INDUSTRIAL COMPUTER SOURCE
TM

# Model EXM 2A Series
# Product Manual

**MANUAL NUMBER : 41417-020-4B**


***INDUSTRIAL COMPUTER SOURCE***

*9950 BARNES CANYON ROAD, SAN DIEGO, CA (619) 677-0877 (FAX) 619-677-0895*

***INDUSTRIAL COMPUTER SOURCE EUROPE*** *TEL (1) 69.18.74.30 FAX (1) 64.46.40.42* • ***INDUSTRIAL COMPUTER SOURCE (UK) LTD*** *TEL 0243-533900 FAX 0243-532949*

# FORWARD

This product manual provides information to install, operate and or program the referenced product(s) manufactured or distributed by Industrial Computer Source. The following pages contain information regarding the warranty and repair policies.

Technical assistance is available at: **1-800-480-0044**.

**Manual Errors, Omissions and Bugs:** A "Bug Sheet" is included as the last page of this manual. Please use the "Bug Sheet" if you experience any problems with the manual that requires correction.

## NOTE

The information in this document is provided for *reference* only. Industrial Computer Source does not assume any liability arising out of the application or use of the information or products described herein. This document may contain or reference information and products protected by copyrights or patents and does not convey any license under the patent rights of Industrial Computer Source, nor the rights of others.

This page intentionally left blank

# Guarantee

A thirty day money-back guarantee is provided on all **standard** products sold. **Special order products** are covered by our Limited Warranty, *however they may not be returned for refund or credit*.

## Refunds

In order to receive a full refund on a product purchase price, the product must not have been damaged by the customer or by the common carrier chosen by the customer to return the goods, and the product must be returned complete (meaning all manuals, software, cables, etc.) within 30 days of receipt and in as-new and resalable condition. The **Return Procedure** must be followed to assure prompt refund.

## Restocking Charges

Product returned *after* 30 days, and *before* 90 days, of the purchase will be subject to a **minimum** 20% restocking charge and any charges for damaged or missing parts.

Products not returned within 90 days of purchase, or products which are not in as-new and re-saleable condition, are not eligible for credit return and will be returned to the customer.

# Limited Warranty

One year limited warranty on all products sold with the exception of the "Performance Series" I/O products, which are warranted to the original purchaser, for as long as they own the product, subject to all other conditions below, including those regarding neglect, misuse and acts of God. Within one year of purchase, Industrial Computer Source will repair or replace, at our option, any defective product. At any time after one year, we will repair or replace, at our option, any defective "Performance Series" I/O product sold. This does not include products damaged in shipment, or damaged through customer neglect or misuse. Industrial Computer Source will service the warranty for all standard catalog products for the first year from the date of shipment. After the first year, for products not manufactured by Industrial Computer Source, the remainder of the manufacturer's warranty, if any, will be serviced by the manufacturer directly.

The **Return Procedure** must be followed to assure repair or replacement. Industrial Computer Source will normally return your replacement or repaired item via UPS Blue. *Overnight delivery or delivery via other carriers is available at additional charge.*

The limited warranty is void if the product has been subjected to alteration, neglect, misuse, or abuse; if any repairs have been attempted by anyone other than Industrial Computer Source or its authorized agent; or if the failure is caused by accident, acts of God, or other causes beyond the control of Industrial Computer Source or the manufacturer. Neglect, misuse, and abuse shall include any installation, operation, or maintenance of the product other than in accordance with the owners' manual.

No agent, dealer, distributor, service company, or other party is authorized to change, modify, or extend the terms of this Limited Warranty in any manner whatsoever. Industrial Computer Source reserves the right to make changes or improvements in any product without incurring any obligation to similarly alter products previously purchased.


**Shipments not in compliance with this Guarantee and Limited Warranty Return Policy will not be accepted by Industrial Computer Source.**

## Return Procedure

For any Limited Warranty or Guarantee return, please contact Industrial Computer Source's Customer Service at **1-800-480-0044** and obtain a Return Material Authorization (RMA) Number. All product(s) returned to Industrial Computer Source for service or credit **must** be accompanied by a Return Material Authorization (RMA) Number. Freight on all returned items **must** be prepaid by the customer who is responsible for any loss or damage caused by common carrier in transit. Returns for Warranty **must** include a Failure Report for each unit, by serial number(s), as well as a copy of the original invoice showing date of purchase.

> To reduce risk of damage, returns of product must be in an Industrial Computer Source shipping container. If the original container has been lost or damaged, new shipping containers may be obtained from Industrial Computer Source Customer Service at a nominal cost.

## Limitation of Liability

In no event shall Industrial Computer Source be liable for any defect in hardware or software or loss or inadequacy of data of any kind, or for any direct, indirect, incidental, or consequential damages in connection with or arising out of the performance or use of any product furnished hereunder. Industrial Computer Source liability shall in no event exceed the purchase price of the product purchased hereunder. The foregoing limitation of liability shall be equally applicable to any service provided by Industrial Computer Source or its authorized agent.

Some *Sales Items* and *Customized Systems* are **not** subject to the guarantee and limited warranty. However in these instances , any deviations will be disclosed prior to sales and noted in the original invoice. ***Industrial Computer Source reserves the right to refuse returns or credits on software or special order items.***

# Table of Contents

# List of Figures



# List of Tables



**Current Revision 4B**

**June 1996**

# Chapter 1: Product Description

This manual contains the information you will need to install and use the EXM-2A expansion module and accompanying software.

The EXM-2A is a semiconductor-based disk device implemented using flash and SRAM memory technology. A variety of different configurations and memory capacities are available.

The flash memory portion of the EXM-2A appears to software as a read-only hard disk, and it may serve as a boot disk (e.g., for diskless operation). The SRAM portion of the EXM-2A is read/write and appears as a non-bootable disk. Software is provided to format the SRAM and flash portions of the EXM-2A.

A read-only jumper for flash memory is provided.

A formatting program and an SRAM driver are provided on a floppy diskette included with the EXM-2A. Read the file **README.TXT** found on the EXM-2A distribution diskette for any up-dated information not included in this manual.

If you are unsure of the capacity of your EXM-2A, capacity can be determined by reading the part number sticker placed on the module itself (see Figure 1-1).

EXM2A-02-512
Flash SRAM

In this example, a card with the part number EXM2A-02-512 has two megabytes of flash memory and 512K of SRAM.

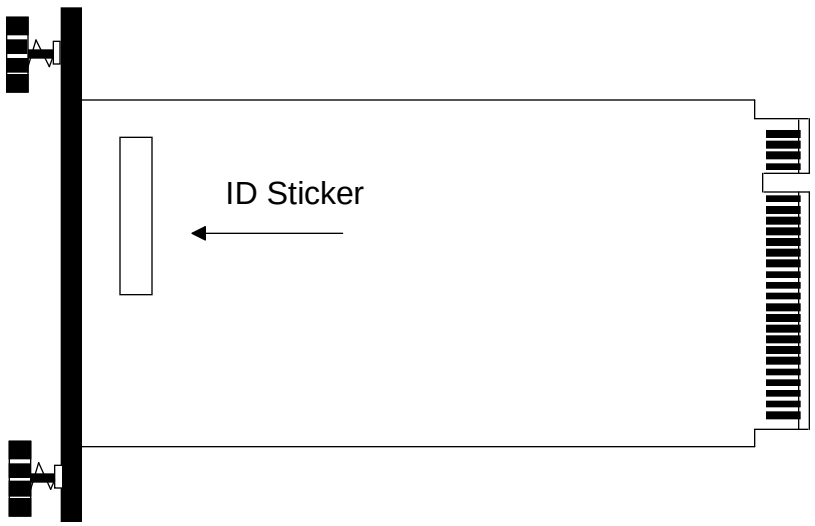


**Figure 1-1:** EXM-2A ID Sticker.

## Specifications

Table 1-1 defines the environmental and electrical specifications of the EXM-2A.

| **Charateristics** | | **Value** |
|---|---|---|
| Environmental Specifications | | |
| Temperature | operating | 0 to 60°C (* see below) |
| | storage | -40 to 85°C |
| Humidity | operating | 5% to 90% non condensing |
| | storage | 5% to 95% non condensing |
| Vibration | operating | .015" displacement with 2.5g peak (max) acceleration over 5 to 2000 Hz |
| | storage | .030" displacement with 5g peak (max) acceleration over 5 to 2000 Hz |
| Shock | operating | 30g 11ms duration, half-sine shock pulse |
| | storage | 50g 11ms duration, half-sine shock pulse |
| Electrical Specifications | | |
| Voltage | +5V Supply | +5V ±5% |
| | +12V Supply | +12V ±5% |
| Current | Typical | 5V @ 400mA |
| | Typical | 12V @ 40mA |
| Miscellaneous | | |
| Weight | | 3.0 ounces |

**Table 1-1:** EXM-2A Environmental and Electrical Specifications.

* Upper temperature limit degrades 2° C per 1000 ft. elevation. Maximum elevation 10,000 ft.

# Chapter 2: Installation

Before installing the EXM-2A, unpack and inspect it for shipping damage.

- Do not remove the module from its anti-static bag unless you are in a static-free environment. The EXM-2, like most other electronic devices, is susceptible to ESD damage. ESD damage is not always immediately obvious, in that it can cause a partial breakdown in semiconductor devices that might not immediately result in a failure.

- Ensure that the installation process as described herein is also performed in a static-free environment.

## Before You Begin

The process for installing and formatting the EXM-2A is as follows:

1. Set the read-only jumper, if necessary (see page 17).

2. Insert the EXM-2A into an EXM carrier.

3. Configure the C: and D: drive(s) in the BIOS as required.

4. Set the EXM-2A configuration in the EXM Setup Screen.

5. Format the EXM-2A flash and SRAM as required.

6. Load the SRAM device driver.

## Read-Only Jumper

If you wish to protect the flash memory from being reprogrammed by software, make sure the JP1 jumper is in the read-only (or write-disabled) position. The factory default position for this jumper is read/write (write-enabled). Note that the EXM-2A cannot be formatted if the jumper is in the read-only position.

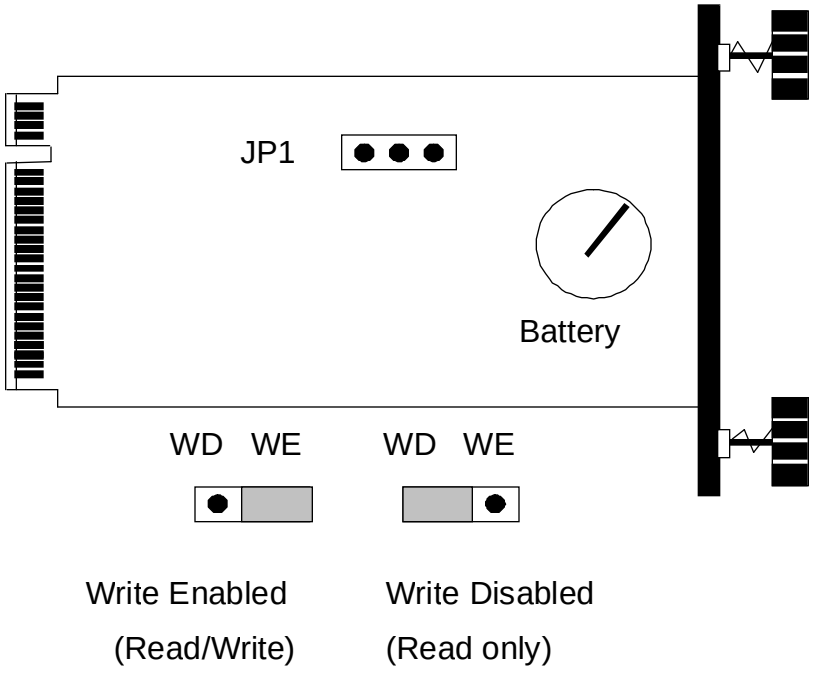


**Figure 2-1:** EXM-2A Read-Only Jumper.

## Insertion into an EXM Carrier

Insertion of the EXM-2A into an EXM carrier is straightforward. Remove a blank EXM panel from the carrier (by unscrewing the thumbscrews) and insert the EXM-2A into the card guides. Firmly press the EXM-2A front panel to ensure that the module is properly seated in the subplane and secure it with the thumbscrews. DO NOT OVERTIGHTEN.

- Make sure that power to your system is off. This EXM is not designed to be inserted or removed from a live system.
- When inserting the EXM, avoid touching the circuit board, and make sure the environment is static-free.
- Insert it with adequate continuous force rather than tapping or hammering on it.

Please note that the EXM-2A will not fit into slot 1 (top EXM slot) of an EPC-4 or slot 1 of an EPC-5. If slot 1 is the only open EXM slot of your EPC-4 or EPC-5, it will be necessary to move another EXM to slot 1 and install the EXM-2A in the vacated slot. Make sure to change all the appropriate values in the EXM Setup Screen.

## BIOS Setup Screen

The EPC BIOS contains a setup function to display and alter the system configuration. This configuration information is maintained in the EPC's battery-backed CMOS RAM and is used by the BIOS to initialize the system hardware.

Once the EXM-2A is inserted into the carrier, turn the system on and boot it up. The setup function can be invoked any time after the POST completes and first clears the screen. Simultaneously press the CTRL+ALT+ESC keys. This may be done during system operation in most, but not all circumstances. Some programs that take control of the keyboard at a low level, such as Microsoft Windows, cause this key sequence to be interpreted differently, or not at all. It should always work, however, when the DOS operating system prompt is shown on the screen.

The main setup screen displays similar to the following:

```
EPC-23 CMOS Setup, BIOS V3.05
486SL, 4MBytes Memory

Date (mm/dd/yy)..............01/12/93
Time (hh:mm:ss)..............07:34:56
Configuration Errors.........Halt on all errors

Diskette Drive A.............1.4M 3.5 inch
Diskette Drive B.............none
Fixed Disk Drive C...........Flash
Dixed Disk Drive D...........none
```

**Figure 2-2:** EXM Main Setup Screen.

## Configuring the Fixed Disk

Only the relevant part of the EXM Main Setup Screen, which concerns the fixed disk drive, is shown in Figure 2-1. This display-only field shows the type of disk selected from the fixed disk menu. Possible values vary with different CPUs. Not all values listed here apply to each CPU. Also, note that "Flash" may appear as "EXM Flash" in some systems. Possible values are None, AT, SCSI, VME and FLASH. To see the detailed characteristics of the device or to change the device, use the **F3** function key to go to the Fixed Disk Menu. The Fixed Disk Menu is used to define the type of hard disk(s) installed in the system.

The Fixed Disk Menu Screen resembles the following:

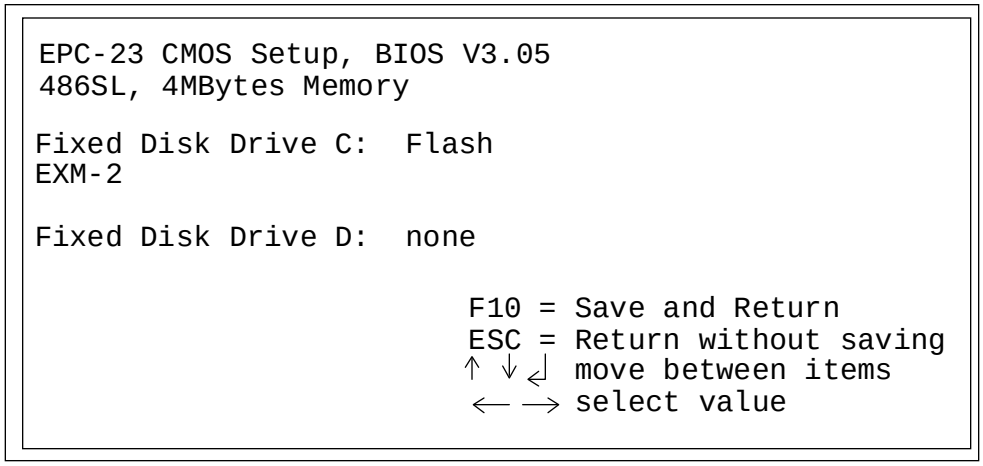


**Figure 2-3:** Fixed Disk Menu Screen.

Use the up and down cursor keys ( ↑ ↓ ) to move between items. Use the left and right cursor keys ( ← → ) to scroll through the available choices for each item.

Choose disk type **Flash**. An EXM flash disk can be made the boot device by making **Flash** the drive C: type.

Note that flash disks cannot be used as drive C: when a standard hard disk (SCSI or AT) is used as drive D:. Instead, set up the standard hard disk as drive C: and set up the flash disk as drive D:.

Once configuration of the fixed disk is complete, press the F10 key to return to the main setup screen.

## Enabling the EXM module

Once in the setup program, a menu displays which function keys are available for further configuration. Press the F2 function key to invoke the EXM menu. The screen display resembles the figure below.

| | ID | OB1 | OB2 |
|---|---|---|---|
| Slot 0 | FF | 00 | 00 |
| 1 | EC | 01 | 00 |
| 2 | **FD** | **01** | **00** |
| 3 | 7C | E1 | FF |
| 4 | F5 | 05 | 00 |
| 5 | FA | 25 | 00 |

**Figure 2-4:** EXM Setup Screen.

The EXM Setup Screen displays the EXM configuration data (in hexadecimal) stored in nonvolatile memory which the EPC uses at power-up to recognize and configure each installed EXM. An example of configuration data for the EXM-2A is shown in bold in slot 2. SLOT, ID, OB1 and OB2 are defined as follows:

SLOT — indicates the slot where the EXM is installed. See the diagram below to determine which EXM slot the EXM-2A occupies. Dashed lines indicate EXM slots not available on all systems.

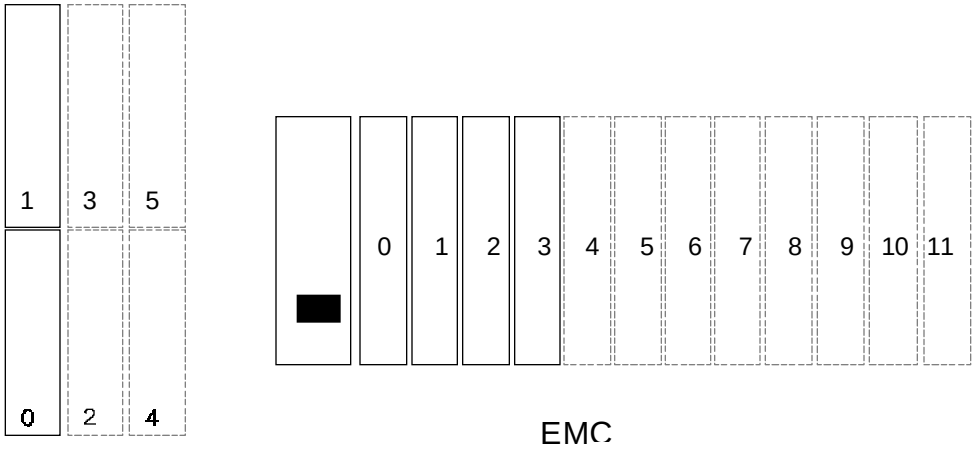


**Figure 2-5:** EXM Slot Numbering.

ID is a hard-wired ID value. Each EXM has a unique ID value.

OB1/OB2 are two bytes of option information.

All slots **not** occupied by an EXM module show an ID of FF and OB1/OB2 of 00 00 indicating no EXM is present.

To add or change an EXM configuration, use the cursor keys (arrows) to move between the fields on the screen. Move the cursor to the appropriate slot entry and type in the correct value.

The **ID** for the EXM-2A should be set to **FD**h.

**OB1** is a hexadecimal value derived by combining the following:

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 Revision | Bit 0 Card Enable |
|---|---|---|---|---|---|---|---|
| x | x | x | x | x | x | 0 | 0 disable<br>1 enable |

**Figure 2-6:** OB1 Register.

Bits 7-2 are undefined.

Bit 1 is a “Revision” bit and is set to 0 to help software distinguish this card from an EXM-2. This bit is not affected by writes.

Bit 0 is the standard EXM Card Enable bit. When set to 0, the card is disabled. Set the bit to 1 to enable the EXM-2A.

A typical OB1 value is 01h (00000001) indicating the card is enabled.

**OB2** is not used and should be set to 00.

# Chapter 3: Operations

## Distribution Diskette Contents

The EXM-2A distribution diskette contains the following files:

| | |
|---|---|
| **XFORMAT.EXE** | Flash formatting program |
| **SRAMDISK.SYS** | SRAM device driver |
| **BB5.00** | Boot block files for DOS 5.0 |
| **BB6.00** | Boot block files for DOS 6.0, 6.1, & 6.2 |
| **README.TXT** | Additional user information added after the manual was printed |

## Installation

Create a directory on the C: drive called **C:\EXM2A** (or another name you choose). Copy the files from the distribution diskette to the **C:\EXM2A** directory.

## Formatting Program

The first task that you are likely to perform with the EXM-2A once it is installed and configured is format the flash memory and copy directories and files to it. This procedure is performed by the program named **XFORMAT**. It is distributed on the floppy diskette shipped with the EXM-2A. This program formats both flash and SRAM.

When **XFORMAT** is used to format flash memory, the function first checks the status of the battery to determine if enough voltage is present, then erases all flash memory, formats it, optionally adds system files to make it a boot device, and the optionally copies a directory structure and files into it. The program is invoked with the following command line:

```
XFORMAT    [options]    [srcdir]
```

All files in directory srcdir (typically a directory on your hard disk) and all of its subdirectories (if any) and their files are copied into the EXM-2A. To make changes to the flash disk you must rerun the **XFORMAT** function.

Because SRAM disks are read/write, the `[srcdir]` parameter is optional for SRAM drive types. Files can be written to an SRAM device using standard DOS copy commands (**COPY**, **XCOPY**).

For flash, each file is created with the same attributes except the files are also marked as read-only and unarchived (see the DOS **ATTRIB** command for more information). For SRAM, each file is created with the same attributes as the corresponding source file. The date/time stamp for file modification or creation stays the same also.

The following flags are available when using the **XFORMAT** function. Note that a dash (-) can be used interchangeably with a slash (/) to set flags.

**/B=<sysdir>** **Boot Disk Flag**. Creates a bootable disk image using the DOS system files specified by the **<sysdir>** parameter. The function formats and copies files to the formatted disk. A bootable disk image is created. The operation fails if it cannot find the operating system files.

It is possible to create a bootable disk image from a non-bootable source drive. Refer to Appendix B for details.

This flag is for use with flash disks only.

**/F<name> <size>** **File Output Flag**. Outputs the disk image to the file **<name>** with size **<size>**. **<size>** is a hex value that specifies the number of Kbytes in the target file. This option is useful for creating VME memory disks and images for users that are going to write their own flash writing program, and is not strictly applicable to only the EXM-2A, but is documented here for completeness.

**/H** **Help Flag**. Displays a list of available flags and their options.

**/N=<ver>** **Non-Boot Version Flag**. Creates a bootable disk image using the boot block file specified by the **<ver>** parameter. (The location of the boot block file is specified by **/B** or **/S** flags.) This option is useful for creating bootable flash disks from non-bootable sources such as network and RAM disk drives (see Appendix B for more information on creating bootable images from non-bootable disks). The boot block file for **/N=5** is **BB5.00** (DOS 5) and the boot block file for **/N=6** is **BB6.00** (DOS 6.X). The boot block files are supplied with the EXM-2A.

This operation fails if it cannot find the operating system files.

This flag is for use with flash disks only.

**/P=<slot>** **Position Flag**. The slot number specified is the EXM slot number where the EXM-2A is located in the chassis.

**/Q=m** **Quiet Mode Flag**. Suppresses progress display messages. May be combined with the No Reboot Flag (i.e., **/Q=mr**).

**/Q=r** **No Reboot Flag**. Suppresses reboot upon successful completion. May be combined with the Quiet Mode Flag (i.e., **/Q=rm**).

It is necessary to flush the DOS buffer areas prior to accessing newly formatted disks. This is accomplished by rebooting. This reboot suppression flag is strictly for use when **XFORMAT** is used in a batch file where the reboot command is at the end of the batch file.

**/S** **System Flag**. Creates a bootable disk image using the DOS system files on the disk that **XFORMAT** is invoked from.

This operation fails if it cannot find the operating system files.

This flag is for use with flash disks only.

**/T=<type>** **Type Flag**. Specifies the type of format to be performed. When **<type>** is S, the function formats the SRAM. When **<type>** is **O**, the function formats the onboard flash (flash memory on the CPU board, not EXM-2A flash memory). When **/T=O** is selected, the **/P** flag is not allowed.

Specifying no option results in formatting the EXM flash.

**/V** **Volume Label Flag**. Prompts for a disk volume label. A volume label identifies the disk and can be a maximum of 11 characters. The label conforms to the MS-DOS convention, thus the following are not allowed: * . + , : ; < > = ? [ ] \ / | ( ) ^ & . Label letters are converted to upper case.

Note that not all flags are valid for every system. For example, **/T=O** is used only for systems that contain onboard flash memory (i.e., flash memory on the CPU). Also, the **/S**, **/B**, and **/N** flags are not valid for SRAM drives and generate an error message when attempts are made to use these flags on SRAM.

Some examples of the **XFORMAT** invocation are described below.

```
XFORMAT /B=a: a:
```

Formats the EXM-2A flash memory as a DOS system disk from the DOS system disk in drive a:, and copies all files from device a: into the EXM-2A (including subdirectories).

```
XFORMAT /T=S C:\SRAM
```

Formats the EXM-2A SRAM and copies all files and subdirectories of directory **C:\SRAM** into the EXM-2A.

```
XFORMAT a:
```

Formats the EXM-2A flash memory and copies all files from device A: into the EXM-2A (including subdirectories). The EXM-2A will not be bootable if not formatted with the **/S** or **/B** flag.

```
XFORMAT /N=6 /b=d: C:\FLASH
```

Format the flash memory using the system files and the boot block file **BB6.00** from directory **D:** and copy all files from **C:\FLASH**. Note that you cannot simply copy a file to a flash disk. You must use **XFORMAT** to erase, format, and copy the contents of a directory to the flash disk.

Unless the **/F** option is selected, **XFORMAT** automatically determines the amount of available space on your EXM-2A. It also calculates the amount of space needed for the files to be copied, and aborts without erasing the EXM-2A if there is insufficient space.

**XFORMAT** aborts if the source device (for either the file copy or for the system files) is the same EXM-2A being formatted.

Since the EXM-2A is represented to DOS as a fixed (non-removable) disk, DOS may have buffered data from the EXM-2A's files prior to its being formatted. As a result, you should reboot your system prior to accessing files in a newly formatted EXM-2A.

## SRAM Disk Device Driver

The SRAM driver **SRAMDISK.SYS** is used to configure the EXM-2A SRAM as a disk. The SRAM disk is not bootable, and is not visible to the operating system until the SRAM is formatted using **XFORMAT**, and the driver is loaded.

When **SRAMDISK.SYS** is loaded, it will check the status of the battery voltage. If the battery voltage is above 2.5V, the driver will load. If the battery voltage is below 2.5V, an error message displays for a few seconds before continuing.

To load the **SRAMDISK.SYS** driver, edit the **CONFIG.SYS** file and insert the following line:

```
DEVICE=[path]SRAMDISK.SYS /[slot]
```

The SRAM driver can be loaded high to conserve conventional memory. Use the following line under DOS 6.X:

```
DEVICEHIGH=[path]SRAMDISK.SYS /[slot]
```

The *path* you insert depends on what directory you create when you copy the EXM-2A software from the Industrial Computer Source distribution diskette to your target drive.

The *slot* you select depends on where the EXM-2A is installed in the EXM Setup Screen. For example, to configure the SRAM in an EXM-2A located in slot 3, use the following command:

```
DEVICE=C:\EXM2A\SRAMDISK.SYS /3
```

When installing SRAM drives, make sure you run **XFORMAT** on each drive ***before*** running the **SRAMDISK.SYS** driver for that drive. The driver(s) will not recognize the SRAM disk until it has been formatted using **XFORMAT**.

SRAM data integrity may be jeopardized when power is removed during a series of write operations to the SRAM. For this reason, **XFORMAT** checksums each sector. The **SRAMDISK.SYS** driver must compute the same checksum in order for the sector read to be successful. Corrupt sectors may be repaired with standard disk tools such as Norton Utilities.

## Error Messages

The following alphabetically listed error messages display during error conditions using **XFORMAT.EXE** or **SRAMDISK.SYS**.

| | |
|---|---|
| **Error Message** | BATTERY NEEDS REPLACEMENT |
| Explanation | The **XFORMAT** program or the **SRAMDISK.SYS** driver have detected that the battery power is below 2.5V. |
| Resolution | Replace the battery. Refer to Appendix A. |

**Error Message** CANNOT READ COMMAND.COM

Explanation The command processor **COMMAND.COM** could not be located.

Resolution Copy **COMMAND.COM** to the appropriate directory and re-invoke **XFORMAT**.

**Error Message** <path> COMMAND.COM WILL NOT BE COPIED TO THE TARGET

Explanation A version of **COMMAND.COM** was found in the top directory of the **<srcdiv>**.

Resolution Delete **COMMAND.COM** from **<srcdiv>** directory and re-invoke **XFORMAT**

**Error Message** INSUFFICIENT SPACE ON TARGET DEVICE

Explanation The source directory specified is too large to fit onto the target.

Resolution Remove some of the files from the source directory

**Error Message** INVALID SLOT NUMBER SPECIFIED

Explanation The slot number is not a decimal number

Resolution Ensure that the slot number does not contain any alpha or special characters

**Error Message** INVALID SOURCE PATHNAME

Explanation **XFORMAT** could not find the directory specified as the source pathname.

Resolution Ensure that the source pathname exists.

**Error Message** INVALID TARGET SIZE SPECIFIED

Explanation An illegal character was detected in the file size portion of the**/F** parameter.

Resolution Specify the file size using only hex characters.

**Error Message** MORE THAN ONE SOURCE PATH WAS SPECIFIED

Explanation Multiple source paths were detected on the command line.

Resolution Remove one of the source paths from the **XFORMAT**'s invocation line.

**Error Message** NO ARGUMENT SPECIFIED

Explanation The command line does not specify any operation

Resolution Execute **XFORMAT** with **/H** to display flags and options.

**Error Message** NO SLOT SELECTED FOR THE EXM-2A.

Explanation **XFORMAT** requires specification of a slot when formatting SRAM and EXM flash memory.

Resolution Rerun **XFORMAT** using the **/P**(osition) flag to indicate which EXM slot to format.

**Error Message** NO SRAM MEMORY IN SPECIFIED SLOT

Explanation The **/T=S** option was selected, however, no SRAM memory was detected on the card in the specified slot.

Resolution Ensure the **/P**(osition) flag is correct or that the **/T**(ype) flag is specified correctly.

**Error Message** SOURCE PATH NAME TOO LONG

Explanation Insufficient storage for the pathname string

Resolution Copy the files to a directory that is closer to the root.

**Error Message** SLOT <n> DOES NOT CONTAIN AN ENABLED EXM-2A.

Explanation The slot number specified in the **SRAMDISK.SYS** configuration line in **CONFIG.SYS** does not contain an enabled EXM-2A

Resolution Enable the EXM-2A in the EXM Setup Screen, or verify that the EXM-2A is in the correct slot.

**Error Message** SLOT NUMBER OUT OF RANGE

Explanation The EXM slot number listed does not contain an EXM-2A.

Resolution Ensure that the EXM slot specified contains an EXM-2A.

**Error Message** SRAM FILE SYSTEMS ARE NOT BOOTABLE. DO NOT SPECIFY THE /S, /N OR /B PARAMETERS.

Explanation **XFORMAT** does not create bootable SRAM disk.

Resolution Remove the **/S**, **/N** or **/B** flag.

**Error Message** SYSTEM DOES NOT CONTAIN ONBOARD FLASH

Explanation **/T=O** was specified for a system that does not contain onboard flash (flash on the CPU card).

Resolution Repeat the command without the **/T=O** option.

**Error Message** TARGET SIZE NOT SPECIFIED.

Explanation The **/F** option is ill-formed.

Resolution The syntax for the **/F** flag is **/FM<filename> <size>**. The filename is specified and is followed by the file size in kilobytes.

**Error Message** UNABLE TO INITIALIZE FLASH DEVICE.

Explanation **XFORMAT** cannot write to flash memory when the write disable jumper is installed, **OR**, no flash memory exists.

Resolution Put the JP1 jumper in the write-enable position prior to executing **XFORMAT**. Check that the system is properly connected to a +12V power supply.

**Error Message** UNABLE TO READ BOOT RECORD FILE

Explanation The boot record files are not in the **<srcdiv>** path.

Resolution Copy the boot record files to the proper directory.

**Error Message** UNABLE TO READ SRAM DRIVE

Explanation The **SRAMDISK.SYS** driver cannot read SRAM memory.

Resolution Ensure that SRAM disk is formatted. The OB1 option byte must be enabled

via the EXM Setup Screen.

| | |
|---|---|
| **Error Message** | UNRECOGNIZED FLAG. |
| Explanation | A miss-spelled or illegal flag was used. |
| Resolution | Check the command line to ensure only legal flags are used. |

## Theory of Operation

The SRAM and flash portions of the EXM-2A are visible to the operating system as separate logical drives. Applications that read the flash memory operate identically as if they were reading any "read only" media, such as CD-ROM disks.

DOS functions such as **COPY** and XCOPY will operate correctly when invoked from the flash disk.

**CAUTION**

Any function or application other than **XFORMAT** that attempts to write to the flash disk will cause the system to lock up. Typical applications that write to disk might be Norton Utilities or PC Tools; typical DOS functions would be **COPY**, **XCOPY**, **FORMAT**, and **FDISK**. **COPY** and **XCOPY** operate correctly from the SRAM disk.

This page intentionally left blank

# Chapter 4: Programming Interface

## Registers

This chapter contains information needed to write custom software drivers for the EXM-2A. If using the supplied software that supports EXM-2A as a disk device, skip this chapter. The EXM-2A defines the following registers in the I/O space.

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | I/O Port |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 100 |
| **Device ID Register** | | | | | | | | |
| x | x | x | x | x | x | x | CDEN | 102 |
| **Configuration Option Byte 1 Register** | | | | | | | | |
| Low Order Bits 0-7 of Flash/SRAM Address | | | | | | | | 8380 |
| **Low Address Register** | | | | | | | | |
| Low-Middle Bits 8-15 of Flash/SRAM Address | | | | | | | | 8381 |
| **Middle Address Register** | | | | | | | | |
| High Middle Bits 16-23 of Flash/SRAM Address | | | | | | | | 8382 |
| **Middle Address Register** | | | | | | | | |
| | | | | | | | | 8383 |
| **Flash Data Access** | | | | | | | | |
| | | | | | | | | 8384 |
| **SRAM Data Access** | | | | | | | | |
| | | | | | | | | 8385 |
| **Reserved** | | | | | | | | |
| High Order Bits 24-31 of Flash/SRAM Address | | | | | | | | 8386 |
| **High Address Register** | | | | | | | | |
| x | x | x | x | x | x | x | Batt ok | 8387 |
| **Battery Status** | | | | | | | | |

**Figure 4-1:** EXM-2A Registers.

The first two registers are standard read/write EXM registers for device identification and configuration. The EXM-2A responds to accesses to ports 100h and 102h only if its EXM expansion interface line -EXMID is asserted. Registers 8380 - 8382 and register 8386 are write-only registers. Registers 8383 and 8384 are read/write, and register 8387 is read-only.

## Configuration Registers

The Device ID register is an 8-bit read-only register at I/O address 100h, which may be read when ~EXMID is asserted. It returns the value FDh, the EXM-2A device ID.

The Configuration Option Byte 1 Register (OB1) is an 8-bit register at I/O address 102h which may be read or written when ~EXMID is asserted. The only writable bit in OB1 is CDEN, which specifies whether the EXM-2A is enabled (1) or disabled (0). If disabled, the EXM-2A will not respond to the 8380-8387 I/O addresses; it will only respond to reads from I/O port 100h and reads and writes from I/O port 102h (if -EXMID is asserted). During reads, the EXM-2A returns bit 1 in OB1 as a 0. This reveals the board as an EXM-2A (as opposed to an EXM-2). Ensure that the CDEN bit is set to 0 if using multiple EXM-2As in a system.

## Addressing Registers

The flash and SRAM devices are accessed by placing an address in the four address registers and then reading or writing the appropriate data register, which causes the data byte at the location specified in the address registers to be read or written. The four address registers are write-only. The EXM-2A does not implement the High address register (8386). The I/O address 8386 is reserved for future use.

The EXM-2A provides a means of performing fast reads and writes of sequential bytes in the flash memory or SRAM. After each read or write access, the Low Address Register is incremented, allowing the next byte of data to be accessed without re-writing the address registers. When the Low Address Register reaches FFh, the next access will increment the Low Address Register to 00h, but it will not affect the value of the other address registers.

Only the lower eight address bits are auto-incremented on a read/write to the SRAM or flash. Thus, the maximum string I/O read or write length that can be issued by software to access the flash or SRAM data is limited to 256 bytes. To obtain this maximum length, software must first load the address registers with an address that is aligned to a 256 byte boundary. To read a subsequent, contiguous 256 bytes, the software must manually update the Low-Middle, High-Middle, and High registers (and the lower address register, if the previous string read/write to flash/SRAM was not 256 bytes in length) before issuing another 256 byte string read/write.

## Flash Data Access

Flash disk data can be read from address 8383. Writing to this address will cause a write to a flash device control register, which may result in unpredictable results. The details of manipulating the flash devices, such as using their command register and identifier, are not specified here; consult the datasheets for the Intel 28F008 flash memory devices. Note that some functions require the write-protection jumper to be in the write-enabled position.

### SRAM data access

The battery-backed SRAM is accessed in a similar fashion as the flash. The three low order address registers are set to the appropriate SRAM address and then an I/O read or write is performed to I/O address 8384 to read or write a byte of SRAM.

Address aliasing occurs when accessing the lower density SRAM chips. This may be used by software to determine the size of memory installed in the EXM-2A.

- When a single 32Kx8 is installed, address aliasing begins at the 32KB boundary.
- When 128Kx8 chips are installed, address aliasing begins at the 1MB boundary.
- When 512Kx8 chips are installed, address aliasing begins at the 4MB boundary.

Note that during power-down transition there is a very small probability that a single byte of SRAM or flash memory could be incorrectly written. This is the same problem that a disk drive has if it is powered off during a sector write.

### Battery Low Condition

If bit 0 is set to 0 in register 8387, the battery needs replacing because the voltage is less than approximately 2.5V. Note that the **XFORMAT** will fail when the battery is low. **SRAMDISK.SYS** issues a warning before continuing. See Appendix A for more information on how to change the battery of the EXM-2A. The remainder of the bits in register 8387 are undefined.

## SRAM Standard Memory Array

Typically, access to the SRAM is via a file system installed by the device driver **SRAMDISK.SYS**. For users who wish to bypass the file system and use the SRAM as a standard memory array, the following ANSI C routine is provided as an example of how to program an executable file.

```
#include <stdio.h>
#include <memory.h>
#include <conio.h>

typedef unsigned char UCHAR;
typedef unsigned short USHORT;
typedef unsigned long ULONG;

#define FAR             _far
#define BYTESPERSECTION 0x100
#define EXMID           0x96
#define EXMENABLE0x00000001
#define LSWLOWBYTE      0x8380
#define LSWHIGHBYTE     0x8381
#define MSWLOWBYTE      0x8382
#define MSWHIGHBYTE     0x8386
#define SRAMDATA 0x8384
```

```
UCHAR     Slot   =      5;           /* presumes EXM-2A in slot 5 */

void
readsection(UCHAR FAR *target, ULONG sramoffset)
{   UCHAR ob;
    register USHORT rinductor;

    /*
    //    Enable the card to allow register access
    */
    outp(EXMID,Slot);
    ob = inp(0x102);
    outp(0x102,ob | EXMENABLE);
    /*
    //    Load initial offset value (sramoffset should be
    // divisible by 256)
    */
    outp(MSWHIGHBYTE,(USHORT) (sramoffset >> 24));
    outp(MSWLOWBYTE,(USHORT) (sramoffset >> 16));
    outp(LSWHIGHBYTE,(USHORT) (sramoffset >>8));
    outp(LSWLOWBYTE,(USHORT) sramoffset);
    /*
    //    Read a section using the autoincrement feature of the EXM-2A.
    */
    for (rinductor = 0; rinductor < BYTESPERSECTION; rinductor++) {
       *target++ = (UCHAR) inp(SRAMDATA);
    }
    outp(0x102,ob);     /* restore the option byte */
}

void
writesection(UCHAR FAR *source, ULONG sramoffset)
{   UCHAR ob;
    register USHORT rinductor;

    /*
    //    Enable the card to allow register access
    */
    outp(EXMID,Slot);
    ob = inp(0x102);
    outp(0x102,ob | EXMENABLE);
    /*
    //    Load initial offset value (sramoffset should be
    // divisible by 256)
    */
```

```
    outp(MSWHIGHBYTE,(USHORT) (sramoffset >> 24));
    outp(MSWLOWBYTE,(USHORT) (sramoffset >> 16));
    outp(LSWHIGHBYTE,(USHORT) (sramoffset >>8));
    outp(LSWLOWBYTE,(USHORT) sramoffset);
    /*
    //    Write a section using the autoincrement feature of the EXM-2A.
    */
    for (rinductor = 0; rinductor < BYTESPERSECTION; rinductor++) {
       outp(SRAMDATA,*source++);
    }
    outp(0x102,ob);     /* restore the option byte */
}

void
main()
{   UCHAR test[BYTESPERSECTION];
    UCHAR readbuffer[BYTESPERSECTION];
    register USHORT minductor;

    for (minductor = 0; minductor < BYTESPERSECTION; minductor++)
   test[minductor] = minductor;
    writesection(test,0);
    readsection(readbuffer,0);
    if (memcmp(test,readbuffer,BYTESPERSECTION))
       printf("Buffer did not compare \n\r")
    else printf("Buffers comparison ok \n\r"):
}
```

This page intentionally left blank

# Appendix A: Replacing the Battery

The battery powers the SRAM when system power is not present. At 60°C, the battery should have a shelf life of over four years. In a system that is powered on much of the time and where the ambient power-off temperature is less than 60°C, the battery is estimated to have a life of 10 years.

The EXM-2A contains a "supercap" to allow the user to replace a mostly drained battery without losing SRAM contents. When replacing a battery that is not fully drained, (i.e., >2.0V) the user has 1 minute to remove and replace the battery without losing the SRAM contents. It is a good idea to make sure you have your replacement battery ready for installation before you remove the old battery.

The battery supplied with the EXM-2A is a 23mm 3V Lithium "coin" battery or equivalent (e.g. Panasonic BR2330 or Rayovac BR2335). It is mounted near the front center of the printed circuit board, close to the front panel. Should the battery fail, you may obtain and install a replacement. Figure 8 below illustrates how to change the battery.

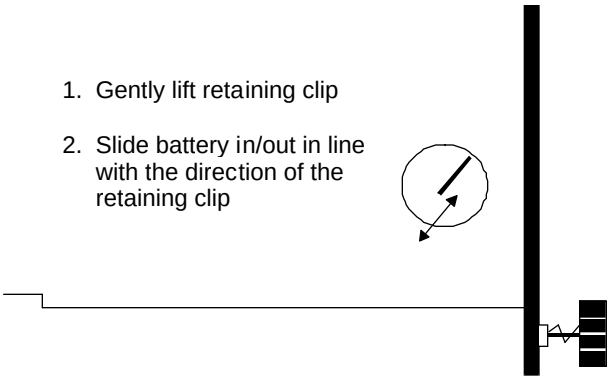


**Figure A-1:** Battery Replacement.

**CAUTION**

When installing the coin cell battery, be careful not to short the battery socket terminals together with the case of the battery. Doing so may cause loss of the SRAM contents.

The EXM-2A can sustain a 5 second short of the battery socket terminals without data loss if the battery being replaced is 2V or greater.

This page intentionally left blank

# AppendixB: Creating Bootable Disks From Non-Bootable Drives

In order to create a bootable disk image from a non-bootable drive, use the **XFORMAT** function with the **/N** flag. Refer to the example below:

```
XFORMAT /N=6 /B=D: C:\FLASH
```

This example assumes that the D: drive is a non-bootable device, such as a network drive or a RAM disk, and contains the necessary system files; that DOS 6 is the operating system; and that files from the **C:\FLASH** directory will be copied to the system disk.

The drive specified by the **/B=** option **must** contain the system files **COMMAND.COM**, **IO.SYS** and **MSDOS.SYS** in addition to the **BB6.00** boot block file. These files are automatically copied to the flash disk root directory when the disk is made bootable. If these system files are not present, an error message displays and **XFORMAT** fails.

Since the **IO.SYS** and **MSDOS.SYS** files are hidden (i.e., do not display when using a **DIR** command), it is necessary to unhide them so that the DOS **COPY** command can transfer the files. For example, to reveal the attributes set for the **MSDOS.SYS** file, type the following:

```
ATTRIB MSDOS.SYS
```

Letters that display include S (system file), H (hidden file) and R (read-only). To prepare the hidden file **MSDOS.SYS** for copying, type either of the following:

```
ATTRIB -s -h MSDOS.SYS
```

or

```
ATTRIB MSDOS.SYS -s -h
```

Refer to the *Microsoft MS-DOS User Guide and Reference* or use the online help by typing **ATTRIB /?** at the command line for more information about the **ATTRIB** command and various file attributes.

This page intentionally left blank

# BUG REPORT

While we have tried to assure this manual is error free, it is a fact of life that works of man have errors. We request you to detail any errors you find on this BUG REPORT and return it to us. We will correct the errors/problems and send you a new manual as soon as available. Please return to:


Attn: Documentation Department
P. O. Box 910557
San Diego, CA 92121-0557

Your Name: _____

Company Name: _____

Address 1: _____

Address 2: _____

Mail Stop: _____

City: _____ State: _____ Zip: _____

Phone: (_____) _____

Product: **EXM 2A Series**

Manual Revision: **41417-020-4B**

Please list the page numbers and errors found. Thank you!