplied update ATR (FIRMWARE.ATR) on drive 1.
3. Enable SpartaDOS X in the U1MB BIOS.
4. Disable 'SIDE Hardware' (original BIOS) or 'PBI BIOS' (new firmware).
5. Ensure system memory is set at '1088K RAMBO'.
6. Save the setting and press 'C' to reboot the machine.
7. You should see the SpartaDOS X prompt 'D1:'. Type UFLASH.XEX and press Return.
8. If your hardware is not automatically detected, do not proceed.
9. Assuming the hardware is detected, navigate to the 'Firmware' entry and press Return.
10. From the resulting file list, select 'FIRMWARE.ROM' and press Return.
11. Press Return again to confirm the flash.
12. Power-cycle the machine when the operation is complete.
13. If updating from the original Candle firmware, you will immediately be sent to the configuration menu (with the message 'Profile 1 reset'), where you should set the system clock and make any other adjustments as you wish, before saving the new configuration and rebooting the machine by pressing 'B'.

Incognito

Since the Incognito firmware is spread around the flash ROM, it's currently not possible to flash all components at once, so one must flash them individually.

1. Remove any cartridges.
2. Mount the supplied update ATR (FIRMWARE.ATR) on drive 1.
3. Enable XL/XE mode, 1088K and SpartaDOS X in the Incognito BIOS.
4. Disable 'SIDE Hardware' (original BIOS) or 'PBI BIOS' (new firmware).
5. Ensure extended memory is set at '1088K RAMBO'.
6. Save the setting and press 'C' to reboot the machine.
7. You should see the SpartaDOS X prompt 'D1:'. Type UFLASH.XEX and press Return.
8. If your hardware is not automatically detected, do not proceed.
9. Assuming the hardware is detected, navigate to the 'Main BIOS' entry and press Return.
10. From the resulting file list, select 'IBIOS.ROM' and press Return.

11. Press Return again to confirm the flash.

12. Repeat the above process for the 'Loader' entry (selecting 'ILOADER.ROM') and the 'PBI BIOS' entry (choosing 'IPBI.ROM'), in that order.

13. Power-cycle the machine when the operation is complete.

14. If updating from the original Candle firmware, you will immediately be sent to the configuration menu (with the message 'Profile 1 reset'), where you should set the system clock and make any other adjustments as you wish, before saving the new configuration and rebooting the machine by pressing 'B'.

Updating with the SIDE XEX Loader

You can update the firmware using the SIDE loader by placing UFLASH and the ROM files in a FAT partition on your Compact Flash card.

Ultimate 1MB

1. Ensure that XL/XE mode is selected, SDX is enabled and RAM size is set to '1088K RAMBO'.

2. Place UFLASH.XEX and FIRMWARE.ROM in the FAT partition of your CF card, plug the card into your SIDE cartridge and attach SIDE to the Atari.

3. Turn on the Atari while holding down the 'L' key to run the SIDE XEX Loader.

4. If already using the new SIDE loader, make sure that 'FAT FMS' is set to 'Enabled' in the Options menu.

5. In the FAT directory listing, navigate to 'UFLASH.XEX' and press Return to run it.

6. If your hardware is not automatically detected by UFLASH, do not proceed.

7. Assuming the hardware is detected, navigate to the 'Firmware' entry and press Return.

8. From the resulting file list, select 'FIRMWARE.ROM' and press Return.

9. Press Return again to confirm the flash.

10. Power-cycle the machine when the operation is complete.

11. If updating from the original Candle firmware, you will immediately be sent to the configuration menu (with the message 'Profile 1 reset'), where you should set the system clock and make any other adjustments as you wish, before saving the new configuration and rebooting the machine by pressing 'B'.

Incognito

1. Ensure that XL/XE mode is selected, SDX is enabled and RAM size is set to '1088K RAMBO'.

2. Place UFLASH.XEX and FIRMWARE.ROM in the FAT partition of your CF card, and plug the card into your Incognito's card slot.

3. Turn on the Atari while holding down the 'L' key to run the SIDE XEX Loader.

4. If already using the new SIDE loader, make sure that 'FAT FMS' is set to 'Enabled' in the Options menu.

5. In the FAT directory listing, navigate to 'UFLASH.XEX' and press Return to run it.

6. If your hardware is not automatically detected by UFLASH, do not proceed.

7. Assuming the hardware is detected, navigate to the 'Main BIOS' entry and press Return.

8. From the resulting file list, select 'IBIOS.ROM' and press Return.

9. Press Return again to confirm the flash.
10. Repeat the above process for the 'Loader' entry (selecting 'ILOADER.ROM') and the 'PBI BIOS' entry (choosing 'IPBI.ROM'), in that order.
11. Power-cycle the machine when the operation is complete.
12. If updating from the original Candle firmware, you will immediately be sent to the configuration menu (with the message 'Profile 1 reset'), where you should set the system clock and make any other adjustments as you wish, before saving the new configuration and rebooting the machine by pressing 'B'.

Re-flashing the Entire ROM

Using either the SIO2xx method or the SIDE XEX loader method, you may flash the entire 512KB ROM with one of the provided 512K binaries. You should choose the appropriate file from the list below:

ROM	Platform	Graphical OS	SDX Size
ULTIMATE.ROM	U1MB	Yes	192K
ULTNOGOS.ROM	U1MB	No	320K
INCOG.ROM	Incognito	Yes	192K
INNOGOS.ROM	Incognito	No	320K
XEL.ROM	1088XEL	Yes	192K
XELNOGOS.ROM	1088XEL	No	320K
XLD.ROM	1088XLD	Yes	192K
XLDNOGOS.ROM	1088XLD	No	320K

In UFLASH, you should select the topmost entry in the list of 'slots' (that entry being the name of the flash ROM chip in the U1MB/Incognito) and flash the 512K ROM file to it.

The 'complete' ROMs are also found on the FIRMWARE.ATR disk images. These 512K ROMs include SpartaDOS X, a selection of Operating Systems, BASIC interpreters and programming tools, as well as the complete firmware package. ROMs which include 'NOGOS' in the name include a 320KB version of SpartaDOS X and omit the WIP GOS. The other ROMs ('ULTIMATE.ROM' for U1MB, 'INCOG.ROM' for Incognito, etc) include the WIP GOS and a 192KB version of SpartaDOS X.

Unlike other upgrade methods, this approach will completely replace the contents of your ROM including any customisations you have already made, and is highly recommended for users upgrading from the original 'stock' firmware, since it removes the patched high-speed OS (which is no longer required), ensures SDX is up to date, and upgrades the firmware in a single operation.

Other Upgrade Methods

The supplied ATR containing the flashing tool and ROMs may also be mounted using the XEX loader (providing the PBI hard disk is enabled). The process is then basically identical to the SIO2xx method, with the advantage that ROMs will be loaded much more quickly.

Note: When updating the PBI BIOS from a PBI-mounted ATR, be aware that you are actually erasing and flashing the ATR handler itself. Because of this, a reboot immediately after a PBI BIOS update is recommended, and if you're flashing all three firmware components, you should update the PBI BIOS last and power-cycle the machine immediately afterwards.

Finally, UFLASH and ROM files may be copied to APT hard disk partitions. The same precautions apply, however, regarding the PBI BIOS as apply to ATR-based updates.

Troubleshooting and Recovering from Flash Problems

While UFLASH is an extremely safe and reliable means of updating the U1MB/Incognito flash ROM, when updating the main BIOS, there is always a small possibility that a 'bad flash' could render the machine unbootable. A bad flash could occur for the following reasons:

- A power failure or system reset half-way through the programming process
- System instability or poor connectivity
- Flashing a corrupt or incorrect BIOS image

UFLASH only allows files with valid headers to be flashed to firmware slots, so it's almost impossible to flash the wrong file to the BIOS slot. System instability, meanwhile, has proved to be to be the biggest obstacle when upgrading to the new firmware. One specific issue especially common on U1MB-equipped 1200XLs is noise on the reset line. This can manifest issues even with the original firmware, but issues tend to be worse with the new firmware because of the new firmware's heightened dependency on properly working hardware.

The solution to the noisy reset issue is simple, however. Solder a 1nF ceramic capacitor between the RST pin of the U1MB (on the reverse side of the board, under the P4 header) and GND (a nearby GND pin of the PLCC flash ROM socket is ideal). The fix prevents noise from the reset line causing the U1MB itself to spuriously reset after the BIOS menu has already been activated, and applies equally well to other Atari models experiencing U1MB BIOS menu screen corruption.

Another simple fix which sometimes works is replacing the 74LS08 IC with one from a different machine. If this doesn't help, sometimes changing the CPU can correct problems.

Other stability issues can be caused by marginal connections on ribbon cables and sockets. On properly installed and stable equipment, however, UFLASH is a safe and reliable means of applying firmware updates.

If you do find yourself with a 'bricked' machine, the only solution is to remove the PLCC Flash ROM and re-flash it in another device. The most common means of doing this is with a USB EPROM programmer. Such devices are now inexpensive and widely available, the MiniPro TL866CS being a good example.

The Splash Screen

By default, the new BIOS displays a 'splash screen' whenever the machine is rebooted (although this can be turned off). The splash screen has a progress bar which fills up while the BIOS waits for a keystroke from the user.

At the bottom of the screen, the selected BIOS menu entry hotkey is displayed, along with the XEX loader shortcut key ('L'). Note that if the system is configured to boot directly to the loader, the loader shortcut will not be displayed. Finally, if the PBI BIOS is enabled, pressing 'D' will invoke the boot drive screen, where the boot drive may be chosen on a 'one-off' basis, temporarily overriding whatever is selected in the hard disk setup menu (see the 'Boot Drive' section).

Note you may press the space bar while the splash screen is displayed to immediately clear it and get on with booting the operating system.

When the splash screen is disabled, one may still make use of any of the available shortcut keys by simply powering on the machine with the selected key held down.

Note: To immediately clear the splash screen and proceed with booting the operating system, press the space bar or Escape key. You can also temporarily disable SpartaDOS X or the GOS (or the XEX loader if the system is to boot directly to the loader) by pressing 'X' while the splash screen is displayed. The ROMs will remain disabled until the next forced cold boot.

The BIOS Setup Utility

The opening menu of the new BIOS setup menu looks like this:

The eight icons represent categorized menus, which you may move between by using the left and right cursor keys or a joystick connected to port 1. To select an option in a menu, use the up and down cursor keys or the joystick to move the highlight bar.

Initially, the menu will display 'Profile 1 reset!', and all options will be set to default values.

Note: Users of prior firmware versions will notice that the highlight is now in the left-hand column rather than the right. This is because – as of version 4.0 - settings are now 'modal', which is to say, they are not in an editable state by default. Enable/disable settings work just as they did before, but anything option with more than two settings (a list, spinner, etc) now enters an edit state when Return is pressed. In the edit state, you can use the up and down cursor keys to cycle through the available settings, and the left and right cursor keys to cycle through fields (for example, moving from the 'hours' field to the 'minutes' field when setting the clock). Once the desired setting is displayed, press Return to confirm or Escape to abandon the edit and reinstate the original value.

To change a setting, move the cursor to the item you want to change, and press Return. An enable/disable will simply flip to the opposing value, but with list items, the highlight bar will move to the right column, allowing you to move through the list with the up and down cursor keys. Once you find the desired setting, press Return again to confirm it. The highlight bar will then move back to the left-hand column, allowing you to select a different option. To cancel a setting change, simply press Escape instead of Return.

Joystick control is somewhat simplified in version 4.0 thanks to the new modal operation. Simply move the cursor to the desired option and press the joystick trigger to toggle a setting on or off, or select an item from a list. In the case of a list, pressing the trigger again confirms the selection.

Settings do not become permanent until saved and are otherwise discarded from the menu upon Reset or when the menu is left by other means (see later). You will be reminded of any unsaved settings when leaving the menu via the reset key or via the selections in the 'Save and Exit' menu at the extreme right. Save settings by navigating to the 'Save and Exit' menu, and selecting one of the available choices with the Return key or joystick button, or by pressing one of the shortcut keys listed at the right-hand side. For example: press 'S' to save settings without leaving the setup menu.

You will notice that some settings are mutually exclusive: for example, Graphical OS (if it is available), is automatically disabled when SpartaDOS X is enabled, and vice versa. Notice also that some menu items are 'dimmed' or 'greyed out', and as a consequence not selectable. Sometimes this depends on the state of other settings (for example, no hard disk menu items can be changed when the PBI BIOS is disabled), and sometimes it depends on the underlying hardware (for instance, the 'XEGS Slot' options are only available when the machine is an Atari XEGS, or at least when the Ultimate 1MB is configured for that machine type).

Over the course of the next few pages, we will look at each menu in turn.

System Settings

The *System Settings* menu (previously pictured) permits the selection of extended RAM size (Disabled, 320KB RAMBO, 576KB CompyShop, or 1088KB RAMBO), one of four operating systems, one of four BASIC ROMs (which may be any 8KB ROM, but occupying the internal BASIC space), one of four XEGS 8KB ROMs (occupying the *Missile Command* slot and only available when Ultimate is configured for an XEGS machine), and the status *of SpartaDOS X*, the WIP Graphical Operating System, and the SIDE XEX Loader. Note that the GOS (Graphical Operating System) will only be available if the Ultimate 1MB has previously been configured for a 192KB build of SDX (SpartaDOS X). Details on how to do this are provided in the 'ROM Configuration' section.

By using UFlash or a compatible ROM editor, it is possible to populate the Operating system, BASIC and XEGS ROM slots with any content desired. In the *OS ROM* slot, the stock Ultimate ROM includes, for example, a Stock XL/XE OS, Atari OS B (PAL), Q-Meg OS, and the XEGS OS.

Note that saved changes to the RAM, OS ROM or SDX/GOS configuration will trigger a mandatory cold restart when exiting the menu by any means, including the Reset key.

Tip: To temporarily disable SpartaDOS X or the GOS when booting, press the 'X' key on cold powerup or prior to an OS reboot. This is somewhat easier when the splash screen is enabled (otherwise you must press – for instance – 'C' and 'X' in very quick succession).

All settings are briefly described below.

System RAM	Select one of four 'System RAM' options to determine the total amount of memory available. Options are '64K' (no extended RAM), '320K RAMBO', '567K CompyShop' and '1088K RAMBO'. Only the '576K CompyShop' extended RAM scheme allows separate CPU and ANTIC banking. Selecting 'Stock' will deactivate the U1MB's extended RAM and make available any stock extended RAM (such as the extra 64K on the 130XE). Changing the RAM size will force an OS restart.
OS ROM	Select from one of four user-defined operating system ROMs. Changing the OS will force an OS restart.
BASIC ROM	Select from one of four user-defined internal 8K BASIC ROMs. Almost any 8K cartridge ROM can be flashed to one of these slots. Changing the BASIC ROM will force an OS restart.
BASIC state	As of BIOS version 2.0, it's possible to suppress the internal BASIC ROM during the OS boot via the 'BASIC state' option, which may set to 'Default' or 'Disabled'. However, this functionality (which merely affects the default state of BASIC; it remains accessible via the usual writes to

bit 1 of PORTB) depends on the PBI BIOS being enabled. When the PBI BIOS is disabled, 'BASIC state' will be dimmed and set to 'Default', meaning that the default state of BASIC (usually enabled) will be preserved. Since disabling of BASIC in this manner is (like the PBI BIOS notice) achieved by on-the-fly patching of the OS display handler, it is not guaranteed to work with custom operating system ROMs. If problems are observed, set this option to 'Default'.

Note: the XEX loader had its own BASIC enable/disable option which will override the main BIOS's setting. Therefore, if you wish to boot via the loader an ATR or load an XEX which requires BASIC, there's no need to first go back into the main BIOS setup menu and enable BASIC should you have it set to 'Disabled'. The main BIOS setting takes precedence again on subsequent normal system reboots.

XEGS ROM

The 'XEGS ROM' setting will be greyed out unless the U1MB board is configured (via a jumper) in 'XEGS' mode. The setting permits the selection of one of four user-defined XEGS 'Game' ROMs. The first 'Game' slot is commonly occupied by 'Missile Command'.

The following settings are mutually exclusive (only one of the three may be enabled at any one time).

SpartaDOS X

This setting controls the state of the U1MB's internal SpartaDOS X ROM. SpartaDOS X (SDX) must be enabled when updating the flash ROM. SDX may be temporarily disabled by issuing the 'COLD /N' command from the DOS command prompt.

Graphical OS

The 'Graphical OS' setting toggles the state of the (optional) 128K GOS ROM. If the SDX ROM is 256K or 320K in size, this setting will be unavailable. The SDX ROM size may be adjusted by the UFLASH tool.

Boot to loader

If you use the system primarily for launching games, you may wish to configure the system so that it always boots directly into the SIDE Loader. Enable 'Boot to loader' to accomplish this.

SIDE Cart ROM
(SIDE/SIDE2 only)

This new setting allows the 'external' ROM of the SIDE/SIDE2 cartridge to be used alongside the PBI hard disk. In prior versions of the firmware, enabling the PBI hard disk always caused the external ROM to be suppressed, but this setting allows it to be used at the expense of 'ATR swap button' functionality.

When the PBI hard disk is disabled, the SIDE Cart ROM setting will be unavailable and set to 'Default'. If the PBI HDD is enabled and the 'ATR swap button' setting is enabled, the SIDE Cart ROM setting will be greyed out and set to 'Disabled'. The only way to make the setting available is to enable the PBI HDD and disable the ATR swap button. One may then choose whether the SIDE cartridge ROM is available or turned off as soon as the system boots.

Note that when the PBI HDD is disabled, no attempt to manage the SIDE cartridge ROM is made, since the SIDE cartridge may not even be present. In these circumstances, leaving the SIDE cartridge attached to the machine will usually result in the system booting the SIDE cartridge's external loader.

Thanks to Eric Bacher's efforts, at the time of writing, patched versions of all four OSS development tools (Action!, MAC/65, BASIC XL and BASIC XE) are available in a 'combo' SIDE/SIDE2 ROM, and modifications to the stand-alone SIDE loader (which is also part of the ROM) allow the cartridge to persistently boot the OSS ROMs. The active ROM should be set with the SIDECFG tool supplied with the toolkit.

System Clock and Features

The System Clock and Features menu allows setting of the system clock and – if the default plugin is installed - selection of the VBXE hardware base address (if present), Stereo POKEY (if present), and Covox.

System date   To set the system date, simply highlight the setting and press the Return key to edit the fields. You can then use the left and right cursor keys to move between fields, and the up and down cursor keys to scroll through the available values for each field. Once the date is correct, press Return again to leave to finish editing.

Notice that when changing the date, the day of the week is automatically updated, and now corresponds correctly with the day of the week shown by the SpartaDOS X TD line.

Unlike other settings, changes to the system clock become permanent immediately (regardless of whether you press Escape after editing or leave the setup menu without saving your settings).

System time   Set the system time using the same keystrokes or joystick input as when setting the system date.

VBXE base   If VBXE is detected, this setting will allow the selection of the area of memory in which the FX core can be addresses in software. Options are 0xD640, 0xD740, and 'Disabled'. Selecting the latter will render the FX core registers completely unaddressable.

This option is useful if the registers of other hardware on the system might otherwise clash with those of VBXE. For example, if you have the CSS 'Black Box' connected (which uses the 0xD6xx range), setting 'VBXE Base' to 0xD740 will resolve any conflicts between the two devices.

Note: this and following settings on the 'System clock and features' menu depend on the presence of the VBXE and Stereo POKEY plugins. Your configuration may differ. See the 'Plugins' section.

NTSC Palette   If VBXE is detected and the default plugin is used, this setting will persistently install an NTSC palette in the first palette slot.

Stereo POKEY   If stereo POKEY hardware is present and connected to the U1MB's 'M0' pin, this setting will turn stereo on or off.

PBI BIOS and Hard Disk

The PBI BIOS and Hard Disk menu is the largest menu of all and provides numerous settings which apply to the integral hard disk (when used with the SIDE/SIDE2 cartridge). We shall discuss the individual options in some detail. Note that because the PBI BIOS is now able to (optionally) provide OS-agnostic high-speed SIO drivers (configured using the next menu: HSIO Settings), the presence of a SIDE2 cartridge is no longer a pre-requisite for PBI functionality.

PBI BIOS
Unlike the original Ultimate Setup which described the PBI BIOS as 'SIDE Hardware', enabling the PBI BIOS does not now automatically imply that the SIDE hard disk has become active. A separate setting ('Hard Disk') must be set to 'Enabled' before the PBI hard disk will become active. When 'PBI BIOS' is disabled, all other items are greyed out and unavailable, but they will revert to their prior settings when the PBI BIOS is enabled again.

PBI device ID
Once the PBI BIOS is enabled, the next available setting is 'PBI Device ID'. This allows the PBI device number (0-7) to be changed (the Atari can manage up to eight PBI devices, each with a unique ID). Only the even ID numbers (0, 2, 4, 6) can be assigned to the Ultimate PBI device.

PBI notice
When 'PBI Notice' is enabled, a boot message will be displayed showing the PBI BIOS version number and the HDD controller's PBI device ID. The notice only appears when SDX is present, although it now appears more predictably than in the past (SDX's base ROM bank need no longer be the one present when the OS boots).

Hard disk
The 'Hard Disk' setting simply switches the hard disk on and off. If the hard disk is off, 'ATR swap button', 'Boot Drive', 'Redirect D1:', 'CONFIG.SYS', 'HDD write lock' and 'HDD rescan key' will be unavailable and deactivated.

Note: the U1MB PBI BIOS makes the SIDE driver (a SpartaDOS X SIO driver allowing access to the SIDE cartridge's hard disk) completely unnecessary. To use the PBI hard disk, place the SIDE's switch in the upper (loader) position. This – together with the PBI BIOS – disables all ROMs on the SIDE cartridge, since none are required (SDX and the XEX loader on the U1MB ROM should be used instead).

Boot drive
Setting 'Boot drive' forces the system to boot from a particular hard disk partition when SDX is not active, or when SDX is active but the 'CONFIG.SYS' option (see later) has not been set. In older firmware

versions, the boot partition was always specified in the partition table on disk, and no provision was made to override this. Now, however, one may override the boot setting in the partition table. Settings are as follows:

- Default: The OS will boot from the default boot drive, which is usually drive 1.
- D1:-DO: Drives 1 through 15. When SDX is disabled, the OS will boot from the specified drive (i.e. HDD partition). If no partition exists on the specified drive number, the OS will boot from the default boot drive (usually drive 1).
- APT: The boot drive number will be taken from the partition table. If no boot partition is specified in the partition table, the OS will boot from the default boot device (usually drive 1).

Note that boot drives should refer to hard disk partitions, not serial disk drives or other devices not managed by the PBI BIOS. If no HDD partition is mounted on the specified drive number, the setting will be ignored and the system will boot from the OS default (usually drive 1).

Redirect D1:

The next item on the menu is 'Redirect D1:'. When the PBI BIOS is enabled, this allows the redirection of all 'D1:' IO requests to the partition specified by the 'Boot drive' setting, and references to the boot drive to be likewise redirected to 'D1:'. References to the two drives are thereby swapped.

This feature will only work if both 'D1:' and the drive referenced by 'Boot drive' are valid hard disk partitions, and since the main BIOS does not know in advance whether this is the case, the setting will silently fail even when enabled if these conditions are not met. If partitions do not exist on both drive numbers, no IO redirection will occur at all.

When 'Boot drive' is set to 'default' or 'D1:', the 'Redirect D1:' setting will be greyed out (since no redirection is then possible).

The envisaged purpose of boot drive redirection is to allow rapid switching between two bootable disk operating systems – for example, MYDOS on partition 1 and SpartaDOS 3.x on partition 2. If partition 1 is assigned the drive ID 'D1:' and partition 2 assigned 'D2:', then enabling and disabling 'Redirect D1:' between cold boots will effectively swap those first two partitions around, allowing the machine to boot from one or the other without the need to persuade the operating system that the boot device is anything else than the usual 'D1:'. Aside from that, <u>the setting should be used with caution, since if enabled unintentionally it will likely cause significant confusion.</u>

CONFIG.SYS

The 'CONFIG.SYS' setting causes SDX to read 'CONFIG.SYS' from the specified drive, which also becomes the default logged drive once SDX has booted. When SDX is disabled (either in the BIOS settings or via 'COLD /N') the CONFIG.SYS setting is ignored and the OS boots from the 'Boot drive' partition instead. If the 'CONFIG.SYS' setting is disabled, SDX will fall back to the 'Boot drive' setting and attempt to load CONFIG.SYS from that drive instead. If both settings are disabled, D1: will be used.

The idea of having separate 'CONFIG.SYS' and 'Boot drive' settings is to allow convenient switching between SDX and another DOS (e.g. MYDOS) without having to change the boot drive setting each time.

HDD write lock

'HDD write lock' is a system-wide hard disk write lock, and has three settings: Disabled, Physical Disk, and Global. 'Disabled' allows writes to all areas of the hard disk. 'Physical Disk' allows writes to partitions (logical drives), but disables low-level, physical disk writes to the MBR, APT records and other areas of the hard disk. 'Global' prevents writes to all areas of the disk, including ATRs. 'Physical Disk' is a useful way of safeguarding the partition table during normal operation, and need only be disabled again when changes are made to the logical disk geometry. Note that one may also disable writes on a per-partition basis by setting a logical drive's lock flag in FDISK.

HDD refresh

This setting toggles use of the 'Shift' key (pressed with the Reset key) as a means of re-reading the partition table from the hard disk, obliterating any ATR mounts and reinstating the partition mapping table as defined on the hard disk. This is useful if you want to revert the partition table to its power-up state after having mounted disk images or mounted or unmounted partitions.

ATR swap button

The ATR swap button – when pressed – rotates the drive numbers of multiple mounted SIDE/SIDE2 disk images (see the section on the *SIDE Loader*). When no ATRs are mounted, or when the button is disabled, it has no effect.

On the U1MB, a side-effect of the ATR swap button being enabled is that the external cartridge is suppressed, and enabling the button will disable the 'SIDE Cart ROM' setting on the 'System Settings' menu. Therefore, if you wish to use the PBI HDD in conjunction with another cartridge (assuming you are somehow able to attach the SIDE and another cartridge at the same time), the ATR swap button must be disabled. Likewise, should you wish the SIDE's external cart ROM to remain active,

you must disable the ATR swap button, and set the 'SIDE Cart ROM' setting (which will become available when the button is off) to 'Enabled'.

The Incognito and 1088XEL/XLD firmware imposes no such limitations on cartridge use, and the swap button may be freely enabled and used alongside cartridges. Indeed, the 1088XEL/XLD currently has no ATR Swap button enable setting; it is active all the time, while on Incognito machines, disabling the button merely renders it inactive in those situations when accidental pressing of the button would be inconvenient.

SIO and CIO Drivers

The SIO and CIO Drivers menu allows configuration of the high-speed SIO driver written by Matthias Reichl (adapted to work with SIO2BT and PCLink devices as well as high speed serial disk drives and other peripherals) and an RTC handler accessible from BASIC and other languages. The PBI BIOS must be enabled to gain access to the SIO/CIO settings, so if all items in this menu are greyed out, enable the PBI BIOS first via the PBI BIOS and HDD menu.

SIO driver

'SIO Driver' activates a serial IO handler in the PBI BIOS which can drive high-speed SIO devices (Happy 1050, US Doubler, XF551, Ultraspeed, etc) or be used with the SIO2BT device at any of three different baud rates to access a number of devices emulated by the *RespeQt* peripheral emulator software and the SIO2BT app for Android devices. It also allows high-speed operation with the PCLink device, and even use of PCLink across an SIO2BT connection.

The four possible SIO driver settings are:

Disabled: No serial IO is handled by the PBI BIOS.

HSIO: Devices selected by the 'SIO devices' setting (excluding tape drives and the 'New Poll' devices) will operate at up to 126Kb/s.

SIO2BT: Devices selected by the 'SIO devices' setting (excluding tape drives and the 'New Poll' device) will operate at standard speed (19.2Kb/s) with an extended SIO2BT command timeout.

HSIO+SIO2BT: A combination of the *HSIO* and *SIO2BT* modes, which is to say: high speed operation and an extended command timeout. Furthermore, in this hybrid mode, SIO2BT is also subject to a high-speed poll, allowing the device to operate at 19.2Kb/s, 38.4Kb/s, or 57.6Kb/s.

In HSIO+SIO2BT mode, any device which responds only at 19200 baud is then subject to the standard high-speed poll, allowing a mixture of high-speed devices and SIO2BT connected devices. Devices which fail to respond at 19200 baud are assumed to be offline and remain 'pending' in the speed table and will be re-polled each time they are accessed until such time as their operating speed can be established.

Note: SIO2BT requires peripheral emulator software (such as RespeQt) to employ use of a software handshake. Because the software handshake analyses the SIO data stream for valid command frames, it's possible that data sent to one device may be misidentified as a valid command frame for another device, causing a bogus write or other unintended operation and therefore risking data corruption. Care should be exercised when using software handshake with other devices on the SIO bus, whether those other devices operate at high speed (where the chance of failure is smaller) or standard baud rates. A safe method of copying data from

SIO2BT to another disk is to use a RAM disk or hard drive as temporary storage and to disconnect Bluetooth before finally copying files to their destination.

SIO devices

Once the SIO driver has been activated using one of the four available modes (HSIO, SIO2BT, or HSIO+SIO2BT), you may use the 'SIO Devices' setting to define the *range* of drives and other devices handled by the SIO driver. There are three settings:

D1-D4
Disks+PCLink
All

The *D1-D4* setting applies the chosen SIO driver mode to disk devices 1-4 (SIO device ID 0x31-34). The exact behaviour of the *D1-D4* setting depends on the SIO driver mode:

In *HSIO* mode, drives 1-4 will be handled using the high-speed SIO code.
In *SIO2BT* mode, drives 1-4 will run at standard speed but with a long SIO2BT timeout
In *HSIO+SIO2BT* mode, drives 1-4 will have the long SIO2BT timeout applied, be subject to the SIO2BT speed-poll, and will operate at high speed (if supported by the hardware), enabling a mixture of HSIO and SIO2BT drives (subject to the prior caveat regarding data loss).

The *Disks+PCLink* setting applies the chosen SIO driver mode to all disk devices (units 1 to 15 on device 0x3x), *and* to PCLink IO requests:

In *HSIO* mode, drives 1-15 and PCLink are handled using high speed SIO code.
In *SIO2BT* mode, a long timeout is applied to all disk drives and to the PCLink device.
In *HSIO+SIO2BT* mode, a long timeout and SIO2BT speed polling are applied to all disk drives and to the PCLink device, and those devices are also allowed to run at high speed.

The *All* setting applies the chosen SIO driver mode to all devices aside from cassette decks and the OS 'New Poll' device:

In *HSIO* mode, all devices are handled using high speed SIO code.
In *SIO2BT* mode, all devices are handled at standard speed (19.2k) using the long timeout.
In H*SIO+SIO2BT* mode, all devices have a long timeout and BT speed-poll, and will operate up to the maximum possible speeds.

SIO driver D1:
SIO driver D2:
SIO driver D3:
SIO driver D4:

Whichever 'SIO devices' range you choose, you may toggle the handling of disk drives 1-4 using the four settings at the foot of the menu, allowing very flexible control of just how the SIO code is employed. Be sure to enable at least one of these four drive toggles when SIO is restricted to 'D1-D4', however, otherwise HSIO will not be applied to any devices at all.

Although the settings may appear complex, in the most common scenario one may simply set 'SIO driver' to *HSIO* and 'SIO devices' to *Disks+PCLink* to obtain accelerated IO with any high-speed devices connected to the machine. Conversely, SIO2BT users should set 'SIO driver' to *HSIO+SIO2BT* and 'SIO devices' to *Disks+PCLink* in the most common circumstances. Advanced configuration is intended to allow for situations where SIO2BT is mixed with other devices, or where there are specific reasons to exclude certain drive numbers from being handled by the custom SIO driver. For instance, if drives 5-15 are commonly polled but are never online, lengthy device polling can be avoided by confining 'SIO devices' to *D1-D4* when SIO2BT is used, and one can further exclude drives 1-4 on an individual basis is further filtering is required.

Below are some examples. If you just wanted high speed operation on drives 1 and 2, without SIO2BT or PCLink, you would use the following settings:

```
SIO driver:        HSIO
SIO devices:       D1-D4
SIO driver D1:     Enabled
SIO driver D2:     Enabled
SIO driver D3:     Disabled
SIO driver D4:     Disabled
```

Note that it doesn't matter if you happen to mount an ATR (on the SIDE's CF/SD card) on D1 or D2 at some point, or if you already have a hard disk partition attached to either of these drive numbers. The ATR or hard disk partition will always work regardless of the SIO settings.

As a second example: imagine you use SIO2BT at 19.2k and you have a large number of emulated drives on a PC or Android device (served by RespeQt or similar peripheral emulation software), but you like to keep the first three drive numbers for hard disk partitions and ATRs mounted using the SIDE XEX loader. You might set things up as follows:

```
SIO driver:        SIO2BT
SIO devices:       All
SIO driver D1:     Disabled
SIO driver D2:     Disabled
SIO driver D3:     Disabled
SIO driver D4:     Enabled
```

Here, the SIO driver is in SIO2BT mode, and will handle serial drives 4-15, PCLink, and all other devices aside from the cassette and the 'New Poll' device at the standard IO baud rate of 19.2Kb/s. Drives 1-3 will be handled by the OS SIO code, or by the DOS SIO driver (in the case of SpartaDOS), or may be devoted to hard disk partitions or PBI mounted ATRs.

A final example: supposing you have SIO2BT set up with mounted ATRs on drives 2 and 3, a Mega-Speedy 1050 on drive 1, a PCLink volume on drive 15, but wanted the OS to handle IO requests on D4:, a suitable configuration would be:

```
SIO driver:        HSIO+SIO2BT
SIO devices:       Disks+PCLink
SIO driver D1:     Enabled
SIO driver D2:     Enabled
SIO driver D3:     Enabled
SIO driver D4:     Disabled
```

Of course, this is subject to the previously mentioned caveat regarding software handshaking.

IO sound

This setting controls IO sounds for serial devices (SIO), FAT-hosted disk images (ATR), and hard disk partitions (HDD). The settings are disabled, SIO, SIO+ATR, and SIO+ATR+HDD. Note the ATR and HDD sounds are simulations of actual SIO noise, and the SIO setting only takes effect on drives handled by the U1MB SIO driver (if you disable IO sound entirely, any devices handled by the OS SIO driver or the SDX SIO driver, for example, will not be silenced).

Z: CIO driver

This setting toggles the presence of a 'New Device' CIO real-time clock driver. The driver is compatible with the ICD handler described in *SpartaDOS Construction Set: R-Time 8 Owner's Manual and Supplement to SDCS Owner's Manual by ICD*. All functions except TD Line On and TD Line Off (XIO 38 and 39) are implemented.

Device Control

The Device Control menu is really a place-holder for BIOS plugin functionality, although the two items ('Device 3' and 'Device 4') do work and will toggle binary 0 and 1 on P4 header pins S0 and S1 respectively (corresponding to bits 2 and 3 of the Ultimate 1MB auxiliary register). Devices 0 and 1 control Stereo Pokey/Simple Stereo and Covox respectively in the default BIOS plugin, but these may also be reassigned or removed if not required, or if the M0 and M1 signals are to be reassigned. You may connect any device which responds to a binary logic signal to pins S0 and S1 and control the device using the 'Device 2' and 'Device 3' settings. Lotharek's 'u-Switch' is a useful aid here, as it converts binary logic to analogue switch logic and even allows logic inversion (which may otherwise be accomplished in software via a BIOS plugin). The author's 1200XL has an internal IDEa HDD interface, various aspects of which (including activation and deactivation via the S0 signal) are controlled by the Ultimate BIOS, using a customised plugin. See the 'Plugins' section.

System Information

The System Information menu simply provides a summary of the hardware installed in the machine.
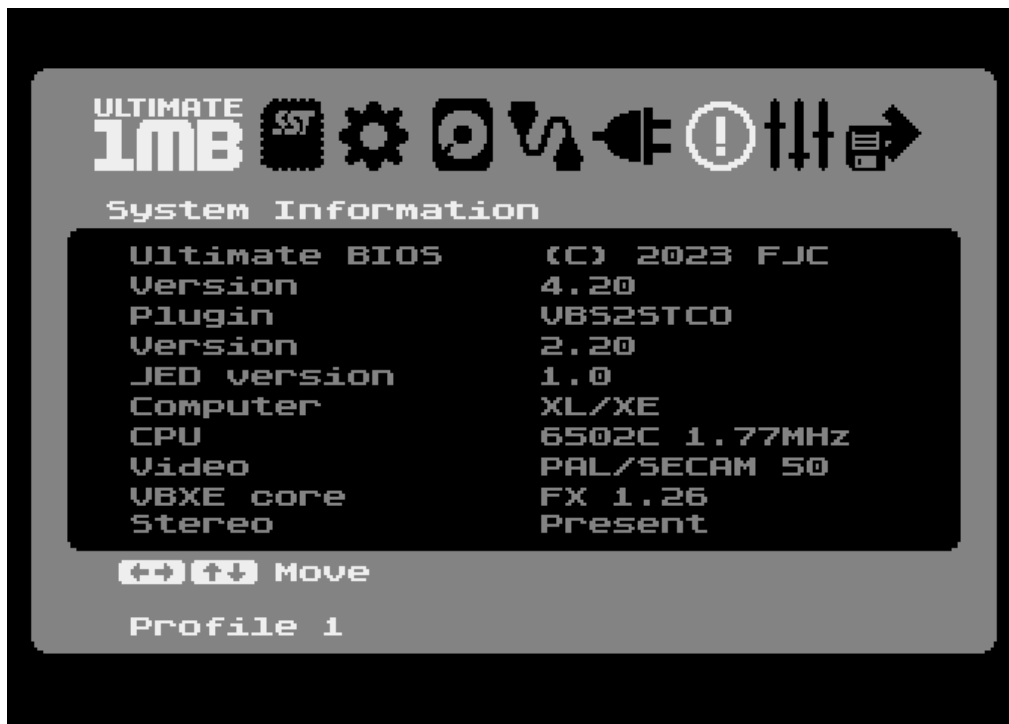
As well and Main BIOS and PBI BIOS versions, the machine type and CPU are reported, along with information about the video and audio. Note that VBXE and SoundBoard will still be detected even if deactivated in the BIOS settings. 'Stereo' will only be displayed if the default plugin or any other plugin which detects and controls Stereo Pokey is present.

As of BIOS v.1.36, both the GTIA video standard (PAL/SECAM or NTSC) and the ANTIC frames per second count are reported in order to accurately represent 'hybrid' systems (those with a mix of PAL and NTSC chips). Starting with version 4.0 of the firmware, the 'JED version' (CPLD core revision) is also reported.

In addition, the GTIA core for VBXE is now properly detected (in prior BIOS versions, only the FX cores could be detected).

BIOS Settings

The BIOS Settings menu allows the customisation of various elements of the BIOS. 'BIOS Menu Key' and 'Cold Boot Key' permit the activation of different system-wide 'hot keys' which cause setup menu entry and a system cold boot respectively when pressed with 'Reset'. By default, the BIOS menu hotkey is 'Help' (or the Atari Logo key on the Incognito), and cold boot and HDD refresh are deactivated. Changing these hotkeys may cause or resolve problems with other devices (for example, the IDE Plus 2.0 uses the Start key with Reset for setup menu entry, requiring use of 'Help' for U1MB BIOS entry).

BIOS menu   Defines the hotkey used to enter the BIOS setup menu. The default is 'Help+Reset' (Atari Logo+Reset on the Incognito), but you can use Start+Reset if you prefer (if, for example, the default hotkey combination triggers some operation in the active operating system).

Cold boot   Defines an (optional) hotkey which triggers a cold reboot of the operating system from outside of the BIOS setup menu. The settings are 'Select+Reset' or 'Disabled'.

Enabling this feature also enables two additional hotkeys. SHIFT+CTRL+ESC+RESET will cause a 'BIOS' reboot: i.e. simulate a power-cycle with the exception that the partition table (if the PBI BIOS is active) will NOT be read from disk (as it is on a cold power-up). This allows use of

the Boot Menu, for example (reached by pressing 'D' while the splash screen is present).

The second hidden shortcut activated by this setting is L+RESET, which will launch the loader. Bear in mind that this also causes an OS restart, so the contents of RAM will be lost.

BIOS logo

The splash screen can now be set to appear at every BIOS reboot, or only when the machine is first powered on. It may also be disabled entirely.

ROM flashing

ROM flashing (commonly via UFLASH) is enabled by default, but you may use this setting to prevent flashing of the 512KB ROM in the Ultimate 1MB (the setting does not exist on the Incognito). Flashing must be enabled prior to any BIOS or other ROM update, however.

Joystick

Allows joystick control of the BIOS setup tool. Port 1 or Port 2 may be used, or joystick input disabled entirely. Disabling joystick input corrects issues when returning to BIOS setup from the Turbo Freezer menu.

Colour

Permits the selection of any of the sixteen Atari hues as the base colour for the BIOS Setup menu. 'Sound' toggles the key click sound in the BIOS menus.

Sound

Toggles the keyclick in the BIOS setup menu. Joystick menu input will also produce the click sound when 'Sound' is enabled.

Screensaver

Toggles a five-minute screen timeout (at which point a banner-type screensaver becomes active).

Save and Exit

The 'Save and Exit' menu displays all the available options for saving current settings, switching profiles, restoring default settings, rebooting the machine, and returning to the underlying operating system or application. Note that the listed shortcut keys work in *any* menu, so – for instance – you can press 'S' while in the 'System Settings' menu and settings will be saved to the current profile.

Note: the SIDE Loader option (which previously had a menu all to itself) now lives in the 'Save and Exit' menu. This was done to free up an extra menu panel for the SIO settings (it being rather wasteful to have an entire menu comprising only a single item). Of course, the loader may still be launched by pressing 'L' anywhere in the menu system, as before.

Certain changes to settings will automatically cause a cold reboot even when a warm reset was desired (i.e. when a reboot is not explicitly requested). Changing and saving the memory configuration, operating system, SpartaDOS X status, or Graphical OS status will always cause a cold reboot, since DOS and/or the OS must then reconfigure itself in fairly dramatic ways. If you leave the menu by pressing *Reset*, no warning will be given of the pending cold reboot. However, if you leave the menu via the *Escape* key, another shortcut key, or menu item, you will be warned of the forthcoming restart.

'Boot diagnostic cart' is an advanced feature suggested by Phaeron, and runs a diagnostic cartridge (if present) with the Ultimate configuration unlocked and the BIOS Setup menu still present in ROM (instead of the Atari's operating system). This is envisaged as a useful tool for flash recovery and diagnostics. Note you can also trigger the diagnostic feature by booting the machine with Shift, Control and Help pressed.

The 'Profile' menu item is described in the next section.

Configuration Profiles

In the 'Save and Exit' menu, selecting 'Profile' (or pressing 'P' anywhere in the BIOS setup tool) will switch between four configuration profiles (named '1'-'4'), an increase from the three profiles available in firmware version 2.0. You will be reminded to save any pending changes to the current profile before switching to a different one. The profiles allow the storage of completely different machine configurations which can be swapped with a single keystroke. This might prove useful if, for example, you like to use both MYDOS and SpartaDOS X with the hard disk feature. You can have different HDD boot partitions in each profile, with SDX disabled in one but enabled in the other. Likewise, you might use one profile for gaming (perhaps with the HDD disabled) and another for productivity.

As well as cycling through profiles with the 'P' key, you can jump directly into the desired profile by pressing '1', '2', '3' or '4'.

Boot Drive Selection

While earlier versions of the alt BIOS provided a simple method for one-off selection of the boot drive during a cold start, latterly this facility has been extended to provide a special boot drive selection menu which is – as before – accessed by pressing 'D' during a cold power-up or force reboot. You'll see the boot drive shortcut displayed on the splash screen (if enabled) when the PBI hard disk is active.

The boot drive menu displays a map of all fifteen available drive numbers, each with an associated shortcut key.
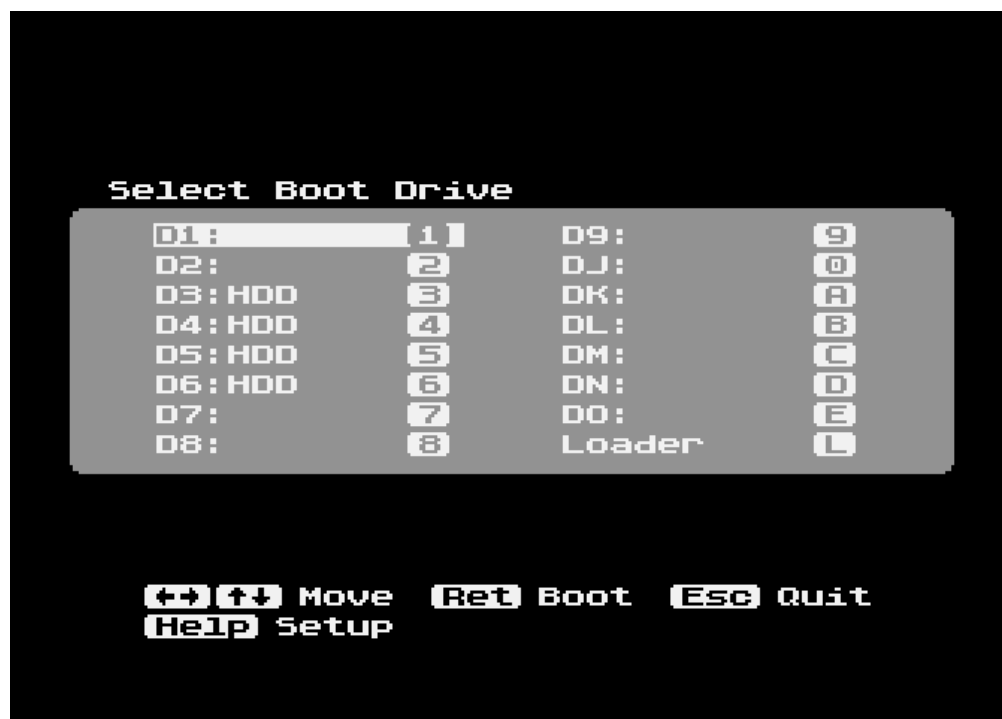
To boot the operating system from the desired drive, you may either move the cursor to it (using the keyboard or joystick) and press Return, or press the corresponding shortcut key.

Note that when the boot drive menu is invoked during a cold power-up, no partition information will be shown since at that stage the OS has not yet initialised the PBI hard disk firmware. On subsequent OS restarts, however, the boot drive menu will show 'HDD' or 'ATR' next to drive entries if they represent hard disk partitions or mounted disk images respectively. The final item in the menu, following the drive number list, is 'Loader'. Selecting this entry will launch the SIDE loader.

You may also boot as normal (without changing the boot drive) by pressing Escape, and to access the BIOS setup menu, simply press Help/Start (for U1MB) or the Atari Logo key/Start key (Incognito).

Note: the boot drive menu provides a means of booting from the specified drive on a 'one-off' basis. To persistently boot from a given drive, you should set the boot drive in the PBI BIOS menu.

When SpartaDOS X is active, setting the boot drive causes DOS to read CONFIG.SYS from the specified drive. This temporarily overrides the 'CONFIG.SYS' BIOS setting (until the next OS restart).

ROM Configuration

Because the original SpartaDOS X build for Ultimate/Incognito did not leave sufficient space for a 128KB Graphical OS, the option now exists to use a 192KB SDX build (prepared with the SDX Imager). Moreover, users not interested in running the GOS may use the first 320KB of the flash ROM to house an enlarged build of SpartaDOS X. The latest version of UFlash allows configuration of the SDX slot sizes (192, 256 or 320KB), and the new BIOS automatically deactivates the GOS slot if anything larger than a 192KB SDX ROM is used.

Editable Slot Descriptions

Since the 'OS ROM', 'BASIC ROM' and 'XEGS ROM' slots house user-defined content (four 8KB ROMs each for BASIC and XEGS, four 16KB ROMs for the OS; Incognito's editable slots are 'Colleen OS' and 'XL/XE OS'), it's possible to edit the titles of the slots to reflect their content.

Flash ROM editors such as UFlash allow editing of the slot names, which must then be re-flashed to the BIOS ROM. As well as merging any existing slot descriptions into a new BIOS ROM, UFlash will attempt to preserve slot descriptions from the original Ultimate or Incognito BIOS when upgrading to the new BIOS. If slot descriptions do not appear as expected following the BIOS upgrade, please let me know so that I can try to ascertain the cause.

There are a couple of important differences to consider regarding the new slot description encoding. Firstly, Ultimate slot descriptions may now be one byte longer than was permissible with the original BIOS (fifteen bytes instead of fourteen in the original U1MB BIOS). The U1MB description size was increased in order to properly accommodate the 'Missile Command' title of the default XEGS slot

(which was previously named 'Missle Command'. Incognito slot descriptions, meanwhile, were always fifteen characters long, and have therefore not changed.

Secondly, slot descriptions are no longer encoded using internal character values. ATASCII is now used. Strings should be NUL terminated and of no more than fifteen characters long (sixteen bytes including the NUL terminator). Readers interested to know how slot descriptions and other metadata are encoded in the BIOS ROM may refer to the accompanying technical/developer documentation.

Plugins

Plugins provide absolute flexibility regarding use four of the IO signals on the Ultimate 1MB, and can add a wealth of extra functionality and hardware support to the core firmware. The default plugin (Stereo/VBXE/SIDE) mimics the behaviour of the original Ultimate BIOS and controls Simple Stereo via the M0 signal on the P4 header. The default plugin also adds an 'Stereo' entry to the System Information menu and generic toggle settings for the three other P4 signal pins.

As of firmware version 3.0, all device-specific hard disk functionality was moved to the plugin. This means that – for example – facilities specific to the SIDE/SIDE2 cartridge (i.e. the 'ATR Swap button') may be removed and replaced with functionality targeting a different hard disk host adapter (such as the XEL-CF adapter used in the 1088XEL).

As of firmware version 4.0 (which saw the introduction of plugin support on the Incognito when a suitable CPLD update is detected), VBXE support has also been moved to the plugin, and the plugin size itself has been increased from 1KB to 2KB. Because plugin support on the Incognito requires a CPLD update, both the U1MB and Incognito firmware will now work without any plugin present at all (i.e. nothing at all flashed to the plugin buffer). Note that stereo POKEY detection on the Incognito is now unavailable on boards which are not 'plugin ready'.

By swapping the default plugin for a different one, it's possible to control up to four user-defined hardware devices using the P4 logic signals. Plugins may also perform any hardware tests required on system initialisation, set up menu items in a contextual manner, and perform hardware writes on system reset. Forty bits of NVRAM configuration data are also reserved for plugins, so they may store data in the battery-backed RAM for retrieval by one of the target devices.

Plugins must carefully observe the guidelines described in the plugin developer guide, which is part of the technical documentation for the new firmware. **Note: plugins from prior versions of the firmware are totally incompatible with firmware version 4.0.**

What follows is a brief description of the most common plugins. The letters in parenthesis are the plugin ID (you can see the IDs for all the installed plugins in the 'System Information' menu).

VBXE Plugin (VB)

The VBXE plugin provides a setting for the VBXE core base address ($D6xx/D7xx/disabled) in the 'System clock and features menu', and the larger VBXE plugin includes an 'NTSC palette' setting which, when enabled, automatically installs an NTSC palette when the machine is powered on.

Stereo POKEY Plugin (ST)

The stereo POKEY plugin allows the toggling of stereo audio on the M0 pin on the U1MB, as well as providing stereo hardware status in the 'System Information' menu. This plugin may be used with any stereo POKEY board which implements a logic-driven switch which may be connected to the M0 pin on the U1MB (this includes several 'Simple Stereo' boards and PokeyMAX with the U1MB Stereo Control core flashed).

Covox Plugin (CO)

The COVOX plugin allows the toggling of COVOX hardware on the M1 pin on the U1MB, and is specifically intended for Candle O'Sin's COVOX (SimpleStereo) board. Another situation in which this plugin proves useful is with the Altirra emulator, which disables *any* emulated COVOX hardware on U1MB systems unless the M1 signal is set to 'enabled'. Therefore, if you are emulating an U1MB system with any COVOX hardware present, you should use this plugin.

On real hardware systems with other COVOX hardware not connected to the U1MB M1 signa (for example, PokeyMAX2/3), you should use the appropriate plugin (e.g. the PokeyMAX plugin).

PokeyMAX Plugin (P2)

The PokeyMAX plugin is designed for PokeyMAX 2 and PokeyMAX 3 boards with anything EXCEPT the U1MB Stereo Control core (for which you should use the standard Stereo POKEY plugin) and the 'Auto' Stereo core. All other PokeyMAX cores (Stereo/Covox, Quad/Covox, SID, PSG, etc) will benefit from the PokeyMAX plugin, which allows full software control of the several of the most commonly changed audio settings. Features such as Stereo, Quad, Covox, linear mixing, mono detect, SID emulation, PSG, etc, may be toggled via settings in the 'Device control' menu. Since PokeyMAX has a 'features' register, only those settings relevant to the installed core are activated (the rest being dimmed). Moreover, U1MB will maintain separate PokeyMAX settings in each configuration profile, allowing convenient switching between up to four different PokeyMAX configurations.

Sophia 2 Plugin (SO)

The Sophia 2 plugin allows direct management the Sophia 2 video output settings from within the U1MB menu and includes 'test modes' which provide a safe way to verify the output settings before committing them. The following settings are available:

Resolution

Available settings:

720x480p/576p (3:2/5:4)
1280x960 (4:3)
1280x1024 (5:4)
1344x960 (14:10)
1440x900 (16:10)
1536x960 (16:10)
1600x900 (16:9)
1704x960 (16:9)

A change in resolution will result in a test of the newly selected resolution for a few seconds. The newly selected resolution needs to be accepted by pressing RETURN. If RETURN is not pressed, the video output will revert to the previously selected (active) resolution.

VGate

When enabled, VGate limits the width of the playfield by removing borders (playfield is then 336 pixels wide).

2 colour hi-res

This setting allows for bi-colour hi-resolution graphics and text modes. For example, when 2 colour hi-res is enabled, graphics 0 can display characters in a different colour to the background.

16 luma levels

When enabled, all 16 luminance levels can be used for all graphic modes.

RGB output

Five settings are available for the analog output on the DVI connector:

VGA
RGB (Csync)
RGB (H/Vsync)
YPbPr (Csync)
YPbPr (H/Vsync)

Interlace

When the RGB output is set to either RGB or YPbPr, either progressive (interlace disabled) or interlace mode can be selected. Note that Interlace mode also affects legacy (s-video/composite) output, so use this setting with care.

With each change in video output, changes need to be committed by saving the changed settings to the current profile. If settings are not saved to profile, they will be lost with the next soft or hard reboot. Note this additional save to profile step was purpose built to prevent users from accidently selecting a video output signal not accepted by the connected monitor.

Note also that the Sophia 2 plugin stores the video settings in the Sophia's NVRAM rather than the U1MB NVRAM, and for this reason multiple Sophia 'profiles' are not maintained (which is to say, each U1MB configuration profile will always reflect the same Sophia configuration). In addition, note that because the Sophia settings do not exist in the U1MB NVRAM, you will not be prompted to save unwritten changes to the Sophia configuration. Please remember to save them before leaving the U1MB setup menu.

SIDE Plugin (S2)

The SIDE plugin manages the SIDE or SIDE2 cartridge when the PBI HDD is enabled, and adds the 'ATR swap button' setting to the 'PBI BIOS and Hard Disk' menu. It also adds the 'Z: driver' setting to the 'SIO and CIO drivers' menu. In addition, and extra setting will appear on the 'System Settings' menu: 'SIDE Cart ROM'. This setting only applies when the ATR swap button is disabled, and permits enabling and disabling of the SIDE2 cartridge ROM (this facility is primarily used to run Ebiguy's patched OSS ROMs). When the ATR button is enabled, the firmware transparently suppresses the ROM on the cartridge at all times, making the position of the cartridge switch unimportant.

When the PBI HDD is disabled, the plugin performs no probing for the SIDE cartridge, thereby ensuring that other cartridges are not affected by unwanted access to the CCTL area.

The plugin depends on the presence of the SIDE PBI BIOS in the U1MB's 'PBI BIOS' ROM slot.

SIDE3 Plugin (S3)

The SIDE3 plugin manages the SIDE3 cartridge when the PBI HDD is enabled, and adds the 'ATR swap button' setting to the 'PBI BIOS and Hard Disk' menu. In addition, and extra item appears on the 'Save and Exit' menu: 'Remove emulated cart'. This action allows quick un-mounting of any SIDE3-hosted cartridge image previously mounted via the SIDE3 Loader.

The plugin also handles detection of the SIDE3, and causes the SIDE3 Loader on the cartridge to be launched instead of the U1MB's built-in SIDE Loader when the user presses 'L', chooses 'Loader' via the menu, or configures the system to 'Boot to loader'. When the PBI HDD is disabled, no probing of the CCTL area is performed (this allows other cartridges to function properly).

The plugin depends on the presence of the SIDE3 PBI BIOS in the U1MB's 'PBI BIOS' ROM slot.

XEL Plugin (X3)

The XEL plugin is usually employed on the 1088XEL/1088XLD machines in order to manage the XEL-CF2 and XEL-CF3 Compact Flash adapters, but the plugin may also be used on 'standard' Ataris if and when compatible versions of the XEL-CF adapters become available. The plugin adds the 'ATR swap button' and 'Slave drive' settings to the 'PBI BIOS and Hard Disk' menu, and the 'Z: driver' setting to the 'SIO and CIO drivers' menu.

The plugin depends on the presence of the XEL-CF PBI BIOS in the U1MB's 'PBI BIOS' ROM slot.

Installing Plugins

Plugins are supplied a 2KB (as of firmware version 4.0; plugins for prior versions are 1KB in size) modules which should be flashed to the 'BIOS Plugin' slot using UFLASH.

After flashing a new plugin, all plugin-specific settings in the firmware setup menu will be reset to default values on first use. Note that other firmware components (the PBI BIOS, the loader) can be dependent on plugin components. For example: unless the SIDE3 plugin is installed, the U1MB firmware menu will be unable to start the SIDE3 loader. Likewise, unless the SIDE2 plugin is installed, the U1MB SIDE PBI BIOS will not work correctly.

The default plugin selection comprises SIDE2, VBXE, Stereo POKEY and Covox.

The SIDE/SIDE3 Loader

Together with a new setup menu, the revised U1MB/Incognito firmware brings with it a new SIDE Loader. You can launch the loader in a number of ways:

- In the 'Save and Exit' menu, navigate to 'XEX Loader' and press Return
- Press 'L' anywhere in the BIOS setup menus
- Selecting 'Loader' from the BIOS boot drive menu
- Pressing 'L' on the BIOS splash screen (if enabled) or starting the Atari with 'L' held down
- Enabling the 'Boot to loader' option in the 'System Settings' menu

The 'Boot to loader' method will persistently start the loader on every cold start or reboot until the option is disabled. Note also that starting the computer with 'L' naturally only applies if the system is not already set to boot direct to the loader.

When you start the new XEX loader with a suitably formatted CF/SD card in the SIDE cartridge (or on-board CF card slot if using Incognito or the 1088XEL with an XEL-CF adapter), you'll be confronted by the launcher menu:

The loader contains eight menus, just like the main BIOS setup tool. By default, the loader opens the third menu (the launcher) if it's able to open a FAT partition. The launcher menu presents an alphabetised list of all the files handled by the loader contained in the root directory of the first (or last accessed) FAT16 or FAT32 partition on the hard disk. The loader displays subdirectory names and files with the following extensions:

To launch a program or disk image or open a directory, simply use the cursor keys or the joystick to highlight the desired item and press the Return key or the joystick button.

Movement, Selection and Shortcuts

You may control the loader using the keyboard or joystick (joystick control must be enabled in the Options menu for the latter to work). Methods of movement and control are as follows:

Action	Keyboard	Joystick	1200XL F-Key
Previous Menu	LEFT ARROW	LEFT	F3
Next Menu	RIGHT ARROW	RIGHT	F4
Previous Item/setting	UP ARROW	UP	F1
Next Item/setting	DOWN ARROW	DOWN	F2
Page Up	SHIFT+CTRL+UP ARROW	N/A	SHIFT+F1
Page Down	SHIFT+CTRL+DOWN ARROW	N/A	SHIFT+F2
Select/Open/Edit	RETURN	Button	CTRL+F3
Select/Open/Edit with modifier	CTRL+RETURN/SHIFT+RETURN	N/A	CTRL+F4
Set Drive Number	TAB	UP with button	CTRL+F2
Tag with next drive number	CTRL+SPACE	DOWN with button	CTRL+F1
Tag with specified drive	CTRL+Number (1-9)	N/A	N/A
Cancel drive selection, exit search mode, or back up to the parent directory	ESC	LEFT with button	N/A
Delete the previous character of the search phrase	DELETE/BACKSPACE	N/A	N/A
Clear the search phrase	CTRL+CLEAR	N/A	N/A
Start of List	CTRL+A	N/A	SHIFT+F3
Toggle BASIC	CTRL+B	N/A	N/A
Refresh Disk(s)	CTRL+D	N/A	N/A
Open location of logical drive image or found item	CTRL+F	N/A	N/A
Home (Root) Folder	CTRL+H	N/A	N/A
Next 250 entries of long directory (SIDE/XEL)	CTRL+M	N/A	
Parent Folder	CTRL+P or ESC	N/A	N/A
Restart	CTRL+R	RIGHT with button	N/A
Swap ATRs	CTRL+S or ATR Swap Button	N/A	N/A

Action	Keyboard	Joystick	1200XL F-Key
Undo Mounts	CTRL+U	N/A	N/A
Reboot to SpartaDOS X (U1MB/SIDE3 only)	CTRL+X	N/A	N/A
End of List	CTRL+Z	DOWN+RIGHT	SHIFT+F4
Abort search	BREAK	N/A	N/A

Since version 1.28 of the loader, joystick movement actions performed in conjunction with the button will auto-repeat just like those without the button. In addition, versions 1.28 and beyond have a built-in type-ahead keyboard buffer.

Starting with version 3.1, keyboard and joystick input on the up/down axis will accelerate after 20 repetitions. This primarily to allow faster movement up and down a long list of filenames via joystick input without requiring the page up/down keys, but also works with direct keyboard input. You can disable the feature via the Options menu.

Function Modifiers

CTRL held with the Return key has a special meaning depending on the selected item's type and the context of the operation:

Item type	RETURN	SHIFT+RETURN/CTRL+RETURN
XEX file in logged folder	Run in logged folder	
XEX file in search results	Run in own folder	Run in logged folder
ROM or CAR file (SIDE3 only)	Mount and reboot	Mount
BASIC program in logged folder	Run in logged folder	
BASIC program in search results	Run in its own folder	Run in current folder
ATR	Mount and reboot	Mount
MAP file	Mount and reboot	Mount

Joystick Control Quick Reference

For reference, joystick actions with the trigger released are as follows:

Up

Left · Right
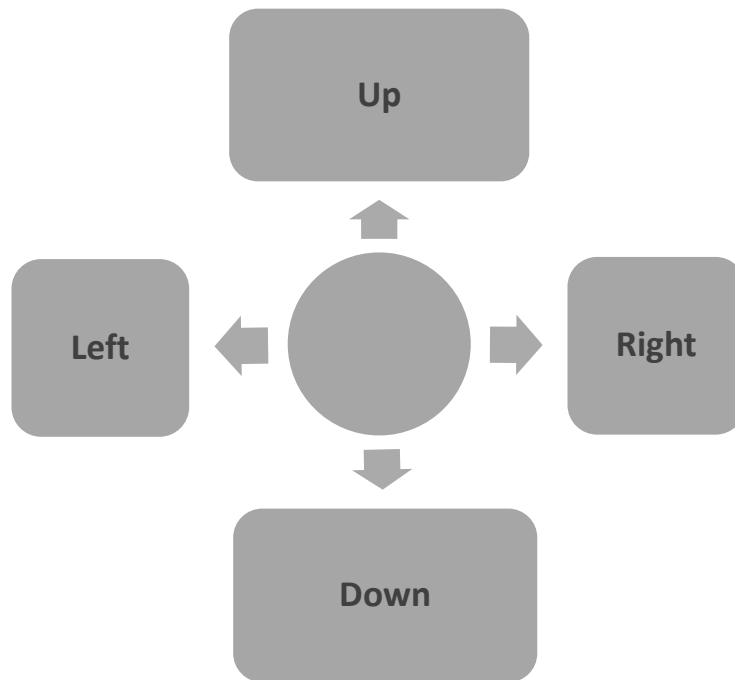
Down

Joystick actions with the trigger held down are as follows:

Set Drive

Cancel · Open · Reboot

Tag Image
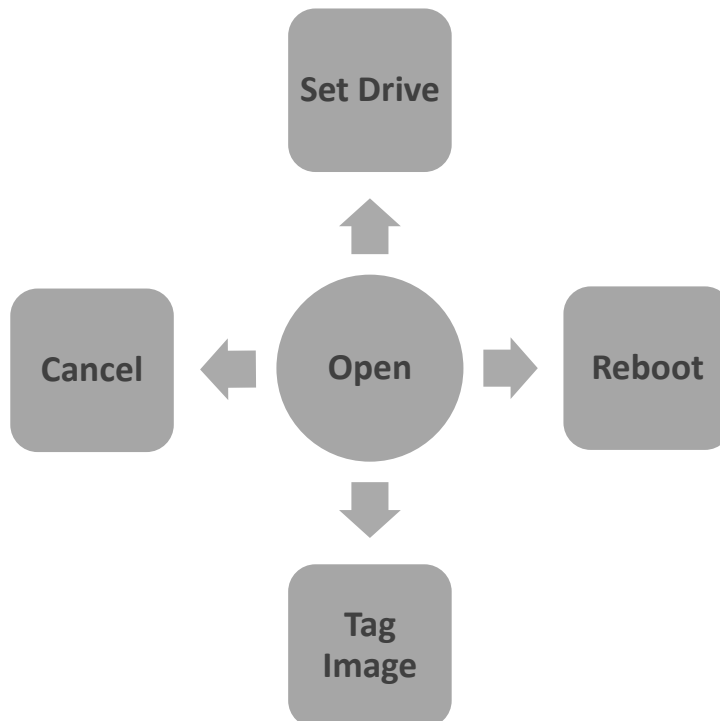
Device Menu

Using the left cursor key, move the menu icon highlight to the leftmost icon to open the 'Device' menu. This menu displays the physical controllers attached to the system which are recognisable to the loader, i.e. 'SIDE' or 'SIDE3', followed by the manufacturer and/or model of currently inserted CF/SD card. Pressing Return on the chosen device name logs all partitions on the device and opens the partition menu (if there is more than one partition on the card) or the FAT/APT browser menu (if there is only one partition on the card).

Partition Menu

The second menu is the Partition Menu, and this displays a list of all recognized partitions on the hard disk. The title of the menu is the device name of the selected physical disk, as per the device selected in the 'Device' menu. The list displays FAT16 partitions, FAT32 partitions, and the APT partition, if present (which contains any Atari partitions created using the FDISK partition editor).

As usual, you can move the highlight using the up and down cursor keys, and open a partition by pressing Return. When you open a partition, the contents are immediately displayed in the launcher menu. If you open the APT partition, the Atari partitions contained therein will be displayed in the launcher menu, where you may mount partitions by selecting them.

You may notice a number appearing next to partition names after ATR disk images have been mounted. The number denotes the host partition ID, which is the ID of the FAT volume which contains the mounted ATR image(s). Up to four host partitions can be registered, and this is done automatically by the loader as disk images are mounted. If you have many partitions on the card, it's theoretically possible to exhaust the four-host partition limit, but even in the unlikely event that this

happens, logging the disk or unmounting all partitions (with "Refresh disk" or "Undo mounts" respectively) unregisters all the FAT partitions, enabling one to start afresh.

Launcher Menu

The launcher menu (symbolised by a 'rocket' icon and whose actual title is the volume name and absolute path of the current directory) is displayed when the loader starts and allows loading of XEX files and the mounting of ATRs and APT partitions. In the SIDE3 loader, you can also use the launcher menu to catalogue and mount CAR and ROM files (cartridge images). The launcher has two 'views': FAT and APT Partition. You'll see FAT View when you open a FAT partition, and APT Partition View when you open the APT container.

FAT View

FAT view displays all the folders and supported files in a FAT16 or FAT32 partition.

Long filenames are displayed and names too long to fit entirely on the screen are abbreviated with an ellipsis ('…') in the middle of the name. When you highlight such entries, they begin to scroll slowly left and right so that you can see the filename in its entirety. Beginning with version 3.1, the file extension remains in a fixed position even while a highlighted filename scrolls left and right. Folders are denoted by a 'folder' icon in the left margin, while any mounted ATRs are denoted by the current mount point (i.e. drive number) next to their name.

File Types

The launcher displays the following file types:

Extension	File Type
XEX	Binary executable
EXE	Binary executable
COM	Binary executable
OBJ	Binary executable*
OBX	Binary executable*
ATR	Disk image (only if U1MB is present and the PBI hard disk is enabled)
BAS	Tokenised BASIC program
MAP	Script describing multiple ATR mount points
ROM	Raw cartridge ROM image**
CAR	Cartridge ROM image with header describing banking scheme/geometry**
PDM	Pulse Density Modulated audio (FujiConvert) files**
PDS	Stereo PDM files**
COV	Covox files**
COS	Stereo Covox file**

*Newly supported in version 3.1
**SIDE3 only

The file types are described in detail below.

BASIC files (*.BAS)

Tokenised BASIC programs (typically run under Altirra BASIC or Atari BASIC) may be launched by the loader by means of the built-in FAT file system handler. If the file system is disabled, it will nevertheless be installed when running a BASIC program, since the it is via the FMS that the BASIC program is loaded. Once the BASIC program is running, it has full read-only access to the FAT partition from which it was launched. The CIO FMS handler is reset-proof, and calling DOS from BASIC re-runs the loader (ROM version only; an exit to DOS does nothing at all in the disk-based version of the XEX loader).

On an Incognito machine running in Colleen mode, you will need to manually enable the 'BASIC Cartridge' in the BIOS setup menu.

Executable files (*.COM, *.EXE, *.XEX, *.OBJ, *.OBX)

Executable files are loaded by code residing at 0x0700 which does not require the file system handler to be present. However, enabling the file system handler provides read-only CIO file access

to launched programs. One application for this is launching UFLASH.XEX from the XEX loader, and then loading ROM files from the FAT partition. The Ultimate 1MB/Incognito flash ROM may thereby be updated without booting DOS and without copying large ROM files into Atari partitions (they need simply be copied to a FAT partition instead).

When the file system handler is not present (i.e. disabled in the Options menu), MEMLO resides at approximately 0x0A00 (the loader code being some 768 bytes long). When the file system handler is enabled, MEMLO sits at around 0x1680, although XEX files may still overwrite anything beyond the end of the loader (0x09FF) without problems during the launching process. If an application is designed to work with DOS (i.e. loads beyond 0x1F00 or 0x2000 and implements standard CIO calls), it should load and work with the read-only FAT FMS.

On an Incognito machine running in Colleen mode, you will need to manually disable the 'BASIC Cartridge' in the BIOS setup menu in order to run most XEX files, since software control of said BASIC ROM is not possible.

Disk images (*.ATR)

ATR disk images are mounted by the loader only if the PBI Hard Disk is enabled in the main BIOS setup menu. ATRs are not available on the Incognito running in Colleen mode. After mounting an ATR, the loader will usually restart the operating system and boot the system from drive 1 (D1:). As well as mounting and launching ATRs with the Return key, more precise control is possible. See the later section on Image and Partition Mounting for more information.

Map files (*.MAP)

Map files describe the mount points for ATRs. Map files are plain DOS/Windows/Unix text files containing lists of drive specifiers and ATR filenames. See the section on Image and Partition Mounting for more information.

Cartridge files (*.ROM, *.CAR) (SIDE3 only)

The SIDE3 cartridge is capable of mounting cartridges of many different types in CAR and ROM formats. CAR files are preferred since they include a header with a description of the banking scheme (if any) used by the cartridge, whether it is a left or right slot cartridge, whether it is an 8K or 16K cartridge, etc. CAR files – if of a supported type – will automatically mount without any further input from the user. ROM files lack the header information, so unless the loader can guess the cartridge type from the size of the file or other characteristics, it will present you with a list of compatible cartridge types from which you should choose whatever is appropriate for that cartridge image.

Cartridge images can be made 'persistent' (i.e. only being unmounted when you explicitly unmount the image) or can be forced to unmount automatically every time you restart the loader. See the 'Unmount carts' setting in the 'Options' menu. By default, a mounted cartridge automatically unmounts when you re-enter the loader.

By default (when cartridge images are launched with the RETURN key), a reboot is immediately performed. If you wish to mount a cartridge image without an immediate restart, simply mount the image with CTRL+RETURN. You can then perform other tasks (such as mounting ATRs, etc) or reboot into SpartaDOS X with CTRL+X.

PDM/COVOX audio files (*.PDM, *.PDS, *.COV, *.COS) (SIDE3 only)

Version 0.34 of the SIDE3 Loader can play mono and stereo Pulse Density Modulated (PDM) and Covox audio files created by 'FujiConvert' (https://lybrown.github.io/fujiconvert/). Select 'IDE Player (flashjazzcat)' as the output format in the FujiConvert tool:

Needless to say, Covox playback is only possible on a Covox-equipped machine. You should use the following naming convention on files created with FujiConvert:

Extension	File Type
PDM	44KHz mono PDM
PDS	22KHz stereo PDM
COV	44KHz mono COVOX
COS	22KHz stereo COVOX

You may specify the base address of the Covox hardware with the 'Covox base' setting in the 'Options' menu. PokeyMAX users with a Covox-enabled core (and Covox enabled in their PokeyMAX configuration) will find the 'Covox base' setting auto-populated and greyed out (so there is no need to manually set the Covox base address).
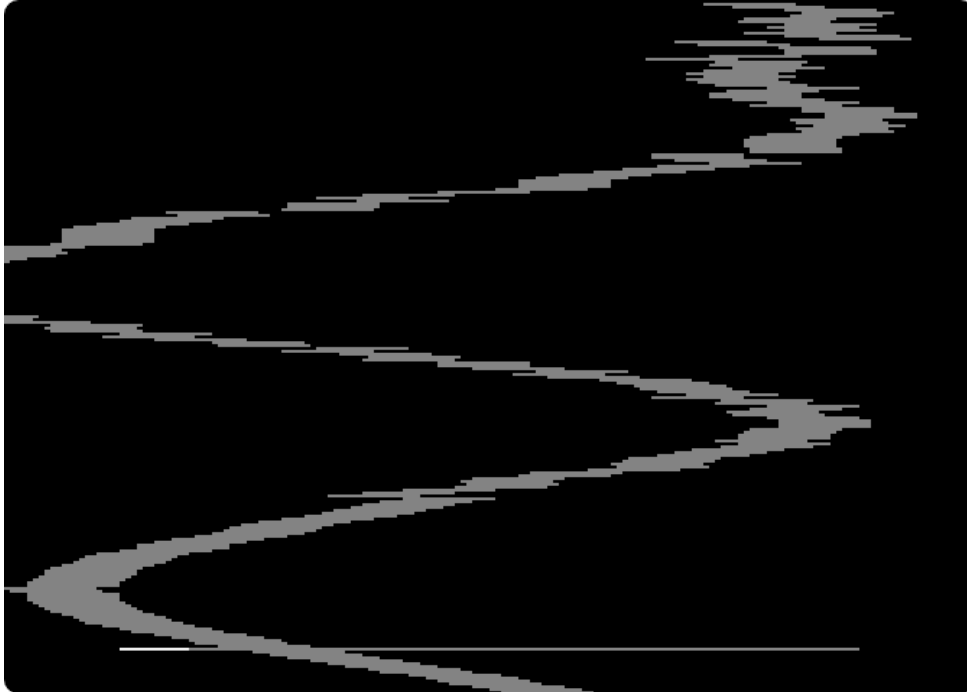
In the SIDE3 PDM player, you can press the left and right arrow keys during playback rewind and fast forward, and tap the space bar to pause and resume playback. The ESC key abandons playback and takes you back to the file browser.

Directory Tree Navigation

Subdirectory names in the launcher menu are grouped together at the beginning of the file list and are prefixed with 'folder' icon. To open a subdirectory, simply highlight it and press Return or the joystick button. Once inside of a subdirectory (i.e. not in the root directory), two additional entries will appear right at the top of the file list: the parent folder entry (a crooked left-pointing arrow) and the root folder entry (a straight upward-pointing arrow). Opening the parent folder will take you one level back up the folder tree and opening the root folder entry will take you back to the root folder. Note that when you enter a subdirectory and then back up out of it again, the subdirectory name will remain highlighted in its parent folder.

As of version 3.0 of the loader, you may reach the parent of the logged folder by pressing CTRL+P (as an alternative to 'Esc'), and the home (root) folder by pressing CTRL+H. The 'home' shortcut may be used even when in search mode; it will cancel the search and return to the top of the root directory.

Handling Large Directories

From version 3.0, the loader is able to handle directories containing more than the previous maximum of 250 files and subdirectories. It does this by paging through longer directories 250 items at a time.

When you first open a directory containing more than 250 items, a '…' entry will appear at the very top of the filename list (visible here immediately above the parent directory symbol). A number will also be displayed immediately after the item count, denoting the sequential numbered segment of the directory currently displayed.

If you open the '…' entry by highlighting it and pressing RETURN (or by pressing CTRL+M at any time), the next 250 entries from the same directory will be catalogued and displayed and the segment number shown after the item count will increase. Once the end of the directory is reached, you'll notice that the catalogued item count drops below 250, indicating that the loader has just catalogued the last entries in the folder. At this point, selecting the '…' symbol or pressing CTRL+M causes the directory to 'wrap around', and you'll again see the first 250 entries in the folder (segment '[1]').

Note that when dealing with lengthy directories such as these, you may not always end up viewing the start of the directory by default. For example: if you load entries 251-400 and enter a subdirectory in that range, when you later back up out of that subdirectory (to display its parent), the segment of the long directory containing the child directory you have just left will be displayed, with the child entry highlighted. Likewise, and search result displayed in its containing folder (via CTRL+F) may cause a higher segment of its directory to be displayed unless it resides within the first 250 entries.

Large Directories in the SIDE3 Loader

Since the SIDE3 cartridge has a large amount of on-board RAM, the SIDE3 Loader is able to read directories of up to 64KB in size (or 1,024 filenames; whichever limit is reached first). There is therefore no need to bother with paged directories, etc.

When reading a long directory (which may take a few seconds to fully process), the loader will report the progress of the directory read operation for every 256 filenames read into memory. Note that when the filename buffer is full (either with 1,024 filenames or 64KB of filename data), no 'paging' of the directory is possible, so 1,024 filenames per folder is the 'hard limit' here. See the SIDE3 loader manual for more information.

Searching for Files

Rather than paging through a long list of filenames, the loader allows rapid searching for a partial filename match simply by typing characters, and in version 3.0 of the loader, this facility has been greatly enhanced to allow recursive searching of directory trees of virtually unlimited size.

As an example: to locate all files and folders whose names include the word 'TUR', simply type the phrase and the search will begin automatically. To narrow or broaden the search, just type additional characters or shorten the string with the Delete/Backspace key. Every time the search string changes, the search begins again, looking for the new string.

While the list of search results is being populated, you may move through the list and perform operations on filenames just as you would in a normal directory listing. It is therefore not necessary to wait for a search to complete before launching the file of interest as soon as it appears.

Pressing the Break key while a search is underway will terminate the search but leave the search results on the screen. Pressing Esc will exit search mode completely and return to the directory which was logged prior to search mode being invoked. You may also exit search mode by pressing the Backspace key on an empty search string.

Note that the search is conducted on the currently logged folder and all folders below it in the directory tree. To search from the root directory of a volume, you must first log the root folder.

Occasionally, one may wish to enter characters into the search string which are normally used as navigation keystrokes (for example, the '-' or '_' characters). To enter such characters in the search string, simple press CTRL+ESC first.

To completely clear the search string without actually leaving search mode, press CTRL+CLEAR or SHIFT+CLEAR.

In the list of search results, as well as launching and mounting files, one may open a file's containing folder by highlighting the file and pressing CTRL+F.

Note that extremely complex and populous directory trees and/or a large number of matching files may cause the loader's buffer to become full. The maximum number of filenames which may be displayed at any one time as still 250 in the SIDE/XEL Loader, and 1,024 in the SIDE3 Loader. Note that there is no facility to 'page' through a longer list of search results in the SIDE/XEL Loader,

despite directory paging being possible. Likewise, 1,024 search results is an absolute limit in the SIDE3 Loader.

Despite the limitation on the number of matching entries which may be simultaneously displayed (250 for the SIDE Loader and 1,024 in the SIDE3 Loader), the search facility is extremely powerful. In tests, the loader was able to scan a 970MB FAT volume containing over 17,000 files and some 5,400 folders in under four minutes. Since the currently logged directory is always scanned in its entirely before any sub-directories are searched, the search facility is also a convenient means of quickly locating files in the current folder.

APT Partition View

The launcher's other 'view' is 'APT Partition View', and you'll see this when opening the APT container via the partition menu. After opening the APT, you'll see a list of all Atari partitions complete with their current drive assignments, partition names (where present), partition IDs, partition types, and sizes.

Mounting, tagging and launching work just as they do when XEX and ATR files are being displayed.

In the partition type column, you'll see 'DOS' or 'Ext' since these are the only two APT partition types catalogued by the loader. 'DOS' denotes a standard partition, while 'Ext' denotes an 'Extended' APT partition: namely one which is mapped directly to an MBR FAT partition located outside of the APT area. External entries (which are described in the APT partitioning manual) may be created via the APT FDISK partitioning tool, and they provide an excellent method of sharing data between PC operating systems (such as Windows and Linux) which recognise and mount MBR partitions, and Atari disk operating systems (such as SpartaDOS X) which can read FAT partitions. External entries also provide an essential means of assigning drive numbers to MBR FAT partitions at the partitioning stage (since the Atari – unlike a PC – needs to store persistent drive assignments on the disk).

At the extreme right, the logical partition size is shown. Note that this size does not necessarily represent the physical size of the partition on disk. Rather, it is the addressable size of the partition. Although a 65,535-sector partition occupies 32MB on disk, if the logical sector size of the partition is 256 bytes, the size will be shown as 16MB.

Mounts Menu

The Mounts menu displays a two-page list of logical drive assignments (for drives 1 through 15) in the SIDE/XEL loader:

Drives may have one of three states: SIO, HDD, or ATR. 'SIO' simply means the drive number refers to a device on the SIO chain (regardless of whether any such device actually exists on that drive number). 'HDD' means the drive number refers to an APT hard disk partition, and 'ATR' means the drive is mapped to a mounted disk image.

APT entries (partitions) have their partition ID and (optionally) partition name displayed next to them. The partition ID is the same one displayed alongside each partition entry in the launcher menu. ATR entries, meanwhile, have their host partition ID displayed alongside, followed by the name of the mounted ATR (if said filename has been picked up during directory scans). The host partition ID corresponds to the ID displayed on the Partition menu next to the FAT which contains the disk image file. Therefore, several ATRs may share the same host partition ID.

ATR and HDD names are only displayed if they happen to have been logged on the launcher menu (by logging the ATR's containing folder, opening the APT, or running a recursive search which happens to match the mounted ATR). This 'lazy' approach to populating names in the drives menu is preferred to recursively scanning every folder in every MBR partition every time the loader is launched with ATRs already mounted.

New to version 3.0 is the ability to perform a reverse lookup on a drive slot. Pressing CTRL+F on a drive slot will now display the highlighted ATR/partition in its containing folder/APT. In the case of an ATR, finding the containing folder may involve a scan of the entire folder tree, although if the

corresponding ATR exists in the currently logged directory or search results, location of the file will be instantaneous.

This feature can be rather useful if the name of the ATR or partition has not populated the slot of interest on the Drives menu. In that case, pressing Return on the entry will take you directly to the image mounted on that drive number, simultaneously populating the name field on the Drives menu in the process.

Pressing RETURN on an entry still unmounts the drive as in prior versions of the loader.

The SIDE3 loader shows the currently mounted cartridge image as well as logical drive assignments:

As with logical drives, pressing RETURN will unmount the highlighted cartridge image.

Tools Menu

The Tools menu has three items: 'Unmount ATRs', 'ATR swap', and 'Disk refresh'. 'Unmount ATRs' simply re-reads the partition table from disk, unmounting all ATRs and restoring the partition map to the state it was in when the computer was first booted (or the last time changes were written to the partition table by the FDISK tool, or the last time the partition table was refreshed). You may access 'Unmount ATRs' from any of the loader menus by pressing CTRL+U. 'Unmount ATRs' duplicates the functionality provided by SHIFT+RESET when the 'HDD refresh' option is enabled in the main BIOS setup menu.

The 'ATR swap' option rotates (swaps) all mounted disk images by one place. For example, if ATRs are mounted on drives 1, 2 and 3, rotating or swapping the ATRs will move drive 1's ATR to drive 3,

drive 2's ATR to drive 1, and drive 3's ATR to drive 2). This is exactly equivalent to rotating ATRs by pressing the ATR swap button on the SIDE cartridge. You can rotate ATRs in this manner anywhere in the loader by pressing CTRL+S.

The last item – 'Disk refresh' – was introduced primarily to support the XEL-CF adapter designed by MytekControls for the MPBI interface of the 1088XEL motherboard. Since the XEL-CF lacks a card removal sense register, swapping a CF card requires a card reset to be explicitly issued following card reinsertion. Selecting 'Disk refresh' (you can also press CTRL+D anywhere in the loader) will issue a card reset, and refresh the partition table and FAT/APT catalogues. This facility should also allow card hot-swapping with the original SIDE cartridge (which lacked a card removal sensor), although this has not been tested.

Options Menu

The *Options* menu provides a number of user configurable settings:

The SIDE3 loader presents many more settings:

The settings are described below (please see the SIDE3 loader manual for a description of SIDE3 settings):

Option	Purpose
BASIC	Toggle the presence of internal BASIC (and the external SIDE cartridge ROM where available) when launching XEX or ATR files. *Note: this setting will be dimmed on an Incognito 800 running in Colleen mode, since no internal BASIC exists. Moreover, the integral BASIC Cartridge provided in Colleen mode cannot be disabled in software outside of the setup menu, so to run the majority of XEX files in Colleen mode, you should first DISABLE the BASIC Cartridge in the BIOS setup menu. Conversely, to run a BASIC program from the Incognito loader in Colleen mode, ensure that the 'BASIC Cartridge' setting in the BIOS setup menu is set to 'Enabled'.*
FAT FMS	Toggle the presence of the read-only FAT CIO file system handler
Show size	Toggle display of XEX/ATR and logical volume sizes
Joystick	Change the port for joystick control or disable it entirely
Colour	Select one of sixteen colour schemes
Sound	Enable or disable the key click sound
Screensaver	Toggle the five-minute screen timeout
Accelerate	Toggle up/down cursor auto-repeat acceleration

Note that unlike the main BIOS menu, settings are immediately 'remembered' in the loader (without any need to explicitly 'save' settings).

Info Menu

The *Info* menu displays the loader version number and copyright notice.

Exit Menu

The *Exit* menu contains one or two items depending on the platform. In all versions of the loader, you can reboot the machine by pressing CTRL+R anywhere in the loader. You would typically restart after manually mounting one or more disk images. Note that the current BASIC/Cart setting dictates the presence or absence of both internal BASIC and any external SIDE cartridge ROM (if available) after the machine is rebooted.

The Ultimate 1MB's built-in SIDE loader also permits restarting the machine with SDX enabled. You may access this feature with CTRL+X. When booting into SDX, only the state of any active external SIDE cartridge ROM will be affected by the current BASIC/Cart setting (since the state of internal BASIC is managed by SDX itself).

Image and Partition Mounting

Several methods exist for mounting disk images and APT partitions. Pressing CTRL+SPACE (or moving the joystick to the right with the trigger held down) assigns a partition or ATR then next logical drive number (or at least the drive number the loader thinks is the most logical suggestion), displays the drive number next to the filename, and moves the selection bar to the next entry. Pressing CTRL+SPACE on a partition or ATR already mounted unmounts it, removing the drive number.

To manually specify the drive number, press the Tab key on a filename, or move the joystick left with the trigger held down. An editable drive number then appears next to the ATR or partition name. Increase or decrease the drive number using the cursor keys, and press Return, Tab or the joystick trigger to accept the drive number and mount the volume. You can press Escape to abandon the process. Selecting the 'Off' entry in the drive number list unmounts the volume.

Volume mounts are immediately reflected in the Drives menu, should you require a quick visual cue of which drive numbers are assigned and which are free.

Map Files and Disk Flipping

Sometimes it's necessary to mount multi-disk ATR sets and tagging each file individually can be a tedious process. 'Map' files make this easier by automating the mounting procedure. A map file is simply a plain DOS/Windows/Unix text file containing EOL terminated lines comprising a drive number, optional path and filename:

```
Dn:[path]filename.ext
```

'Dn:' represents the target drive number on which the image will be mounted, and the path/filename of the image file should immediately follow on the same line. For example, you might create a file called *Ballyhoo.map* and add the following two lines:

```
D1:Ballyhoo (s1).atr
D2:Ballyhoo (s2).atr
```

When you press Return on *Ballyhoo.map*, the loader will mount the two ATRs on the specified drives and – if successful – reboot the Atari. In this case, the ATRs must reside in the same folder as the map file. If the ATR files reside in a different folder to the map file, include the path:

```
D1:\ATRs\Demos\Sweet Illusions (s1).atr
D2:\ATRs\Demos\Sweet Illusions (s2).atr
```

Either the right wedge (>) or backslash (\) characters may be used as path delimiters, and the path will be relative from the currently logged folder unless the path begins with a path delimiter (which implies and absolute path starting from the root directory).

Once the ATRs are mounted, the Atari will boot from drive 1. In the case of *Ballyhoo (s1).atr* and *Ballyhoo (s2).*atr, when the software requests its second disk, pressing the SIDE/ATR swap button prior to accepting acknowledging the disk swap request will move *Ballyhoo (s2).atr* from D2: to D1:,

and Ballyhoo (s1).atr from D1: to D2:. The SIDE ATR swap button *rotates* the entire ATR set down by one place in the drive mappings, and wraps around the ATR on the lowest drive number to the vacated highest slot.

Take a set of three ATRs as an example. When mounted, the assignments are thus:

```
D1:Image1.atr
D5:Image2.atr
D6:Image3.atr
```

After the first disk rotation, the mount points are as follows:

```
D1:Image2.atr
D5:Image3.atr
D6:Image1.atr
```

A second disk rotation results in:

```
D1:Image3.atr
D5:Image1.atr
D6:Image2.atr
```

As can be inferred, once the number of rotations equals the number of mounted ATRs, the assignments will return to their original positions. Hard disk partitions (which may be used freely with mounted ATRs) are unaffected by the rotation process and remain fixed.

Image rotation can also be accomplished from the loader itself via the 'Swap ATRs' option on the 'Disk' menu.

Should you wish to execute a MAP file without an instant reboot, press CTRL+RETURN instead. The volumes will be mounted but no further action will be taken.

BASIC/SDX Enable Metatags

Executables and ATRs which require internal BASIC to be enabled on a 'one off' basis (assuming the loader is set to disable BASIC during normal use) may specify their dependency on BASIC by including the *[BASIC]* metadata tag anywhere in the FAT filename. For example, the following ATR will always enable internal BASIC upon booting (if launched via the Return key):

```
BASIC Games [BASIC].atr
```

Note that the BASIC metatag on an ATR file is not observed if the volume is mounted via tagging (using CTRL+SPACE) or via the TAB key.

U1MB builds of the SIDE loader also recognize the 'SDX' metatag, which causes subsequent reboots to be performed with SDX active. This is useful if you wish to mount an ATR or partition which relies on the presence of SpartaDOS X.

CF/SD Card Hot-Swapping
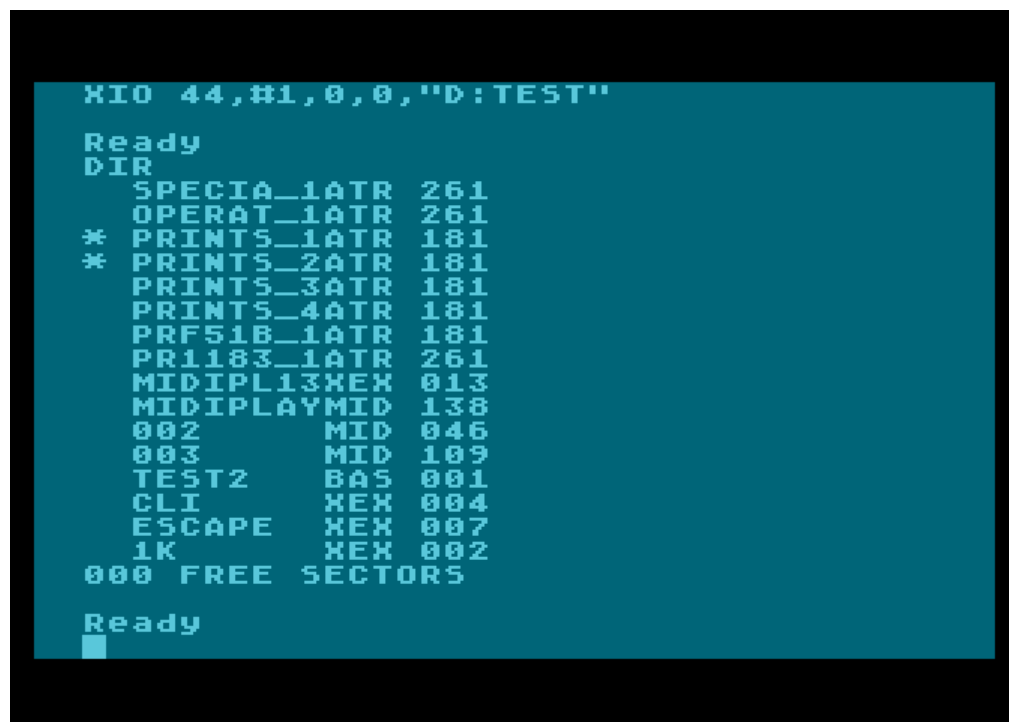
If you use a SIDE2 or SIDE3 cartridge (not Incognito), you can hot-swap CF/SD cards while in the loader. All ATR mounts will be removed following a card swap. Owners of the original SIDE cartridge or the XEL-CF adapter may swap cards by forcing a 'Refresh disks' operation (see the 'Disk' menu section earlier in this document; 1088XEL owners should also consult the 1088XEL section).

The Built-In CIO FAT File System Handler (Mini DOS)

As mentioned, when 'File system' is enabled in the Options menu, a small, read-only CIO file system handler is installed in low RAM when launching executables or BASIC programs. The file management system permits read-only access (via 'D1:') to the entire FAT partition from which the program was launched. Note that while long filenames are not supported, the short (8+3) aliases of long filenames will be displayed and are accessible.

The screenshot below shows the default directory being changed in Altirra BASIC (via the XIO 44), and a directory listing obtained with the 'DIR' command.

As of version 3.0 of the loader, subdirectory support allows navigation of the entire FAT folder tree via standard CIO commands compatible with SpartaDOS X. However, since the loader's file system driver does not identify itself as SpartaDOS at $700/701 (in the loader, locations $700-$702 contain 'L' followed by the major and minor revision number), software wishing to take advantage of the subdirectory-related commands may wish to explicitly check that it's running under loader version 3.0 or above.

As with prior loader versions, the launched application's containing FAT folder is the default working directory, and any DOS 2.x-style programs will consider this folder a flat file system. However, by issuing the XIO 48 'Get Current Path' command, an application may deduce where in the folder tree the default working directory resides. XIO 44 (Change working directory) is also supported, and folders appear in directory listings with a colon (':') immediately preceding their name.

Even applications which don't support or recognize subdirectories may access them if file input dialogues offer sufficient space for the entry of relative and absolute paths. Likewise, listings of

directories other than the current working directory may be obtained if an application does not impose limitations on the format and length of a directory search mask.

The FMS supports all SpartaDOS X compatible path delimiters. Some examples:

```
D:>SUBDIR>*.BAS
D1:\SUBDIR\SUBDIR2
D:..\
D1:<SUBDIR\TEST
```

The left-pointing wedge is functionally identical to the '..\' parent directory notation, while backslashes or right-pointing wedges may be freely used as path delimiters. It should be noted that since only drive 1 is supported, the device ID may be 'D1:' or 'D:' (the latter implying the 'default logged drive', which in this case is always drive 1).

The CIO handler supports OPEN, CLOSE, GET and STATUS and XIO commands. PUT, NOTE and POINT are not supported and will return errors, as will any attempt to open a file for write. As of loader version 3.0, formatted directories (accessed by opening with 6 in ICAUX1) express the sector count (i.e. file size) as multiples of 512 bytes (since the FAT sectors are 512 bytes long) rather than the simulated 128-byte sector count employed in previous versions. A 1KB file will now be shown to occupy 2 sectors, for instance, rather than 8. ASCII characters in directory listings which cannot be expressed using the Atari character set are represented as underscores. When attempting to open such files, underscores should appear in the same positions in the file specification.

The limit of sixty-four files on DOS 2.x directories is not enforced, so a FAT directory comprising hundreds of files will be listed in its entirety. Where this (or indeed the existence of folder names in directories) would cause compatibility issues for programs designed to run with DOS 2.x, it's advisable to launch said programs from a subdirectory containing a) no child subdirectories, and b) no more than sixty-four files.

CIO Commands for Directory Handling

As well as basic commands for open, get, status, etc, the FMS driver supports CIO commands 44 (CHDIR) and 48 (Get Current Directory Path).

Get Current Directory Path

```
iccom = 48
icbal = low byte of 'D:[path]' address
icbah = high byte of 'D:[path]' address
icbll = low byte of buffer address
icblh = high byte of buffer address
```

This command returns the path to the current working directory of the specified drive. If a path is also specified, the returned path will represent the target of the supplied path relative to the current working path. For example, if the current working path is '>FOLDER1>TEST' and command 48 is issued with the path 'D:<', the buffer will return '>FOLDER1' (i.e. the name of the parent).

Change Current Directory (CHDIR/CWD/CD)

```
iccom = 44
icbal = low byte of 'D:path' address
icbah = high byte of 'D:path' address
```

The CD function may be accessed via BASIC thus:

```
XIO 44,#IOCB,0,0,"D:path"
```

Using UFLASH with the built-in FMS

The latest version of UFLASH (supplied with the firmware) takes advantage of the FMS's subdirectory support when the application is launched from the XEX loader. The same directory tree navigation functions available with SpartaDOS X are present in the file selector when accessing a FAT volume.

It is therefore no longer necessary to keep UFLASH in the same folder as the ROM images when updating the firmware directly from FAT.

Booting Custom ROMs

Thanks to Eric Bacher (ebiguy), patched versions of all four OSS language carts (MAC/65, BASIC XL, BASIC XE and Action!) may now be flashed to the SIDE/SIDE2 cartridge. A combo ROM containing all four OSS ROM as well as the SIDE loader is available in the firmware archive, along with a tool (SIDECFG) which allows selection of the active ROM. Once set, the SIDE loader will silently cause the selected OSS ROM to boot (instead of the loader) until the active ROM is changed, the loader is re-selected, or until START+SELECT+OPTION are held down while starting the cartridge. Holding down START+SELECT+OPTION boots the loader and causes it to become the default ROM until SIDECFG is again used to select one of the OSS ROMs.

It is hoped that this facility will be expanded in the future as more cartridge conversions become available for SIDE/SIDE2. Since 256KB of ROM space is available (SDX occupying the lower half of the ROM), several patched cartridges could occupy the external cartridge space.

Error Codes

Error codes returned by the loader correspond to those issued by Atari DOS and SpartaDOS X.

128	$80	**User break abort**
		The user pressed the BREAK key during an IO operation.

130	$82	**Non-existent device**
		The device name supplied does not exist. Since the loader's FMS supports only one drive, this error will be returned when any drive specifier other than 'D:' or 'D1:' is used. In the context of MAP files, this error will be returned when an invalid mount point is provided (i.e. a drive outside of the range 'D1:'-'DO:').

133	$85	**File not open**
		An attempt was made to read from a file which has not first been opened.

136	$88	**End of file**
		The end of the file was reached.

139	$8B	**Device NAK**
		The device failed to respond. This error may be caused by an attempt to read a Compact Flash card which has been removed or is failing.

142	$8E	**SIO overrun**
		This error indicates that the IO is out of sync with the Compact Flash card, usually as a result of system instability preventing reliable communication between the computer and the storage device. Use a different card or address system instability.

146	$92	**No function in device handler**
		An attempt was made to perform an operation not supported by the loader's FMS driver, such as opening a file for read, attempting to manipulate the file pointer, etc.

150	$96	**Path not found**
		The specified directory path does not exist. This error may be returned by the FMS driver (if attempting to access a non-existent directory) or when the path in a MAP file references a non-existent location.

161	$A1	**Too many channels open**
		The loader's FMS supports only one open file at any time. Attempting to open a second file without closing the first will cause this error.

165	$A5	**Bad filename**
		The filename contains illegal characters or is otherwise malformed. While MAP files may contain long filename entries in the path and target file specification, filenames passed to the FAT FMS driver should be in standard '8+3' format.

170	$AA	**File not found**
		An attempt was made to access a file which does not exist.

Limitations

Maximum number of files and folders

The SIDE/XEL Loaders can display a maximum of 250 entries (files and folders) in a given folder at one time, although directories of unlimited length may be displayed by paging through the directory 250 items at a time via the '…' symbol or the CTRL+M shortcut.

The list of search results may not exceed 250 items. If many lengthy filenames are indexed, this limit may be reduced if the buffer is filled up (filenames may be up to 128 characters in length). The FMS driver (which allows CIO access to FAT volumes), meanwhile, imposes no restrictions on the number of files per directory.

The SIDE3 Loader is capable of displaying up to 1024 files and folders, or 64KB of directory data (whichever limit is reached first). Paged directories have been removed in the SIDE3 loader for this reason, and if you encounter directories which will not fit in the SIDE3 loader's buffer, you should reorganise the FAT volume to use smaller directories.

Recursive search results are likewise limited to 1024 items or 64K of directory data, whichever is encountered first.

With all loaders, recursive searches (starting at the currently logged folder) may scan a virtually unlimited number of folders, nested to any depth.

MEMLO

The XEX loader resides between 0x0700 and 0x09FF (768 bytes), and an executable file must not overwrite this memory area. The lowest load address supported for an executable file is therefore 0x0A00. If an XEX is to use the built-in FAT FMS, it should not load below 0x1480, and a standard 'safe' load address (0x1F00 or 0x2000) is highly recommended in case of future FMS expansion.

Differences between the original and new firmware

Aside from what has already been discussed, there are some further key differences between the underlying behaviour of the new BIOS and loader and that of the original.

- 'One-shot' configuration changes are no longer possible. In the original BIOS, unsaved changes were written to the hardware registers and persisted through the next system restart until the next time the BIOS Setup Utility was invoked. This is no longer possible. Any unsaved setting is immediately lost when exiting setup.

- The POSIX standard (Monday = Day 1) is no longer used when calculating the system clock's day of the week. A method compatible with SpartaDOS X is used instead (Sunday = Day 1).

- The new BIOS uses addresses 0x20-2F, 0x40-4E and 0x60-6E of the Ultimate 1MB's SD1305's NVRAM. SpartaDOS X uses bytes 0x30-31, so the entire 0x3x address range is deliberately avoided by the BIOS. The Incognito BIOS uses slightly fewer bytes, but 0x20-2F, 0x40-4F and 0x60-6F should now be considered reserved for Ultimate and Incognito. While it's unlikely that NVRAM usage will significantly increase in future BIOS revisions, it would be unwise to make assumptions when designing your own drivers or any other software which makes use of the NVRAM. It would probably be wise, if in doubt, to use the 0x70-7F range, since 0x50-0x5F is used by the XEX loader.

- The new BIOS knows whether the Ultimate 1MB is configured for XEGS mode or XL/XE mode, and dims the XEGS ROM slot menu item if the host machine is deemed not to be an XEGS.

- The main BIOS ROM signature and versioning information is somewhat enhanced compared to the original, and is completely incompatible with it.

- The new XEX loader does not clear the mount table on entry, so previously mounted ATRs will remain mounted and visible when re-entering the loader. To restore the partition table to its original state, select 'Undo mounts' on the loader's 'Disk' menu.

- The new PBI BIOS fixes several bugs in the version designed to work with the original main BIOS, and it is not compatible with Candle's original BIOS. Likewise, older versions of the PBI BIOS should not be used with the new main BIOS and XEX loader.

- The new PBI BIOS uses DUNUSE (0x0307) for bits 16-23 of the sector address when bit 7 of DDEVIC (0x0300) is clear. Use of DUNUSE was reinstated to ensure compatibility with the IDE Plus 2.0 BIOS, and to support versions of the SDX FATFS.SYS driver which relied on it (although as it happens, the newest version of FATFS.SYS dropped reliance on DUNUSE, using the XDCB instead).

- The new U1MB BIOS includes a copy of the OS character set at the usual address (0xE000). This enables the Turbo Freezer's menus to display properly if the Freezer is invoked from *inside* the BIOS setup menu. In addition, the BIOS setup menu's NMI handler vectors DLIs and VBIs through VDSLST (0x0200) and VVBLKI (0x222) respectively. The Incognito BIOS lacks the standard OS character set but uses the same interrupt vectors.

- The new BIOS – wherever possible – establishes via software the presence of attached hardware (VBXE, Stereo Pokey, Soundboard, etc) and disables software control of said devices of they are not present or not attached to the U1MB's IO signals (one exception is Covox, which cannot be detected in software). This methodology has already uncovered previously undiagnosed hardware issues on a few machines.

- The new BIOS has migrated all VBXE, SIDE, Stereo, Covox and Soundboard configuration functionality out to the default plugin module. As a result, support for these devices is entirely optional and may be completely removed and replaced by support for other hardware by means of a different plugin module.

- The new PBI BIOS no longer automatically 'registers' as a host partition the first FAT it finds on the disk. This was done in earlier versions as a convenience for the original loader (which did not register host partitions at all), but has been dropped since the new loader explicitly handles all host partition registration.

- The most recent U1MB firmware optionally allows use of the SIDE cartridge's 'external' cartridge ROM at the expense of 'ATR swap button' functionality. This allows repurposing of 256K of flash ROM for use as a bank-switched application cartridge alongside the PBI hard disk and PBI-hosted disk images.

Incognito Features

Although broadly similar to the U1MB firmware, the Incognito firmware differs in a few important respects. Most significantly, the Incognito offers two modes of operation: XL/XE mode and 'Coleen' (800 compatibility) mode. Colleen mode provides some useful enhancements over the base Atari 800 hardware, too, including a built-in AXLON 1MB RAM expansion and a built-in 8K BASIC cartridge ROM.

Colleen Mode Settings

To switch between XL/XE and Colleen modes, simply press RETURN on the 'Hardware' setting.

Each of the four configuration profiles can reflect either hardware type, so you could – for example – have two readily available Colleen configurations and two XL/XE configurations and switch between them easily.

Hardware Allows selection of Colleen or XL/XE hardware modes. XL/XE mode is almost identical to an U1MB machine, with the exception that there is only one BASIC ROM and no possibility of running the machine in XEGS mode (thus there are no XEGS game ROMs either). Colleen mode presents an Atari 800 with an integral (and optional) 1MB AXLON memory upgrade, and a 8KB BASIC cartridge which may also be toggled (and flashed with any 8K ROM content).

System RAM   Use this setting to select the amount of base RAM in the machine: 16K, 32K, 48K or 52K (the latter providing RAM at $C000-$CFFF).

OS ROM   Select from one of four user-definable 10K 400/800 operating system ROMs. The default selection includes PAL and NTSC versions of OS A and OS B.

BASIC cartridge   Allows the user of an integral 8K cartridge ROM, commonly occupied by Atari BASIC. Note that the 8K BASIC cartridge in Colleen mode does NOT work like internal BASIC on an XL/XE. There is no BASIC enable/disable bit in the PORTB register (since PORTB is used for the controller ports); the 8K ROM works just like an invisible cartridge plugged into the left cartridge port. For this reason, the SIDE Loader will be UNABLE to automatically suppress BASIC in Colleen mode if BASIC is enabled in the setup menu. To run most programs, you will first need to manually disable the internal BASIC cartridge ROM in the BIOS setup menu.

1024K AXLON RAM   Use this setting to enable or disable the built-in 1MB of AXLON extended RAM. See the AXLON documentation for information on the AXLON RAM banking scheme.

System Clock and Features

In Colleen mode, the 'System Clock and Features' menu has only two options, for the system date and time. In XL/XE mode, however – since the PORTB register is used for ROM and extended RAM control – there is also a setting to enable controller ports 3 and 4:

When this option is enabled, the third and fourth controllers may be read in the following manner:

- Configure PORTB as an input, but using the registers at $D341 and D343 instead of PORTB and PBCTL
- Read joysticks 3 and 4 at $D341
- Read joystick triggers 3 and 4 at $D033

It should be noted (and apparent) that the feature was never intended to allow games which support four joysticks to be run without modification in XL/XE mode.

The PBI BIOS and Hard Disk menu is very different in Colleen mode, having only a single setting: Hard Disk.

Once the hard disk enabled in Colleen mode, booting SpartaDOS X will cause the COLLEEN.SYS driver to load and provide access to the hard disk. Unfortunately – since no PBI support exists in Colleen mode – hard disk partitions are not bootable or accessible from any other DOS.

Plugin Support

As of firmware version 4.0, the Incognito gains the plugin support already seen on the U1MB. However, plugin support requires a CPLD firmware (JED) update to enable use of the self-test ROM when the configuration is unlocked.

Plugins – once active – will generally display their settings in the 'Device Control' menu, although you may also see additional settings appearing in other menus. With the CPLD update applied, the 'EXTSEL mode' setting will also become active (it is present but unavailable with the older JED).

Incognito JED Update

The Incognito JED update provides the following fixes:

- Un-mapping of original TRIG3 register without implementing cartridge sense functionality in XL/XE mode fixed
- Plugin support (i.e. self-test ROM mapping when configuration is unlocked) added
- Incorrect handling of CASINH signal (required by VBXE) corrected

- Different EXTSEL behaviour in Colleen and XL/XE modes corrected (fixes slot 3 cards)
- VBXE support added

Once the CPLD update is applied and the version 4.0 firmware flashed, plugins will become active and the 'EXTSEL mode' setting will become active (in the 'BIOS Settings' menu), the available options being 'Colleen' and 'PBI' mode.

1088XEL Features

The special build of the U1MB firmware for the 1088XEL includes some features unique to that platform. Although most differences are confined to the 1088XEL plugin module (which defines usage of the four IO signals on the U1MB), a few changes to the main BIOS, PBI BIOS and XEX loader were also necessary. Most notably, the 1088XEL version of the PBI BIOS is designed not to support the SIDE cartridge but the XEL-CF module. Since the XEL-CF can house up to two CF cards in a master/slave configuration, the PBI BIOS and XEX loader can access them both.

Note: it's possible to drive a SIDE/SIDE2 cartridge via the internal U1MB firmware on a 1088XEL board. To do so, simply flash the standard U1MB/SIDE PBI BIOS and XEX Loader modules to the appropriate slots and replace the 1088XEL/XEL-CF main BIOS plugin with the 1088XEL/SIDE plugin. The SIDE plugin removes slave drive support and reinstates the SIDE cartridge's ATR swap button functionality.

When turning on the 1088XEL for the first time, you'll see a special boot logo which stays visible for around three seconds. This logo appears irrespective of whether or not the "splash screen" is enabled, and it allows the TK-II keyboard interface to initialise so that the keyboard is responsive to hotkeys (for BIOS menu entry, disabling internal BASIC, etc) at the appropriate time.
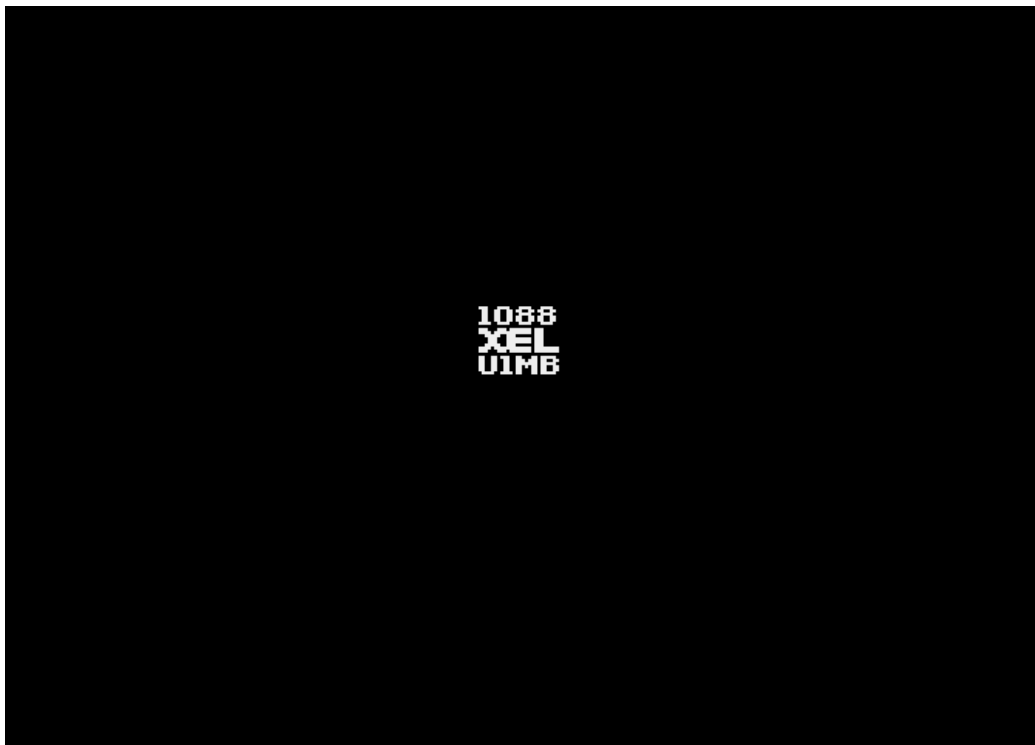
Inside the BIOS setup menu, you'll see several items specific to the 1088XEL BIOS plugin, which we'll cover here.

System Clock and Features

Stereo POKEY

Turns the second POKEY on or off.

2nd POKEY IRQ

Turns the second POKEY IRQ on or off. Will be disabled and greyed out should there be no second POKEY present.

GTIA V-Gate

Controls the V-Gate function (crops horizontal scan). V-Gate, when enabled, also crops the Sophia display (subject to Sophia firmware support).

PBI BIOS and Hard Disk

Slave Drive

Turns XEL-CF slave drive polling on or off. If you're using a dual-card adapter with the XEL-CF interface and have both card slots populated, turn this option on. Otherwise (if you're only using the master card slot), turn it off to prevent delays caused by the system polling the second drive and timing out in the process. Note that the XEX loader has its own Slave drive option which will also need enabling to facilitate cataloguing of the second card. However, if you simply want to use two cards with standard HDD partitions, enabling Slave support in the PBI BIOS is sufficient.

MPBI Device Control

Since U1MB's auxiliary IO header is no longer accessible in the 1088XEL, and since three of the signals are already allocated to internal devices, only one unassigned signal remains (M1), and since M1 appears on the 1088XEL's MPBI connector, the 'Device Control' menu is entitled 'MPBI Device Control' in the 1088XEL BIOS's default plugin. Of course, this may be changed – along with the naming and functionality of the M1 control option – via a custom BIOS plugin. One such plugin, designed to control the Rapidus core via the M1 signal – is included in the firmware distribution package.

XEX Loader

Slave Drive

 The 1088XEL/XLD loader differs from the standard U1MB loader in one key respect: namely that it (optionally) supports two physical disks when a suitable dual Compact Flash adapter is used with the XEL-CF interface.

When the PBI hard disk is activated in the main BIOS settings, the loader will inherit the main BIOS slave drive setting (the loader's own slave setting will be greyed out). When no PBI hard disk is active (or when the PBI BIOS is completely disabled), the slave disk may be independently toggled on and off in the loader's options menu:

If enabled, the second disk will be polled and the attached media displayed in the 'Device' menu, where one may switch between the two physical disks by pressing Return on the desired entry. If only one FAT partition is present on the chosen physical disk, it will immediately be opened in the Launcher menu. Otherwise – if more than one FAT partition exists – the partition list will appear,

allowing selection of the desired FAT (or APT). If only a single FAT exists, one may cursor back from the launcher menu to the partition menu if one wishes to open the APT.

As with the standard loader, the last accessed partition is remembered, although in this instance it's the last accessed partition on the last accessed physical disk which will persistently be opened when the loader is first launched (until such time as the loader sees that the media has changed since a partition was last selected). If 'Recall folder' is enabled, the last accessed FAT folder will also become the default.

Media Hot-Swapping

Although the XEL-CF has no card removal sensor, one may still swap physical disks (CF cards) by invoking the 'Refresh disks' function on the 'Disk' menu.

This will reset all controllers, re-poll the physical disk(s) and display the Launcher menu or the Device menu, depending on whether the same card was re-inserted.

Frequently Asked Questions

Since the alt BIOS was released, I've compiled a list of the most common questions asked by users.

Q: I've upgraded to the new firmware and all my settings are gone. What happened?

A: The new BIOS's configuration records in NVRAM are incompatible with those of the original firmware, so your original configuration became invalid when you upgraded. Please manage your new settings and set the system clock using the BIOS setup menu.

Q: I've upgraded to the new firmware and the PBI hard disk doesn't work anymore.

A: Check that you have enabled the PBI BIOS and enabled the Hard Disk in the 'PBI BIOS and Hard Disk' menu.

Q: There's a splash screen when my computer starts up.

A: If you don't want to see the boot logo, disable it in the BIOS Settings menu, or press Space or Escape to skip it. You can also configure the logo to appear only after a cold power-up.

Q: When I press the left and right arrow keys, instead of scrolling through the available settings for the selected item, a different menu is displayed. How do I change a setting?

A: Simply press the Return key to toggle an 'Enable/disable' setting, or to move the highlight into a list of settings. Use the up and down cursor keys to select the desired setting from a list, and press Return again when done. Pressing Escape after an edit cancels the change.

Q: How do I save my settings?

A: In the BIOS setup menu, press the 'S' key. All available options for saving settings and restarting the computer can be found in the 'Save and Exit' menu. In the loader, settings changes are saved automatically as soon as you make them.

Q: The XEX loader won't mount ATR disk images anymore.

A: Ensure you have the PBI BIOS *and* Hard Disk enabled in the 'PBI BIOS and Hard Disk' menu.

Q: XEX programs are crashing when run from the XEX loader.

A: Ensure BASIC is not enabled for those programs which require it to be disabled.

Q: My XEX programs fail to load or crash when I start them from the SIDE loader. They worked with the old SIDE Loader? Is the new loader buggy?

A: The new SIDE loader loads executable files 2-3 times more quickly than the original SIDE Loader. It does this by burst-reading sectors into RAM without continually polling the IDE status register's data-ready flag after reading every single byte. As a result, systems which are already prone to bus instability may exhibit read errors with the new loader when they appeared to work well with the old one. The issue is squarely a hardware one, however, and the new loader is designed to work with stable systems. The solution is to a) Fix the instability issue, or b) replace your Transcend, Kingston 'Flower', or no-name CF card with something known to work with the hardware (preferably a SanDisk card).

Q: Stereo Pokey is greyed out in the BIOS menu and I can't turn it on or off. I have it connected to U-Switch and it worked with the old BIOS. What's up with the new BIOS?

A: There's nothing wrong with the new BIOS. Your U-Switch never worked in the first place since it's wired up wrongly or there's some other hardware problem. The original BIOS didn't care whether there was in fact a Stereo Pokey board present, let alone whether it could actually be toggled in software. The new BIOS ascertains whether the hardware is present and whether it can be deactivated in software.

Q: The BIOS menu and/or XEX loader have corrupted displays. Everything was OK with the old BIOS. What's going on?

A: The BIOS menu and loader employ numerous non maskable interrupts (NMIs) in order to generate the three-tone display and various other graphical effects. Corrupted graphics point to system instability which was simply not evidenced by the original firmware. Get your U1MB installation checked over for poor connectivity and noisy reset line issues.

Q: I've attached a cartridge to my Atari but the machine is behaving as if the cartridge isn't there. What's wrong?

A: If you have the PBI BIOS enabled, you must set the 'Hard Disk' to disabled when you're not using a SIDE cartridge, otherwise external cartridges may be suppressed. If high speed SIO is not required, one may also disable the PBI BIOS entirely, although HSIO is designed to work with external cartridges and may usually be left enabled. An exception is the 1088XEL when used with the XEL-CF HDD where – since the HDD is not cartridge based – external cartridges function perfectly well with the internal hard disk.

Q: After upgrading to U1MB firmware 3.0, the SIDE loader boots all the time no matter what I do?

A: You have a SIDE cartridge attached to the machine and the 'SIDE Cart ROM' setting is set to 'Enabled'. Set to 'Disabled' to stop the cartridge's SIDE loader from booting, or simply turn on the Hard Disk and 'ATR Swap Button'.

Q: After upgrading to U1MB firmware 3.0, I can no longer swap ATR disk images with the SIDE cartridge's reset button.

A: The U1MB's 'ATR Swap Button' setting is set to 'Disabled'. Turn it on to enable ATR swap functionality. Turning on the button will automatically suppress the external SIDE cartridge ROM or any other cartridge connected to the machine.

Implementation Notes

Although users interested in developing BIOS plugins or other software designed to interface with the U1MB/Incognito firmware should refer to the separate Technical Documentation, a precis of the general approach taken to the firmware implementation may also be of interest to programmers.

Compression

Since all three components of the firmware ('main BIOS', PBI BIOS and loader) are subject to comparatively severe constraints when it comes to code space, some interesting methods were used to fit as much functionality as possible into a limited ROM footprint. Needless to say, the inventiveness of the methods employed increased sharply with each successive firmware revision, as – naturally – features were added which weren't necessarily envisaged at the outset. While no code or data compression was possible with the main BIOS and PBI BIOS (since both execute directly from ROM and must not disturb user RAM), the loader was simultaneously the hungriest component in terms of code space and the prime candidate for compression, since it has full reign over the base 64K of system RAM. When the XEL-CF version of the loader required modification to drive dual Compact Flash adapters, Run-Length Encoding (RLE) was employed to compress all graphical resources (bitmaps, fonts, etc), and ultimately any data which contained discernibly repetitive information.

RLE compression is implemented on two different axes depending on characteristics of the data being compressed. Bitmaps are compressed on a column-by-column basis (using tools written in C and run on the PC), since it was observed that vertical runs of repeated bytes are more common than horizontal repetition. Other data (lookup tables, etc) is compressed in a more traditional linear manner. Ultimately, RLE compressed data is shrunk to as little as sixty per cent of its uncompressed size, and since the decompressor is trivial and concise, space savings are considerable. The decompressor itself is called via a table of RLE data block addresses, destination addresses, and the number of rows and columns where required.

Efficiency

While every possible measure was taken to maximise performance (code optimisation, loop unrolling in the PBI BIOS, etc) while remaining within the bounds of the firmware's tight memory footprint, some potentially time-consuming operations or representations of complex data required inventive solutions. An example is the loader building an alphabetically sorted list of long FAT filenames when cataloguing a directory. While the original SIDE loader built the file list and then sorted it after the fact, the new loader uses a binary search to place each newly inserted filename at the correct position as it is read from the directory. The file list itself is indexed by a table of pointers. Sorting of the list is thereby accomplished 'on the fly' without moving any filenames in memory. As a result, a directory comprising some 250 long FAT filenames can be read from disk and displayed on the screen in sorted order in less than two seconds.

Searching

While the loader's Google-style search facility is capable of recursively iterating through the entire directory tree, its method of doing so is a little unusual. While a breadth-first tree walker seems preferable owing to the fact it scans the currently logged directory in its entirety before scanning subordinate folders, the memory requirement for the FIFO list of directory cluster numbers is considerable and results in very large folder structures (of 4,000 folders or more) exhausting available RAM before the volume is fully scanned. A depth-first search, on the other hand, while it has minimal memory requirements (needing only to stack the state of the directory scanner each time it encounters a subdirectory), does not exhibit the desired behaviour: namely of finding all matching files in the root or currently logged directory first, prior to processing any sub-folders.

While some kind of hybrid approach might have proved satisfactory (for example: performing an initial match-only scan of the root folder before embarking on a full depth-first scan), the idea seemed kludgy and code space was already incredibly short. The chosen method, then, is a breath-first search which employs a LIFO list instead of the usual FIFO list. This approach is advantageous for several reasons:

- The LIFO list needs only a head pointer instead of head/tail pointers
- Memory consumed by the list (which is taken from the top of the filename buffer) is easily dynamically reclaimed (by removing elements from the list head) for use by filenames (which fill the same buffer from the bottom) without moving data around in RAM
- Files and folders in the root directory are fully searched before any sub-folders are processed (maintaining the search facility's usefulness as a means of quickly moving to the desired entry in the current folder)
- The implementation is simpler than that of a depth-first search

The chief advantage of the LIFO list, however, is its tendency to remain small even when an enormous number of folders are catalogued. A test volume with over 17,000 files and some 5,400 folders (which completely foxed the original LIFO implementation, which was hard-limited to just over 4,000 folders) produces a list of clusters which never exceeds 703 nodes (2,812 bytes) at any stage of the search. While the FIFO method tends to queue up the majority of folders on the volume before processing a significant number of them, the LIFO approach tends to probe the maximum folder depth early and thus keeps the cluster list short.

Perhaps the single major disadvantage of the LIFO method, however, is its property of processing folders in the reverse order to their position in directories on the disk. However, this was easily overcome by iteratively swapping the order of folders gathered from a single directory, so that a directory's children are processed in FIFO order. In reality, the list has properties of both a LIFO and FIFO data structure.

The resulting methodology processes the directory tree in the desired manner, in approximately the expected order, using a reasonably small amount of RAM, while being efficient and simple to implement.

The Horspool pattern matching algorithm is used by the search facility; the extremely light overhead of the shifting table does not adversely affect worst-case search scenarios (where pattern matching repeatedly fails at the very first comparison), and performance gains in average or best-case situations are considerable. Moreover, longer search phrases have barely any impact on search performance.

Prior to the release of version 3.1, it was discovered that superfluous redraws of the scroll bar during searches were wasting a large number of CPU cycles. Once that issue was rectified, search performance improved by 50 per cent. The time taken to complete the control test (searching for a four-character string in a FAT volume containing some 5,400 folders and 17,000 files) was reduced from some 3m55s in prior release versions to 2m58s.

Designing the SIDE3 Loader

Compared to SIDE/SIDE2, SIDE3/3.1 provides a greatly enhanced set of resources, including 8MB of ROM and 2-8MB of RAM. Owing to the fact that the SIDE Loader was already straining the limits of the 16K of ROM space allocated to it on the U1MB and Incognito, with SIDE3, the decision was made to always run the loader from the cartridge itself, even on U1MB systems. This immediately allowed the SIDE3 loader to stretch its legs and permitted me to focus on implementing new features instead of constantly hunting around for code space.

While the SIDE3 loader is essentially a port of my SIDE/SIDE2 loader (adapted to drive SD cards instead of Compact Flash cards), the new project presented an opportunity to exploit more powerful hardware, as well as making some changes which would eventually find their way back to the SIDE/XEL Loaders. The presence of SRAM on the SIDE3 enabled the FAT directory buffer to be enlarged from some 20KB of base system RAM to 64KB of cartridge SRAM. Of course, to actually make use of all this directory space, it was also necessary to overcome the 250 filename limit on the older loaders, and this had to be done without adversely impacting the speed and efficiency of the existing directory reading, sorting, and navigation code. Some time was spent replacing the 8-bit indexed menu pointer tables with 16-bit indexed equivalents, and since the directory is spread across only eight banks of SRAM and with pointers all confined to the same 8K of address space (in a ninth bank), it was possible to pack the bank numbers into the upper three bits of the 16-bit filename pointers (the lower 13 bits being sufficient to describe an address anywhere inside of an 8K RAM bank). Correspondingly, the maximum filename limit per directory was raised from 250 entries to 1,024 entries.

With bigger directories, having only ten lines of filenames per 'page' in the menu seemed insufficient, so the UI was revised to provide sixteen lines instead. Of course this change – along with the higher filename capacity and the greater number of 'pages' in a given directory – required the redesign of the scroll-bar code as well (this domino effect of breakage and re-writes will be familiar to any developer making significant changes to a project!).

Once all this was done, the redraw, formatted number output, and string handling code was revised for efficiency. Finally – although directory reading remains extremely quick – a progress message was added which displays a periodic indication (every 256 items) of how many filenames have been retrieved when a directory is read into memory.

The new loader was tested with the usual worst-case scenario recursive search tests, and performance remained comparable with the older, less complicated loader. Finally, optimised and rewritten code was back-ported (where applicable) to the older loader, as was the revised UI.

Since writing the original SIDE3 loader, I have implemented additional features in an update which is currently in its beta phase. Features include:

- Write access to FAT16 and FAT32 partitions, both by the loader itself and by the small FAT DOS which is installed when running XEX files
- A complete set of file management functions, including file renaming, deletion, and recursive copying of whole directory trees
- A totally rewritten 'FAT DOS' which provides a read/write FAT16/32 file system driver with an API compatible with DOS 2.5 and SpartaDOS
- Auto-mount cartridges (allowing the computer to power on and boot into to a specified cartridge image with no user intervention)

The DMA facility of SIDE3, which is currently used to load cartridge images at high speed from the SD card, is used by the beta loader's recursive file copier when reading from the SD card. It is hoped that a future revision of the hardware will also allow DMA writes.

Feedback

If you find any bugs or have ideas for added functionality, please contact me by email, or at AtariAge via PM (to user flashjazzcat) or in the Alt BIOS discussion thread.

References 92

Information consulted during development of the firmware under discussion and during the writing of this guide includes the following:

Altirra Hardware Reference Manual by Avery Lee

High Speed SIO Patch by Matthias Reichl

SIO2BT by Marcin Sochacki

Atari 1088XEL Mini-ATX Motherboard by Michael St Pierre

The SpartaDOS X Upgrade Project

Boyer–Moore–Horspool algorithm

Jonathan Halliday

January 2023