

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Find the *RadiSys EXM-17* at our website: *Click HERE*

EXM-17

PCMCIA Adapter Module

Reference

RadiSys® Corporation

15025 S.W. Koll Parkway

Beaverton OR 97006

(503) 646-1800

FAX: (503) 646-1850

07-0136-02 April 1994

EPC and RadiSys are registered trademarks and EPConnect is a trademark of RadiSys Corporation. IBM and PC/AT are trademarks of International Business Machines Corporation. Microsoft and MS-DOS are registered trademarks of Microsoft Corporation. Award is a registered trademark of Award Software, Inc. Databook is a registered trademark of Databook, Inc. HP is a registered trademark of Hewlett-Packard Corporation.

March 1993

Copyright © 1993, 1994 by RadiSys Corporation

All rights reserved.

Hardware Warranty

RadiSys Corporation ("RadiSys") warrants the EPC system and component modules to the original purchaser for two years from the product's shipping date. If an EPC product fails to operate in compliance with its specification during this period, RadiSys will, at its option, repair or replace the product at no charge. The customer is, however, responsible for shipping the product; RadiSys assumes no responsibility for the product until it is received. This warranty does not cover repair of products that have been damaged by abuse, accident, disaster, misuse, or incorrect installation.

RadiSys' limited warranty covers products only as delivered. User modification, such as the addition of memory arrays or other devices, may void the warranty, and if the product is damaged during installation of the modifications, this warranty does not cover repair or replacement.

This warranty in no way warrants suitability of the product for any specific application.

IN NO EVENT WILL RADISYS BE LIABLE FOR ANY DAMAGES, INCLUDING LOST PROFITS, LOST SAVINGS, OR OTHER INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PRODUCT EVEN IF RADISYS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, OR FOR ANY CLAIM BY ANY PARTY OTHER THAN THE PURCHASER.

THE ABOVE WARRANTY IS IN LIEU OF ANY AND ALL OTHER WARRANTIES, EXPRESSED OR IMPLIED OR STATUTORY, INCLUDING THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR USE, TITLE AND NONINFRINGEMENT. Repair or replacement as provided above shall be the Purchaser's sole and exclusive remedy and RadiSys' exclusive liability for any breach of warranty.

NOTES

Table of Contents

1. Product Description 1
1.1 Introduction 1
1.2 Specifications 2

2. Hardware Installation 3
2.1 Inspection 3
2.2 Insertion in an EXM Carrier 3
2.3 Configuring the BIOS Setup 4

3. Software Installation 7
3.1 Before Beginning 7
3.2 Running the INSTALL Program 7
3.3 Changing AUTOEXEC.BAT 8
3.4 Changing CONFIG.SYS 8
3.4.1 Memory Cards 9
3.4.2 I/O Cards 9
3.4.3 Booting Considerations 9
3.5 Verifying Software Installation 10
3.6 Installation Error Messages 11

4. Operation 13
4.1 I/O Cards 13
4.1.1 Modem Cards 13
4.1.2 Network Cards 13
4.2 Memory Cards 14
4.2.1 Static Random Access Memory (SRAM) Cards 14
4.2.2 Flash EPROM Cards 14
4.2.3 Which Memory Card to Use? 15
4.3 Formatting Memory Cards 15
4.3.1 Formatting SRAM Cards 16
4.3.2 Formatting Flash EPROM Cards 16
4.4 Specifying Memory Card Information 18
4.4.1 Card Information Structure (CIS) 19
4.4.2 Configuration Files 19
4.4.3 Pre-Defined Logical Configuration Files 20
4.4.4 Specifying Part Numbers 22
4.4.5 Specifying Generic Information 22
4.4.6 Placing CIS onto Memory Cards 23

4.5 Formatting Multi-Partition Cards 24
4.5.1 Overview 24
4.5.2 Installing the Device Driver for Multi-Partitions 24
4.5.4 Formatting a Two-Partition SRAM Card 25
4.5.5 Formatting a Two-Partition Flash Card 26
4.6 Formatting Bootable Memory Cards 28
4.6.1 Making a Bootable SRAM Card 29
4.6.2 Making a Bootable Flash EPROM Card 29
4.6.3 Bootable Flash Card Contents 30
4.7 Using REMAP.SYS 30

5. Restrictions 31
5.1 Overview 31
5.2 Conflicts with VME/VXI 31
5.3 EXM-1 31
5.4 EXM-10 31
5.5 EXM-17 as Serial Port or Modem 32
5.6 EPC-7 Conflicts 32

6. Support and Service 33

Appendix A: TC Commands Reference A-1

Appendix B: Configuration File Keywords B-1

Appendix C: PCMCIA Card Codes C-1

Index I-1

Figures

Figure 1. EXM Setup Screen 4
Figure 2. EXM Slot Numbering 5

Tables

Table 1. EXM-17 Specifications 2
Table 2. EXM-17 OB1 Field 6
Table 3. PCICCARD.SYS Driver Error Codes 12

1. Product Description

1.1 Introduction

This manual contains the information required to install and use the EXM-17 Personal Computer Memory Card International Association (PCMCIA) adapter expansion module.

The EXM-17 is an expansion module containing a single PCMCIA-compatible socket that accommodates Type I and Type II cards. PCMCIA is currently a DOS-only standard, therefore the EXM-17 functions only as a DOS device. The EXM-17 is designed for use with all EMC processors, the EPC-5 VME processor and the EPC-7 VXI processor. Refer to *Chapter 5 - Restrictions* for more information about using the EXM-17.

The EXM-17 PCMCIA adapter supports the following devices:

- solid state disks such as SRAM and Flash memory cards.
- Ethernet LAN cards
- Token-ring LAN cards
- modem and fax interfaces

Contact RadiSys Corp. for information about additional PCMCIA devices supported.

1.2 Specifications

The following table defines the environmental and electrical specifications of the EXM-17.

Environmental		
Temperature	operating	0° to 60° C (* see below)
	storage	-40° to 85° C
Humidity	operating	5% - 95% (non-condensing)
	storage	5% - 95% (non-condensing)
Vibration	operating	.015" PP 2.5g (max) 5-2000 Hz
	storage	.030" PP 5g (max) 5-2000 Hz
Shock	operating	30g 11 msec duration
	storage	50g 11 msec duration
Electrical		
Power	maximum	+5V @ 800 mA
	typical	+5V @ 400 mA

Table 1. EXM-17 Environmental, Electrical, and Physical Specifications.

* Upper temperature limit degrades 2° C per 1,000 ft. elevation. Maximum elevation 10,000 ft

Note that the power consumption specification is tested with no PCMCIA card present. Power consumption varies depending upon the PCMCIA card inserted.

2. Hardware Installation

2 2

2.1 Inspection

Before installing the EXM-17, 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-17, 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.**

2.2 Insertion in an EXM Carrier

Insertion of the EXM-17 into an EXM carrier is straightforward. Remove a blank EXM panel from the carrier (by unscrewing the thumbscrews) and insert the EXM-17 into the card guides. Firmly press the EXM-17 front panel to ensure that the module is properly seated in the subplane and secure it with the thumbscrews.

- **MAKE SURE THAT POWER TO YOUR SYSTEM IS OFF. THE 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.**

2 2

2.3 Configuring the BIOS Setup

The EXM configuration data in the EPC to which the EXM-17 is connected needs to be modified to recognize and enable the card and select from the available options. Follow the instructions in the EPC reference manual for your specific EPC model to invoke the BIOS setup function. This is typically done by simultaneously pressing the CTRL-ALT-ESC keys. EPC system configuration procedures differ slightly depending on BIOS versions.

Once in the setup program, a menu displays specifying which function keys are available for further configuration. Press the **F2** function key to invoke the EXM menu. The screen displays (similar to the one below) the following information:

```
          ID OB1 OB2
Slot 0    DE  01  00
     1    FF  00  00
     2    FF  00  00
     3    FF  00  00
     4    FF  00  00
     5    FF  00  00
     6    FF  00  00
     7    FF  00  00
```

Figure 1. EXM Setup Screen

This display is the EXM configuration data (in hexadecimal) stored in nonvolatile memory, which the EPC uses at power-up to recognize and configure each installed EXM. The displayed data shows SLOT, ID, OB1 and OB2. OB1 and OB2 are option bytes used to set up additional configuration information.

SLOT, ID, OB1 and OB2 are defined as follows:

SLOT indicates the slot in which the EXM is installed. See Figure 2 below to determine which slot the EXM occupies. Dashed lines indicate that not all slots are available in some configurations.

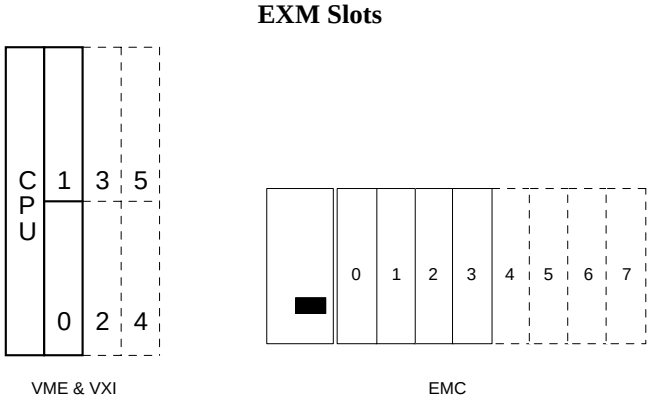


Figure 2. EXM Slot Numbering

Note that all slots are listed even if the system configuration does not have all the possible EXM slots. All slots **not** occupied by an EXM module should show an ID of FF and OB1/OB2 of 00 00 indicating that 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.

ID is a RadiSys-assigned value. The **ID** for an EXM-17 should be set to **DE**h.

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

Bits 7 and 6 are unused.
The ROM base address is configured using bits 5, 4 and 3.
The Select bit is configured using bits 2 and 1.
The card is enabled or disabled using bit 0.

Refer to Table 2 on the next page for more information on OB1.

OB2 **OB2** is not used by the EXM-17 and should be set to **00**h.



Unused (bit 7,6)	ROM Base Address (bits 5,4 & 3)	Reserved bits (bits 2 & 1)	EXM-17 card enable (bit 0)
Set to 0 0	000 - ROM disabled 001 - C8000 010 - CC000 011 - D0000* 100 - D4000* 101 - D8000* 110 - DC000* 111 - E0000	Set to 0 0	0 - disabled 1 - enabled

Table 2. EXM-17 OB1 Field

Bits 7 and 6 are **unused.**

ROM Base Address - Bits 5, 4 and 3 determine where the EXM-17 resides in memory space. This 8K ROM memory is used for booting Flash and SRAM cards. If booting is not used, disable the ROM base address to preserve memory space.

Make sure that the 8K boot ROM memory address chosen for this EXM-17 does not conflict with any other modules that use the same memory address (or range).

001 (C8000) and 010 (CC000) are the only workable settings for the EPC-5.

111 (E0000) is the only workable setting for the EPC-7.

* These addresses are not currently supported. The D-page is a memory window and is consumed by the EXM-17. The D-page is therefore not available for the ROM base address. Memory managers for *Windows*, *EMM386*, and other products must be set to exclude this memory area, and no other I/O cards can be mapped here.

Reserved Bits - Bits 2 and 1 are reserved for future use and should be set to **0**.

Card Enable - Bit 0 enables or disables the EXM-17.

3. Software Installation

3 3

The EXM-17 recognizes both Type I and Type II PCMCIA-compatible cards once the software included with the EXM-17 is installed.

Installing this software is straightforward. A simple DOS batch file is included for installing the software.

Please note that this software is intended for DOS-based environments and has been tested with DOS 5.0. Contact RadiSys Technical Support for information on other supported environments.

3.1 Before Beginning

You must have access to a 3.5" floppy drive and at least one Mbyte of disk storage to install the software.

3.2 Running the INSTALL Program

The supplied disk labeled *EXM-17 System Software* contains an installation program (INSTALL.BAT), a set of TC command utilities such as `TCXCOPY` and a number of configuration files for PCMCIA cards. Be sure to make a backup of the disk before proceeding.

To install the software, place the disk in the floppy disk drive and then type the following:

```
A:INSTALL
```

Do not manually copy the files off the disk! `INSTALL` creates a new directory named C:\PCICCARD and places the new files and subdirectories there. **Please note that you must modify the `AUTOEXEC.BAT` and `CONFIG.SYS` files *manually* after the install program runs.**

3 3

The EXM-17 always maps to the lowest available drive letter. In the following examples, it is assumed that the lowest available drive letter is D:. Note that even though the EXM-17 usually is the D: drive, the drivers that fully enable it are loaded via AUTOEXEC.BAT and CONFIG.SYS from the boot drive.

The installation program creates directories for each different memory card manufacturer and then adds the current configuration files for each model number. Some directories can contain a substantial number of files.

`INSTALL` also supplies a number of other files with information about the various card types supported. Several subdirectories are created that contain predefined configuration files. **Do not remove any of these files or subdirectories unless you make backups.** These files are used to format, program and initialize PCMCIA cards.

3.3 Changing AUTOEXEC.BAT

`INSTALL` does not automatically make changes to the `AUTOEXEC.BAT` file. Change the `AUTOEXEC.BAT` file manually. You must first add the C:\PCICCARD directory to the `PATH` statement to be able to use the EXM-17 system software commands.

To define default values for a particular "TC command" (such as TCFORMAT, TCERASE, etc.) set the appropriate environment variable. Refer to Appendix A for detailed information about each TC command.

3.4 Changing CONFIG.SYS

`INSTALL` does not automatically make changes to the `CONFIG.SYS` file. Read this section carefully to change the `CONFIG.SYS` file manually.

3.4.1 Memory Cards

For memory cards, add the following line to the CONFIG.SYS file to install the EXM-17 memory card driver at power-up:

```
DEVICE=C:\PCICCARD\PCICCARD.SYS
```

If the device driver is moved to a different directory, be sure that CONFIG.SYS is updated accordingly. "DEVICEHIGH" can be substituted for "DEVICE" in the above line (except for EPC-7 users, who should not use DEVICEHIGH).

3.4.2 I/O Cards

I/O cards such as modems are configured similar to memory cards using the PCICCARD.SYS driver loaded by the CONFIG.SYS file.

Token-ring and Ethernet cards require customized installation routines that load the correct device drivers into CONFIG.SYS. The EXM-17 software does not contain any network card drivers. Refer to the documentation and the software accompanying the network card for more information.

3.4.3 Booting Considerations

To configure the PCMCIA card as a boot disk, the card must contain the required drivers and DOS system files. In addition, bits 3, 4 and 5 of the OB1 register in the BIOS setup must be set to something other than 000 to enable boot ROM. During the boot process, the card is automatically assigned as the A: drive. To reassign A: back to the floppy drive, install REMAP.SYS in CONFIG.SYS. Refer to Chapter 4 for detailed instructions about REMAP.SYS.

There are additional considerations when booting from an EXM-17 installed in an EPC-7 VXI controller. Refer to Section 5.6 for more information.

If not booting from a memory card, put the PCMCIA drivers on the C: drive (or A: drive if booting from a floppy) and reference them there.

3 3

3.5 Verifying Software Installation

After installing the software, the computer must be restarted before the EXM-17 is ready for use.

During normal system startup, if the PCICCARD.SYS device is installed with a memory card, messages similar to the following display (depending on the configuration):

```
AWARD / DATABOOK PCMCIA Card Device Driver VX.XX
Copyright (C) Databook Incorporated 1990-92. All rights reserved.
Copyright (C) Award Software Inc. 1992. All rights reserved.
JEIDA/PCMCIA
        Installed as DOS drive D:.
```

To confirm that everything is working once the system has booted up, remove any PCMCIA card from the EXM-17 (now the D: drive) then enter the following command at the DOS prompt:

```
TCFORMAT -V -NOW D:
```

The system should display the following message:

```
tcformat VX.XX Mar 12 1992
Copyright (C) Databook Incorporated 1990-92.
?tcformat-F-can't access card 'd:' -- no card in drive.
```

If the driver is not properly installed the following messages display:

```
tcformat VX.XX Mar 12 1992
Copyright (C) Databook Incorporated 1990-92.
?tcformat-F-can't access card 'd:' -- driver not responding.
```

This error indicates one of the following possible problems:

1. An unsupported version of DOS. The EXM-17 requires DOS version 5.0 or later.

2. Wrong drive letter. Re-check the drive letter assignment message displayed during startup. The EXM-17 is always set for the lowest available drive letter, usually drive D:.

3. Too few drives in DOS. Add or modify the following command line in the CONFIG.SYS file:

   `LASTDRIVE=LETTER`

   Choose a letter that is higher than the drive letter displayed during startup.

3.6 Installation Error Messages

The following error messages may be reported by the device driver during system startup, after displaying the driver version and copyright notice:

`Error in command tail: text from config.sys`

This indicates that the `PCICCARD.SYS` driver found an error and is *not* installed. Correct the command line in `CONFIG.SYS` and restart the computer.

`Drive installation failed: code failure-code. Consult manual.`

This indicates that the `PCICCARD.SYS` driver found a hardware problem while running its diagnostics. The device driver is *not* installed. The failure-code is a 3- or 4-digit hexadecimal code which can help narrow down the cause of the problem. Refer to the following table.

3 3

Code	Meaning
10x	Basic diagnostic failure. Either the EXM-17 is not present, or is set at the wrong address, or there is an address conflict. Or there is a hardware failure. x is a hex code.
20xx	Indicates a failure in the drive address latches. This error appears if there is an addressing conflict. It can also indicate a hardware failure. "xx" is a 2-digit hex code which indicates the bit number that failed.
30x	Failure in the drive address counter. This error may appear if the hardware is defective. Could also indicate an addressing conflict or a compatibility or speed conflict. If x = 0, the hardware is defective or there is an address conflict. If x = 2 or 3, there are speed conflicts or there is a noise problem.
40x	Indicates a failure in the drive address counter. The test, the error codes, and the interpretation are similar to 30x. If x = 1 or 2, then possibly the drive is defective or there is an addressing conflict. If x = 4 or 5, there are speed conflicts or there is a noise problem.
50x	Indicates a failure in the drive address counter. "x" indicates the value of the address counter after the test. A failure in this test indicates either an addressing conflict or defective hardware.

Table 3. PCICCARD.SYS Driver error codes.

4. Operation

This chapter is designed to provide basic information about preparing various PCMCIA cards that operate with the EXM-17. Included are I/O cards, such as modem and network interface cards, and memory cards such as SRAM and Flash EPROM cards.



4.1 I/O Cards

Two types of I/O cards supported by the EXM-17 at this printing are modem cards and network interface cards.

4.1.1 Modem Cards

A data only or fax+data modem PCMCIA card can be used with the EXM-17. If the `PCICCARD.SYS` driver has been installed into `CONFIG.SYS`, the driver "maps" the PCMCIA card to the lowest "free" COM port between COM1 and COM4.

The modem card can be inserted or removed from the EXM-17 while the computer power is *ON*. This "hot insertion" is one of the major advantages of the PCMCIA standard. However, note that it is necessary to **reboot** after the modem is inserted. The modem cable should be connected to the modem card *before* inserting the modem card into the EXM-17. This ensures that the modem is correctly configured for the international telephone system that it is to be used with.

Modem cards can be accessed via standard communications software at the given setting.

4.1.2 Network Cards

PCMCIA-compatible network cards such as the IBM Token-Ring 16/4 Credit Card Adapter provide an interface between the EPC computer and token-ring networks.

PCMCIA network cards are usually supplied with their own software drivers and installation procedures. Refer to the documentation that accompanied the network card for more information.

4.2 Memory Cards

A memory card is a storage device similar in many ways to a disk drive. Since memory cards contain no moving parts they are more robust and more reliable than hard drives or floppy disks. The EXM-17 is designed to accept memory cards that conform to the PCMCIA Type I or Type II standard.

The EXM-17 supports both SRAM (Static Random Access Memory) and Flash EPROM (Erasable Programmable Read Only Memory) types of memory cards. These different memory card technologies require different formatting methods, which are described later in this chapter. Once formatted, the memory card is accessed directly using standard DOS commands. The default DOS drive for the EXM-17 is the **lowest unused drive letter, usually drive D:**. For systems with no hard drive installed, the EXM-17 becomes the C: drive.

4.2.1 Static Random Access Memory (SRAM) Cards

An SRAM card can be written and read at much higher speeds than either a hard drive or a floppy disk. Data can be erased or replaced by simply re-writing with new data. SRAM cards are therefore suitable for storing data or programs that need to be changed frequently.

Each card contains a small battery to maintain the data stored in it while the system power is off or the card is removed from the computer. The life of the card battery is likely to be several years before needing user replacement. Refer to the card manufacturer's documentation for details of battery replacement.

4.2.2 Flash EPROM Cards

Flash EPROM memory cards operate slower than an SRAM memory card, but Flash EPROM has the advantage of non-volatile storage. This means that data cannot be replaced by simply re-writing with new data. Instead the card has to be erased first.

Flash cards are more suitable for storing data or programs that do not need to be changed frequently. Flash cards do not contain batteries.

4.2.3 Which Memory Card To Use?

Follow these general rules if unsure about the type of memory card best for the system in use:

If the memory card is to be used like a floppy disk for storage of frequently changing data or programs, then use an SRAM memory card.

If data or programs need to be written once but read many times before being re-written or erased, then use a Flash memory card.

4.3 Formatting Memory Cards

SRAM and Flash cards must be formatted before use. Formatting involves erasing the card, initializing it, defining logical partitions, and in some cases loading data files onto it. A set of utilities, referred to as TC commands, are provided for formatting and loading data onto memory cards. Memory cards use a Card Information Structure (CIS) to store the formatting information. The CIS is read each time a TC command performs an operation on the card, thereby ensuring the operation is performed within the physical and logical parameters of the card. For more information about CIS, refer to Section 4.4.1.

Because there are so many different kinds of SRAM and Flash memory cards that can be configured in different ways, a number of default configuration files are provided with the EXM-17 to help correctly format the card and build its CIS information. Configuration files may be specified as command line options to the TC commands. Refer to Section 4.4.2 for more details on configuration files.

The basics of using TC commands such as TCFORMAT are described here in Chapter 4. For detailed information about every option available with each TC command, refer to Appendix A. For more information about using keywords to build customized configuration files, refer to Appendix B.

4.3.1 Formatting SRAM Cards

SRAM cards must be **formatted** before use. The `TCFORMAT` utility (described in detail in Appendix A) is provided for this purpose and should be used to format a new SRAM card or re-format (i.e. erase) an SRAM card with old data stored on it.

If CIS already exists on the card, specifying `TCFORMAT` with no options will format the SRAM card properly, as in the example below:

```
TCFORMAT D:
```

Once an SRAM card is formatted, use it like a floppy disk. Use DOS commands and application programs to read and write data to the card.

If the `TCFORMAT` command fails (i.e., error messages display) this generally implies that no CIS exists on the card. Try again using specific or generic configuration information, depending on if you know the part number for the card. Alternatively, you may build a CIS for the card using the `TCINIT` command. Refer to Section 4.4 for more information.

To define multiple partitions, refer to Section 4.5

To add system files to make the card bootable, refer to Section 4.6.

4.3.2 Formatting Flash EPROM Cards

Flash cards must be **erased and formatted** before use. To erase a Flash card, insert the card in the EXM-17 and enter the following command:

```
TCERASE D:
```

If the `TCERASE` command fails (i.e., error messages display) this generally implies that no CIS exists on the card. Try again using specific or generic configuration information, depending on if you know the part number for the card. Refer to Section 4.4.

`TCERASE` may in some cases erase the CIS. Refer to Appendix A for details.

After TCERASE completes successfully, the card is ready to format and load. If TCERASE has erased the CIS, or if this is a new card, use TCINIT to build a CIS. Refer to Appendix A for more information about TCINIT.

Next, use the TCXCOPY command to perform the following functions:

- Automatically format and initialize blank cards in a DOS-compatible format.
- Copy files by name, by wild-card match, and by searching subdirectories.
- Create subdirectories while copying.
- Append additional files to a Flash card that has previously been prepared using TCXCOPY.

TCXCOPY supports most of the options provided by the DOS XCOPY command with a similar syntax.

Set up a directory on the hard drive to hold the files to be copied to the PCMCIA card. Then type the following command:

```
TCXCOPY C:\[path]\*.* D:
```

This would initialize the card and copy all the files in the directory to the PCMCIA card.

If the TCXCOPY command fails (i.e., error messages display) this generally implies that no CIS exists on the card. Try again using specific or generic configuration information, depending on if you know the part number for the card. Or you can build CIS for the card using TCINIT. Refer to Section 4.4.

Note that since TCXCOPY uses an unusual method for putting files on the Flash card, the DOS DIR command indicates the card always has "0 bytes free" even when space remains.

After a Flash card has been initialized with TCXCOPY it is completely compatible with DOS for *read-only* operations. Users cannot re-write a file or delete it without first erasing the whole card. To add more files to a Flash card prepared with TCXCOPY, use TCXCOPY again.

For more information about TCXCOPY, refer to Appendix A.

4.4 Specifying Memory Card Information

In order for the PCMCIA utilities to correctly erase, initialize and format PCMCIA memory cards, physical and logical information about the card must be specified.

Specifying the card type and size information (or the card's manufacturer and part number) gives the software the **physical** data, such as size, type, and access speed, to properly program the card. In addition, you can indicate how to **logically** store data on the memory card. The logical information includes the file system to use on the card, the error detection method, and the number of logical partitions (drives) to divide the card into.

There are several methods used to specify memory card information. This gives the user flexibility to override standard configurations with customized setups.

Memory cards maintain a Card Information Structure (CIS) that records the memory card's physical and logical information. Refer to Section 4.4.1 for more information about CIS, and refer to Section 4.4.5 for information about placing CIS onto a card using the `TCINIT` command.

Cards without CIS can still be properly accessed by the TC commands if card information is specified from the command line using options. The most important options are the `-CONFIG` and `-CARD` options. Use the `-CARD` option to specify the card's manufacturer and part number, which calls up a predefined configuration file containing the physical layout of the card. Use the `-CONFIG` option to specify a configuration file that contains the card's logical layout.

Note that building and inserting CIS on the memory card ensures that all other TC commands work properly, so CIS is highly recommended.

In the worst case where there is no CIS, and the manufacturer part number is unknown, specify the `-TYPE` and `-SIZE` from the command line. This tells the TC command that it is using either SRAM or Flash, and the size of the card, and is enough to perform the desired function. Refer to Section 4.4.2

4.4.1 Card Information Structure (CIS)

The CIS is a data structure that resides on the memory card and contains the card's physical and logical attributes. TC commands use CIS information when accessing the card to improve efficiency. The TCINIT command is used to store CIS information on the card. Refer to Section 4.4.5.

Certain PCMCIA cards contain attribute memory, a protected memory space that records the CIS. When attribute memory space is specified, `TCINIT` attempts to save CIS in this area. If the card doesn't have attribute memory, `TCINIT` stores this information in the first sector on the card. Consult the card's manufacturer specifications for information on whether or not the card has attribute memory space. Cards without attribute memory lose their CIS when erased.

4.4.2 Configuration Files

Two types of configuration files are supplied with the EXM-17 software: configuration files that specify the *physical* layout of specific memory cards from various manufacturers, and configuration files that describe the desired *logical* partitioning of the memory card.

Model-specific physical configuration files for a particular manufacturer are supplied with the EXM-17 software in a directory by the manufacturer's name. TC commands automatically use the appropriate configuration file when you specify the card with the `-CARD` option.

If there is no manufacturer's configuration file, users can specify `-TYPE` and `-SIZE` at the command line when using TC commands such as `TCFORMAT`. These options require that the card type (RAM or Flash) and size, expressed in hexadecimal values, be known. Refer to Section 4.4.5 for more details.

Logical partitioning configuration files for a variety of logical configurations may be found in the top-level EXM-17 directory. These files are named `TC*.CFG`, where `*` represents 0 or more letters.

The logical configuration file to be used by the TC command is specified by the -CONFIG option. If the -CONFIG option is not specified, the TC command defaults to the TC.CFG configuration file, which is the simplest of the predefined files. Note that TC.CFG causes the default file system to be selected when formatting the card. The default file system for Flash is the Flash File System (FFS). Since the EXM-17 software does not include drivers for FFS, you must either modify TC.CFG or specify one of the other configuration files described in the next section.

For example, to use the TC_CD.CFG configuration file supplied with the EXM-17 software, type the following:

```
TCFORMAT -CONFIG TC_CD.CFG D:
```


You may also create your own configuration files for cards and configurations that are not supplied with the EXM-17 software. Configuration files can be edited or created with any DOS text editor. Refer to Appendix B for details on building your own configuration files.

If you edit the TC_CD.CFG file and save it as CUSTOM.CFG or a similar name, use it as the configuration file by typing the following:

```
TCFORMAT -CONFIG CUSTOM.CFG D:
```

4.4.3 Pre-Defined Logical Configuration Files

Since partitioning can be complicated, sample configuration files for a variety of common logical configurations are provided with the EXM-17 software. These files are explained below. Note that some of these configuration files specify the Flash File System (FFS) as the file system supported by the partition. Since the EXM-17 software does not include drivers for FFS, you should either not use these specific configuration files, or modify them to use the DOS FAT file system instead.

Also note that when formatting a Flash card, the TCFORMAT command places the FFS on the card by default. You must therefore use the -CONFIG TC_CD.CFG option with TCFORMAT to ensure that the card is formatted with the DOS FAT file system.

TC.CFG

This file specifies NO partition information, uses an auto-detection scheme instead of specifying the card type, and uses the default file system. This file can be used for initializing both SRAM and Flash cards.

TC_C.CFG

This file specifies NO partition information, and uses a very basic card file with the default file system. This file is useful for initializing both SRAM and Flash cards.

TC_CF.CFG

This file specifies CIS and creates one FFS partition. This is useful for simple initialization of one-partition Flash cards.



TC_CDD.CFG

This file specifies CIS and partitions the card into two FAT partitions - one being 512K, the other taking up the remainder of the card space. Two drive letters are assigned.

TC_CFD.CFG

This file specifies CIS and creates one FFS partition of 512K and one FAT partition taking up the remainder of the card space.

TC_CFF.CFG

This file specifies CIS and partitions the disk into two FFS partitions, one being 512K and the other taking up the remainder of the card.

TC_CD.CFG

This file specifies CIS with one FAT partition on the card.

TC_CDF.CFG

This file specifies creating one FAT partition and one FFS partition. The card boots from the FFS and should only be used by experienced users.

4.4.4 Specifying Part Numbers

In the absence of CIS on the PCMCIA card, specifying the memory card's manufacturer and part number ensures that the EXM-17 software takes into account all physical characteristics of the card when using TC commands. The physical characteristics include the card size, type of memory, access speed, bus width and whether the card has attribute memory.

All TC commands can specify the card's manufacturer and part number via the -CARD option, which has the following format:

```
-CARD MANUFACTURER/CODE
```

In Appendix C a table lists the memory card codes currently supported by the EXM-17 software. Look up the manufacturer and the exact part number, then look across to the **Code To Use**. This code indicates the part number of the card.

To indicate to TCFORMAT a Mitsubishi SRAM card with part number MF35M1-L2DAT01, use the following command:

```
TCFORMAT -CARD MITSUBISHI/MF35M1 D:
```

To use TCERASE on an Intel 4MB Flash card, type the following:

```
TCERASE -CARD INTEL/IMC004 D:
```

4.4.5 Specifying Generic Information

If the part number of the card is unknown or if the card is not listed in Appendix C, specify the card type and size using the -TYPE and -SIZE options. These two options provide the minimum information to program a memory card. Be certain that the card is a PCMCIA (or JEIDA 4.1) standard SRAM or Flash EPROM card before attempting to use it.

All TC commands (except TCINIT) support the -TYPE and -SIZE options in the following format:

```
TCcommand -TYPE CARD-TYPE -SIZE CARD-SIZE
```

CARD-TYPE is Flash or RAM and CARD-SIZE is the hexadecimal representation of the card size in bytes. For example:

0x20000 for 128KB
0x40000 for 256KB
0x80000 for 512KB
0x100000 for 1MB
0x200000 for 2MB
0x400000 for 4MB
etc.

To erase a 512KB SRAM card without specifying the card's manufacturer and part number, use the following command:

```
TCERASE -TYPE RAM -SIZE 0X80000 D:
```

4.4.6 Placing CIS onto Memory Cards

To place CIS onto memory cards, use the TCINIT command. Specify the card's physical characteristics by including a -CARD option. Either use one of the default configuration files or specify the card's logical layout via the -CONFIG option. You can either use an existing .CFG file or create one using predefined files as a template.

For example, to put CIS on an Intel 1MB Flash card (part number iMC001FLKA) in drive D: using the pre-defined configuration file TC_CD.CFG, type:

```
TCINIT -CARD INTEL/IMC001 -CONFIG TC_CD.CFG D:
```

The card is initialized using the physical data specified by the manufacturer and part number, then logically configured using the .CFG file parameters.

If the memory card is not specifically recommended but is compatible with PCMCIA standards, provide the physical data about the card in the same configuration file that contains the logical layout. At a minimum, specify the card type and size. Assuming all this information is in a configuration file called TEST.CFG and the card is drive D:, run TCINIT with just the -CONFIG option:

```
TCINIT -CONFIG TEST.CFG D:
```

Subsequently, other TC commands examine the CIS on the card to properly program and read from it. The information stays on the card until the card is erased using TCERASE or initialized with new information via TCINIT.

4.5 Formatting Multi-Partition Memory Cards

This section provides instructions for formatting a Flash or SRAM card with more than one logical partition.

4.5.1 Overview

A memory card can be formatted using the TCINIT command to have more than one logical partition, just as hard disks are formatted to have more than one logical drive. In fact, each partition on the memory card is treated as a separate DOS drive. For example, with an SRAM card in drive D with three partitions, the first partition is accessed as drive D, the second is drive E and the third is drive F.

4.5.2 Installing the Device Driver for Multi-partitions

The first step in working with multi-partition cards is to set up the PCICCARD.SYS driver to run in multi-partition mode. To install PCICCARD.SYS in multi-partition mode, add the /N switch followed by the number of desired partitions on the appropriate line in CONFIG.SYS.

For example, to install PCICCARD.SYS to support two logical partitions, specify /N:2.

```
DEVICE=C:\PCICCARD\PCICCARD.SYS /N:2
```

Set the LASTDRIVE keyword in CONFIG.SYS file to be a drive equal to or higher than the total number of drives installed.

With the hard disk formatted to contain drive C, after installing PCICCARD.SYS as in the above example, when an SRAM card formatted with 2 partitions is inserted the first partition is drive D and the second drive E. The LASTDRIVE specification should be:

```
LASTDRIVE=E
```

4.5.4 Formatting a Two-Partition SRAM Card

Use this example to prepare a two-partition SRAM card. First, install PCICCARD.SYS with the /N:2 switch in the CONFIG.SYS file. Assume that after rebooting, PCICCARD.SYS reports that drives E and F are now installed on the

system. Assuming a 512K SRAM card, prepare a configuration file that has information similar to the file below.

```
# Physical characteristics
CARDTYPE: RAM
CARDSIZE: 512K

# This information should reflect
# the actual card
PHYSLAYOUT:
      REG-SPACE: NONE
      COMMON-SPACE:
      DEVICE-TYPE: RAM

# Must store CIS on card.
CIS: FULL
CIS-RESERVE: 1
CIS-CHECKSUM: YES


# 2 partitions on card
PARTITIONS:2

# 1st partition has 200K
PARTITION:
      SIZE: 200K
      FILE-SYSTEM: DOS
      ERROR-DETECT: CRC INLINE

# 2nd partition has remainder
# of card.
PARTITION:
      FILE-SYSTEM: DOS
      ERROR-DETECT: CHECKSUM TABLE
```

The above file specifies two logical partitions; the first partition is 200K in size, has DOS FAT format, and uses CRC error detection stored at the end of each sector. The second partition has 312K (since no size is specified, assume the rest of the card). It also has FAT for file system, and uses the CHECKSUM error detection method stored in a table at the end of the partition.

Assuming that this configuration file has the name `RAM.CFG`, run `TCINIT` with the `-CONFIG` option to specify the above file.

```
TCINIT -CONFIG RAM.CFG D:
```

TCINIT records the above CIS in the first sector of the card so that subsequent card commands find the CIS and use it.

Note that the drive specified with TCINIT must be the first drive identified by the PCICCARD.SYS driver.

TCINIT places CIS data for the entire card so only run this command once.

Finally, run TCFORMAT on each partition of the card to put the appropriate file system information in each partition. TCFORMAT operates on the individual partitions so run it once for each logical drive.

```
TCFORMAT D:
TCFORMAT E:
```


TCFORMAT at this point can obtain complete card and partition information from the CIS on the card to format each partition properly.

Use standard DOS commands and applications for reading and writing to the two-partition SRAM card. Only use this card on another system if it supports CIS and multi-partition mode according to the PCMCIA/JEIDA standards.

Note that in the above example, if installing PCICCARD.SYS to handle two partitions, but only preparing a card with one partition, attempting to access the second drive (drive E) results in a "Drive Not Ready" error, similar to accessing a floppy drive without a disk in it.

4.5.5 Formatting a Two-Partition Flash Card

In this example, prepare a two-partition Flash card so that one partition has the Flash File System while the other has the DOS File Allocation Table for file system. First, install PCICCARD.SYS with the /N:2 switch in CONFIG.SYS file. Assume that after installation, PCICCARD.SYS reports drives D and E are now installed.

To read and write to the Flash file system partition on the Flash card, the following device drivers must be in the CONFIG.SYS file: PCFLASH.SYS, IFS.SYS and FEFS.SYS. Refer to the Microsoft Flash File System (FFS) documentation for more information about installing the FFS. The EXM-17 software does not include these drivers, but can still format a Flash partition.

For an Intel 1MB Flash card, a configuration file might include the following:

```
# Must store full CIS on card
CIS: FULL
CIS-RESERVE: 1
CIS-CHECKSUM: YES

# 2 partitions
PARTITIONS:2

# 1st partition is FAT
PARTITION:
      SIZE: 600K
      FILE-SYSTEM: DOS
      ERROR-DETECT: NONE

# 2nd partition is FFS
PARTITION:
      FILE-SYSTEM: FLASH
      ERROR-DETECT: NONE
```



The first partition is 600K in size, has DOS FAT format, and has no error detection. Note that the only way to format a Flash card to have a DOS FAT partition is to use TCXCOPY. Therefore set error detection to NONE for the DOS FAT partition of the Flash card.

The second partition has 424K (since no size is specified, assume the rest of the card). Error detection is not required on the card since it has the Flash File System.

Note there is no specification of information regarding the physical properties of the card. Since the Intel 1MB Flash card is one of the supported cards, specify the manufacturer and part number via the `-CARD` option. Assuming that this configuration file has the name `FLASH.CFG`, run `TCINIT` with the `-CONFIG` option to specify the above file.

```
TCINIT -CARD INTEL/IMC001 -CONFIG FLASH.CFG  D:
```

`TCINIT` records the above CIS in the first sector of the card so that subsequent card commands can examine the CIS to properly operate the card. Note that the drive specified with `TCINIT` must be the first drive identified by the `PCICCARD.SYS` driver.

After placing CIS on the card, format the card to set up the appropriate file systems.

To put DOS FAT on the first partition use TCXCOPY, which simultaneously formats the partition and copies files to it. To format and copy all .DAT files to the first partition:

```
TCXCOPY *.DAT D:
```

Don't specify additional information about the card since TCXCOPY can obtain information from the CIS on the card. Run TCFORMAT on the second partition to put FFS on it.

```
TCFORMAT E:
```

Do not specify more information since TCFORMAT can obtain complete card and partition information from the CIS on the card.

Read files on drive D as normal; to copy more files to this partition, use TCXCOPY.

Note that certain TC commands work on the entire physical card as opposed to individual partitions. These commands are TCERASE, TCINIT and TCREAD. For example, using TCERASE on a 1MB Flash card that has 2 partitions with 512KB for each partition, results in the entire 1MB card being erased.

4.6 Formatting Bootable Memory Cards

Some systems may require the EXM-17 to be a bootable A: drive with a system-formatted memory card. This allows the computer to boot without using either the hard disk drive or the floppy disk drive.

It is not possible to boot DOS from a drive other than A: or C:, so when the system finds a system-formatted memory card in the EXM-17 it will boot from it as though it were the DOS drive A:, even if a floppy disk drive is attached. This means that after bootup the floppy disk drive is not accessible and the EXM-17 becomes drive A:.

Refer to Section 4.7 for information about how to re-enable the floppy drive by using the REMAP.SYS command.

4.6.1 Making a Bootable SRAM Card

To make an SRAM card bootable ensure that it contains the DOS system files. To do this the card must be prepared as follows and then the appropriate files such as AUTOEXEC.BAT and CONFIG.SYS should be copied to it.

Use the TCFORMAT command and the -SYS option by typing the following from the hard drive's root directory:

```
TCFORMAT -SYS D:
```

If a configuration file for the card exists include it with the appropriate options. When using the -SYS option, which is equivalent to the /S switch of the DOS FORMAT command, it is mandatory that the current drive is the boot drive.

Once the SRAM card is formatted use the DOS COPY command to place any remaining files on it.



4.6.2 Making a Bootable Flash EPROM Card

Use an erased Flash card and the TCXCOPY command with the -SYS option. For example to format a Mitsubishi 512KB Flash card type the following from the hard drive's root directory:

```
TCERASE D:
TCXCOPY -SYS D:
```

If a configuration file for the card exists, include it in the TCXCOPY line with the appropriate options.

4.6.3 Bootable Card Contents

In addition to the normal DOS system files, the boot disk requires appropriate AUTOEXEC.BAT and CONFIG.SYS files. These can be copied to an SRAM card with the COPY command, or to a Flash card with the TCXCOPY command. For example, type the following:

```
TCXCOPY C:\AUTOEXEC.BAT C:\CONFIG.SYS D:
```

to place AUTOEXEC.BAT and CONFIG.SYS files on a Flash card.

4.7 Using REMAP.SYS

The floppy disk drive can be made accessible after bootup from a memory card by use of the REMAP.SYS device driver program. When installed in the CONFIG.SYS file on the memory card, REMAP.SYS re-enables the floppy disk as drive A:. This assigns the EXM-17 to the lowest available drive letter (C: in this case). For example, add the following line to the end of the CONFIG.SYS file:

```
DEVICE=A:\PCICCARD\REMAP.SYS
```


and copy the REMAP.SYS file to the root directory of the memory card. It is important that the DEVICE=REMAP.SYS line appears in the CONFIG.SYS file, *after* all other device statements that reference A: as the memory card.

5. Restrictions

5.1 Overview

Due to the limited BIOS addressing space, it is probable that the EXM-17 could "map" PCMCIA plug-in hardware over existing EMC/EPC hardware, creating an address conflict.

As of this writing, the Award software enclosed with the EXM-17 is not updated to be fully compatible with MS-DOS 6.0. For this reason, SRAM cards cannot be made the bootable device when using MS-DOS 6.0 with the EXM-17. When this changes, RadiSys will contact impacted customers.

SunDisk PCMCIA cards can be used with the EXM-17 but ***cannot*** be the bootable device.

The EXM-17 is designed primarily as an EMC product. It functions in the EPC-5 and EPC-7 as long as certain constraints (listed below) are noted. The EXM-17 has such difficult configuration issues and limited capability on the EPC-2, EPC-3, EPC-4 and EPC-6 that it is regarded as non-supportable on those systems.

5.2 VME Systems

If using the EXM-17 in a VME system, typically there is no access to the E-page of memory (0x0E0000 - 0x0EFFFF). This implies that EPC-5 systems can boot from the EXM-17 only if the ROM is configured in the C-page. (Refer to Section 2). This restriction does not apply to the EMC CPUs (EPC-21, EPC-22, and EPC-23).

5.3 EXM-1

Due to memory conflicts, it is not possible to boot from an EXM-1 Ethernet Controller with an EXM-17 installed in the system.

5.4 EXM-10/10A

VME computers with an EXM-10 installed cannot be booted from the EXM-17 unless the EXM-10 is mapped within the C-page. EXM-10A cards using WD8003 mode cannot be booted unless the card is mapped within the C-page. In NE2000 mode, the EXM-10A uses the programmed I/O and there is no problem.

5.5 EXM-17 as Serial Port or Modem

If using the EXM-17 as a modem or serial I/O port, the `PCICCARD.SYS` device driver automatically detects the first available COM port for usage.

5.6 EPC-7 Conflicts

In order to boot from an EXM-17 in an EPC-7 system, the BIOS must be version 3.05.04 or greater, and the `EMAP.SYS` driver must be installed.

Booting from the EXM-17 on an EPC-7 requires the following:

1. Configure the EXM-17 EPROM at address E000. (*see Table 2*)
2. Add `PCICCARD.SYS` to `CONFIG.SYS`. (*see page 9*)
3. Add `REMAP.SYS` to allow access to the floppy drive. (*see page 9*)
4. Add `EMAP.SYS`. (*see page 9*)

Do not use `DEVICEHIGH` for these drivers.

Make sure there are no intervening device drivers and that these drivers appear at *the beginning* of `CONFIG.SYS`. Whenever new software is loaded on the system, check to be sure the order of these drivers does not change. In addition, note that certain memory management programs may move these drivers during the optimization process. Be sure to move them back to the order shown above.

Execution of `PCICCARD.SYS` and `EMAP.SYS` results in a 64K memory window at D000h - DFFFh and replacement of the ROM from E000h to C800h.

NOTE: If SCSI BIOS is enabled, it should not conflict with regions used by the EXM-17. When in the Fixed Disk Menu, ignore the SCSI formatting message. Note that you cannot boot from a Sun disk device.

6. Support and Service

In North America

Technical Support

RadiSys maintains a technical support phone line at (503) 646-1800 that is staffed weekdays (except holidays) between 8 AM and 5 PM Pacific time. If you have a problem outside these hours, you can leave a message on voice-mail using the same phone number. Alternately, you can request help via FAX addressed to RadiSys Technical Support. The RadiSys FAX number is (503) 646-1850. If you are sending a FAX, please include information on both the hardware and software being used and a detailed description of the problem, specifically how the problem can be reproduced. We will respond by phone or FAX by the next business day.

Technical Support Services are designed for customers who have purchased their products from RadiSys or a sales representative. If your RadiSys product is part of a piece of OEM equipment, or was integrated by someone else as part of a system, support will be better provided by the OEM or system vendor that did the integration and understands the final product and environment.

Repair Services

Factory Repair Service is provided for all RadiSys products. Standard service for all RadiSys products covers factory repair with customers paying shipping to the factory and RadiSys paying for return shipment. Overnight return shipment is available at customer expense. Normal turn-around time for repair and re-certification is five working days.

Quick Exchange services (immediate shipment of a loaner unit while the failed product is being repaired) or other extra-cost services can be arranged, but need to be negotiated in advance to allow RadiSys to pool the correct product configurations. RadiSys does not maintain a general "loaner" pool: units are available only for customers that have negotiated this service in advance.

RadiSys does not provide a fixed-price "swap-out" repair service, as customers have indicated that issues of serial number tracking and version control make it more convenient to receive their original products back after repair.

Warranty Repairs

Products under warranty (see warranty information in the front of this manual) will have manufacturing defects repaired at no charge. Products sent in for warranty repair that have no faults will be subject to a recertification charge. Extended Warranties are available and can be purchased at a standard price for any product still under warranty. RadiSys will gladly quote prices for Extended Warranties on products whose warranties have lapsed; contact the factory if this applies.

Customer induced damage (resulting from misuse, abuse, or exceeding the product specifications) is not covered by the standard product warranty.

Non-Warranty Services

There are several classes of non-warranty service. These include repair of customer induced problems, repairs of failures for products outside the warranty period, recertification (functional testing) of a product either in or out of warranty, and procurement of spare parts.

All non-warranty repairs are subject to service charges. RadiSys has determined that pricing repairs based on time and materials is more cost-effective for the customer than a flat-rate repair charge. When product is received, it will be analyzed and, if appropriate, a cost estimate will be communicated to the customer for authorization. After the customer authorizes the repair and billing arrangements have been made, the product will be repaired and returned to the customer.

A recertification service is provided for products either in or out of warranty. This service will verify correct operation of a product by inspection and testing of the product with standard manufacturing tests. There is a product-dependent charge for recertification.

There are only a few components that are generally considered field-repairable, but, because RadiSys understands that some customers want or need the option of repairing their own equipment, all components are available in a spares program. There is a minimum billing charge associated with this program.

Arranging Service

To schedule service for a product, please call RadiSys Technical Support directly at (503) 646-1800. Have the product model and serial numbers available, along with a description of the problem. A Technical Support representative will issue a Returned Materials Authorization (RMA) number, a code number by which we track the product while it is being processed. Once you have received the RMA number, follow the instructions of the Technical Support representative and return the product to us, freight prepaid, with the RMA number clearly marked on the exterior of the package. If possible re-use the original shipping containers and packaging. In any case, be sure you follow good ESD-control practices when handling the product, and ensure that anti-static bags and packing materials with adequate padding and shock-absorbing properties are used.

Ship the product, freight prepaid, to

Product Service Center
RadiSys Corporation
15025 SW Koll Parkway
Beaverton, Oregon 97006-6902

When shipping the product, include the following information: return address, contact names and phone numbers in purchasing and engineering, and a description of the suspected problem. Any ancillary information that might be helpful with the debugging process will be appreciated.

Other Countries

Contact the sales organization from which you purchased your RadiSys product for service and support.

Appendix A. TC Commands Reference

A A

This section gives complete information on the syntax of each TC command.

Title	Page
TCERASE	A-3
TCFORMAT	A-6
TCINIT	A-9
TCPROG	A-11
TCREAD	A-16
TCXCOPY	A-19

General Information

All the TC commands share a common form:

```
COMMAND-NAME OPTIONS PARAMETERS
```

All fields, unless specified to the contrary, are not case-sensitive. `COMMAND-NAME` specifies the name of the TC command, such as `TCFORMAT`. `OPTIONS` allow control of the default operation of the command. `OPTIONS` always begin with a dash ("`-`"), and must be separated from surrounding information by one or more spaces. To enclose a special character within an option argument, enclose the entire argument in quotes.

`PARAMETERS` provide the substantive information to the command: what files to copy, what drive to operate upon, and so forth. `PARAMETERS` begin immediately following the options field. The first parameter must not begin with a dash; otherwise, the program will attempt to interpret the parameter as an option.

A double-dash ("--") always marks the end of the options. For example, to copy the file "-TEST-.DAT" to a Flash card, enter the following command:

```
TCXCOPY -- -TEST-.DAT D:
```

A A

Specifying Default Options

Each TC command option has a default value that is used if the option is not specified. It is sometimes necessary to change the default options assumed by the TC commands. When run, each command examines an environment variable for default options before checking the information from the command line. Each command has its own variable, whose name is formed by attaching the word "FLAGS" to the name of the command. For example, the TCXCOPY command looks for an environment variable named TCXCOPYFLAGS.

Options specified as defaults, whether in the program itself or in the environment variable, can always be overridden on the command line.

NOTE:

It is essential that there be no blanks or tabs between the environment variable and the equals sign ("="). Under some versions of DOS, the trailing spaces prevent the TC command from matching the flags.

TCERASE

Flash cards must be erased before new data can be stored on them. The TCERASE command is provided for this purpose and should be used to re-format (i.e. erase) a Flash card with old data stored on it.

If the card has attribute memory space, TCERASE does not erase this area, leaving the CIS information intact. If CIS is stored anywhere other than in the attribute memory space, it is lost.



Command Syntax

To execute TCERASE from the command line type the following:

```
TCERASE [OPTIONS] DRIVE:
```

DRIVE: specifies the memory card drive containing the card to be erased. As usual, it should be specified as a letter, usually D, followed by a colon, ":".

Options

The OPTIONS control the programming operation. The valid options are:

-?

TCERASE produces a brief command summary as follows:

```
TCERASE    -F-USAGE:    -[BUFFER#    DEBUG    PROGRESS
OVERRIDE SIZE## STATS DF* F* CONFIG* CARD* TYPE* V]
<DRIVE>
```

-BUFFER BUFFER-SIZE

Specifies the size of the memory buffer, in bytes, to be used in programming the card. By default, a buffer of 32,768 bytes is used.

-CARD MANUFACTURER/PART-NUMBER-CODE

For example, to use TCERASE on an Intel 4MB Flash card, type the following:

```
TCERASE -CARD INTEL/IMC004 D:
```

A A

If the card is supported by the EXM-17 software, use this option to specify the card's manufacturer and part number. TCERASE uses this information to look up the physical characteristics. Refer to Appendix C for a list of supported cards.

If the card doesn't appear in the supported list but it is compatible with PCMCIA/JEIDA standards, use the -TYPE and -SIZE options to specify the generic card information.

-CONFIG FILE NAME

Use the -CONFIG command option for optimized use. For example to use the configuration file TC_CD.CFG, type the following:

```
TCERASE -CONFIG C:\[path]\TC_CD.CFG D:
```

Type the path and filename for the configuration file exactly as it exists, or else the command will not find it and will use the default file TC.CFG instead.

-DEBUG

Requests that TCERASE operate in debugging mode. Considerable output is produced regarding the operation being performed.

-OVERRIDE

Requests that the user-supplied card type and size information override the information obtained from the card by TCERASE, no matter what the source. Do not use this option normally.

-NOOVERRIDE

Requests no override of TCERASE. This is the default.

-PROGRESS

Requests that TCERASE give progress reports while working. This is the default.

-NOPROGRESS

Requests that TCERASE not give any progress reports while working.

-SIZE CARD-SIZE

Specifies the size of the card, in bytes. If the card has the size stored as part of the CIS, or if the `-CARD` option is used, there is no need to specify the card size. The simplest way to specify card sizes is in hexadecimal. Some useful values are:

128K	`0x20000`	256K	`0x40000`
512K	`0x80000`	1M	`0x100000`
2M	`0x200000`	4M	`0x400000`

`-STATS`

Requests that `TCERASE` display statistics on its efficiency.

`-NOSTATS`

Requests that `TCERASE` display no statistics. This is the default.

`-TYPE` `CARD-TYPE`

Specifies the type of card to erase. If the `-CARD` option is already specified, or if the card has card type stored in CIS, don't use this option. Supported card types are `RAM` and `FLASH`.

`-V`

Requests verbose operation. If `-V` is specified, `TCERASE` reports about its progress.

Environment Variables

`TCERASE`'s default options are modified by setting up a variable in the DOS environment named `TCERASEFLAGS`. Any options contained in `TCERASEFLAGS` are scanned before the options from the command line. For example, to default to use of the Intel 1MB Flash card, add this line to the `AUTOEXEC.BAT` file:

```
SET TCERASEFLAGS=-CARD INTEL/IMC001
```



A A

TCFORMAT

The TCFORMAT command should be used to format a new SRAM card or re-format (i.e. erase) an SRAM card with old data stored on it.

The simplest way to use the command is as follows:

```
TCFORMAT D:
```

Additional options that are used are listed below.

Command Syntax

To execute TCFORMAT from the command line, type the following:

```
TCFORMAT [OPTIONS] DRIVE:
```

DRIVE: specifies the memory card drive containing the card to be initialized. As usual, it should be specified as a letter, usually D, followed by a colon, ":".

Options

The OPTIONS control the programming operation. The valid options are:

-?

TCFORMAT produces a brief command summary as shown below:

```
TCFORMAT -F-USAGE: -[DF* F* CONFIG* CARD* TYPE* SYS
W V] <DRIVE>
```

-CARD MANUFACTURER/PART-NUMBER-CODE

For example, to use TCFORMAT on an Intel 4MB Flash card, type the following:

```
TCFORMAT -CARD INTEL/IMC004 D:
```

If the card is supported by the EXM-17 software, use this option to specify the card's manufacturer and part number. TCFORMAT uses this information to look up the physical characteristics. Refer to Appendix C for a list of supported cards.

If the card doesn't appear in the supported list but it is compatible with the PCMCIA/JEIDA standard, use the -TYPE and -SIZE options to specify generic card information.

-CONFIG FILE-NAME

Use the -CONFIG command option for optimized use. For example to use the configuration file TC_CD.CFG, type the following:

```
TCFORMAT -CONFIG C:\[path]\TC_CD.CFG D:
```

Type the path and filename for the configuration file exactly as they exist, or else the command will not find it and will use the default file instead.

-V

Requests detailed information while in progress.

-NOV

Suppress verbose output. This is the default.

-W

Requests TCFORMAT to issue interactive prompts. This is the default.

-NOW

Requests no interactive prompting.

-TYPE CARD-TYPE

Specifies the type of card to format. If the -CARD option is already specified, or if the card has CARD-TYPE stored in CIS, don't use this option. Supported card types are RAM and FLASH.

-SYS

Requests TCFORMAT to initialize the card and place DOS system files on the card.





Environment Variables

TCFORMAT's default options can be modified by setting up a variable in the DOS environment named TCFORMATFLAGS. Any options contained in TCFORMATFLAGS are scanned before the options from the command line. For example, to default to the Mitsubishi 2MB SRAM card add the following line to the AUTOEXEC.BAT file:

```
SET TCFORMATFLAGS=-CARD MITSUBIS/MF32M1
```

TCINIT

Use TCINIT to place the Card Information Structure (CIS) on a memory card. If the card has attribute memory, TCINIT stores as much of CIS data as possible in attribute memory. If the card doesn't have attribute memory, TCINIT stores this information in the first sector on the card.



Command Syntax

To execute TCINIT from the command line, type the following:

```
TCINIT [OPTIONS] DRIVE:
```

DRIVE: specifies the memory card drive containing the card to be initialized. As usual, it should be specified as a letter followed by a colon, ":".

Options

The OPTIONS control the programming operation. The valid options are:

-?

TCINIT produces a brief command summary and stops.

-CARD MANUFACTURER/PART-NUMBER-CODE

If the card is one of those supported by the EXM-17 software, use this option to specify the card's manufacturer and part number. TCINIT uses this information to look up the physical characteristics of the card. Refer to Appendix C for a list of supported cards.

If the card doesn't appear in the supported list but it is compatible with PCMCIA/JEIDA standards, add the card's physical data in the same configuration file used to describe the card's logical layout. See Chapter 4 and the -CONFIG option below.



`-CONFIG FILENAME`

Specifies the configuration file to be used by `TCINIT`. The `FILENAME` field for this option may contain a file path.

Typically, the configuration file contains information about the logical layout of the card and how data is stored. If you don't specify a configuration file, `TCINIT` looks for a default file named `TC.CFG` provided as part of the distribution.

`-P`

TCINIT verifies that the selected configuration file is sensible, but does not initialize the card. This is useful when constructing new configuration files. TCINIT requires that a Flash card be blank or freshly erased before it can program CIS on the card. SRAM cards accept CIS without being blank or erased.

TCPROG

Use TCPROG to program a binary image stored in a file to a memory card.

Command Syntax

To execute TCPROG from the command line, type the following:

```
TCPROG [OPTIONS] DRIVE: INPUT-FILE(S)
```

DRIVE: specifies the memory card drive containing the card to be programmed.

The INPUT-FILE(S) field specifies one or more files that supply the data to be used in programming the card. This file may be a binary file, an Intel OMF, or an Intel ASCII-hex file. See section *Programming Cards for Embedded Systems* for more information.



Options

The valid options are:

-?

TCPROG produces a brief command summary.

-BUFFER BUFFER-SIZE

Specifies the size of the memory buffer, in bytes, to be used in programming the card. By default, a buffer of 32,768 bytes is used.

-CARD MANUFACTURER/PART-NUMBER-CODE

If the card is one of the cards supported by the EXM-17 software, use this option to specify the card's manufacturer and part number. TCPROG uses this information to look up the physical characteristics of the card. Refer to Appendix C for a list of supported cards.

If the card doesn't appear on the supported list but is compatible with PCMCIA/JEIDA standards, use the -TYPE and -SIZE options to specify the generic card information.

-CONFIG FILENAME

Specifies the configuration file to be used by TCPROG. The filename field for this option may contain a file path. If a configuration file is not specified, TCPROG looks for a default file named TC.CFG provided as part of the distribution.

-DEBUG

Requests debugging mode. Considerable output is produced.

-NODEBUG

Requests no debugging output. This is the default.

-FILETYPE FILETYPE

Requests that TCPROG use the specified file type when it programs the memory card. Normally, TCPROG determines the file type when it opens the file. Use this switch to override. The valid file types are:

OMF	Intel Absolute Object Module
HEX	Intel ASCII Hex
BIN	binary image (no format)

-OVERRIDE

Requests that the user-supplied card type and size information override the information obtained from the card by TCPROG, no matter what the source. Do not use this option normally.

-NOOVERRIDE

TCPROG checks user input against information stored on the card, prior to programming the card. If there are any conflicts, TCPROG stops with an error message. This is the default.

-PROGRESS

Requests progress reports while working. This is the default.

-NOPROGRESS

Suppresses the progress reports.

-SIZE CARD-SIZE

Specifies the size of the card, in bytes. If the card has the size stored as part of the CIS, or if the -CARD option is used, there is no need to specify the card size.

The simplest way to specify card sizes is in hexadecimal. For reference, some useful values are:

128K	0x20000
256K	0x40000
512K	0x80000
1M	0x100000
2M	0x200000
4M	0x400000

-STATS

Requests that TCPROG display statistics on programming efficiency at the end of a programming cycle.

-NOSTATS

This option tells TCPROG to suppress display of programming statistics. This is the default.

-TYPE CARD-TYPE

Specifies the type of card to program. See the list of generic card types supported later in this section.

If the -CARD option is already specified, or if the card has CARD-TYPE stored in CIS, don't use this option.

-V

Requests verbose operation. If -V is specified, TCPROG produces information about its progress.





-NOV

This option tells TCPROG to suppress "verbose" output. This is the default.

Supported Card Types

Usually the type of card is specified by entering the card's manufacturer and part number code with the -CARD option. However, if the card doesn't appear in the list of supported cards in Appendix C, specify the card using -TYPE and one of the generic types supported below.

FLASH

Use this card type when programming Flash memory cards. TCPROG determines the size of these cards automatically.

RAM

Use this card type when programming SRAM cards with TCPROG. This feature is useful both in test and as a final check that the image files you wish to program are correct. TCPROG can automatically determine the size in this case. For example to specify the correct programming for an Intel Flash card use the following:

```
TCPROG -TYPE FLASH
```

The card type must be written out in full; however, the type can be written in any mix of upper and lower case letters.

Environment Variables

TCPROG's default options can be modified by setting up a variable in the DOS environment named TCPROGFLAGS. Any options contained in TCPROGFLAGS will be scanned before the options from the command line. Thus, to select Flash cards by default and to select a one megabyte default card size, add the following line to the AUTOEXEC.BAT command file:

```
SET TCPROGFLAGS=-TYPE Flash -SIZE 0x100000
```

Programming Cards for Embedded Systems

In addition to its ability to program a card from a card image, TCPROG is able to program cards from Intel OMF or Intel ASCII-hex files. This allows the use of SRAM and Flash cards as general-purpose memory components in embedded systems.

When used in this way, TCPROG has a significant characteristic: it simply programs bytes in the order it encounters them in the input OMF or ASCII-hex files. This works fine for SRAM cards; and in most cases it is entirely adequate for programming Flash cards. Sometimes, though, compilers emit instructions which first cause a byte to be set to a default value, then cause that same byte to be changed. Obviously, this will not work when programming Flash cards.



A A

TCREAD

Use TCREAD to read the image of the card and store it in a binary file.

Command Syntax

To execute TCREAD from the command line type the following:

TCREAD [OPTIONS] DRIVE: OUTPUT-FILE [START] [LENGTH]

DRIVE: specifies the memory card drive containing the card to be read.

The OUTPUT-FILE specifies where the data is to be written.

START and LENGTH, if present, specify the amount of data to be copied from the card. These arguments have two purposes; they allow you to read a portion of the card; and they allow you to specify the size of the card.

When copying file-structured SRAM cards, TCREAD can usually determine the size accurately; but when copying arbitrary data, specify the length of the card. START and LENGTH are both numbers; they can be in decimal format such as 512 or in hexadecimal format such as 0x400.

Options

The [OPTIONS] control the copying operation. The valid options are:

-?

TCREAD produces a brief command summary and stops.

-BUFFER BUFFER-SIZE

Specifies the size of the memory buffer, in bytes, to be used in reading the card. By default, a buffer of 1024 bytes is used. For large copies, on systems with lots of memory, specify a larger buffer size.

-CARD MANUFACTURER/PART-NUMBER-CODE

If the card is one of the cards supported by the EXM-17 software, use this option to specify the supported card's manufacturer and part number. TCREAD uses this information to look up the physical characteristics of the card. Refer to Appendix C for a list of supported cards.

If the card doesn't appear in the supported list but is compatible with PCMCIA/JEIDA standards, specify the card size using -SIZE option below.



-CONFIG FILENAME

Specifies the configuration file to be used by TCREAD. The filename field for this option may contain a file path. If a configuration file isn't specified, TCREAD looks for a default file named TC.CFG provided as part of the distribution.

-SIZE CARD-SIZE

Specifies the size of the card to read. If the card has CIS, or if the -CARD option is used to specify the card's manufacturer and part number, don't use this option. The simplest way to specify card sizes is in hexadecimal. For reference, some useful values are:

128KB	0x20000
256KB	0x40000
512KB	0x80000
1MB	0x100000
2MB	0x200000
4MB	0x400000

-V

Specifies verbose mode. TCREAD produces information about its progress.

-NOV

This option tells TCREAD to suppress "verbose" output. This is the default.



Environment Variables

TCREAD's default options can be modified by setting up a variable in the DOS environment named TCREADFLAGS. Any options contained in TCREADFLAGS are scanned before the options from the command line. For example, to force a larger buffer by default, add the following line to the AUTOEXEC.BAT file:

```
SET TCREADFLAGS=-BUFFER 32768
```

TCXCOPY

Use TCXCOPY to create a DOS-compatible read-only file system on Flash cards and for test purposes on SRAM cards. TCXCOPY's function and syntax are somewhat similar to that of the DOS XCOPY command; hence the name.

Some of TCXCOPY 's notable features include:

- Automatically formats and initializes the card before copying.
- Copies files by name, by wild-card match, and by searching subdirectories.
- Reads a list of files to be copied from an input file.
- Adds files to a card.



Command Syntax

To execute TCXCOPY from the command line type the following:

TCXCOPY [OPTIONS] FILE-DIRECTIVES DRIVE:[PATH]

OPTIONS controls the operation of TCXCOPY. file-directives specifies the files to be copied. DRIVE is the memory card drive; PATH is the output path on the drive. See the following sections for more information on each item.

Options

The options specify the overall operation of TCXCOPY. Valid options are:

-?

TCXCOPY produces a brief command summary and stops.

-BUFFER BUFFER-SIZE

Specifies the size, in bytes, of the memory buffer to be used in programming the card. By default, a buffer of 32,768 bytes is used for Flash and SRAM cards.

-CARD MANUFACTURER/PART-NUMBER-CODE



If the card is one of the cards supported by the EXM-17 software, use this option to specify the card's manufacturer and part number. TCXCOPY uses this information to look up the physical characteristics of the card. Refer to Appendix C for a list of supported cards.

If the card doesn't appear in the list of supported cards, but is compatible with PCMCIA/JEIDA standards, provide the card size and type via the -SIZE and -TYPE options.

-CONFIG FILENAME

Specifies the configuration file to be used by TCXCOPY. The filename field for this option may contain a file path. If a configuration file is not specified, TCXCOPY looks for a default file named TC.CFG provided as part of the distribution.

-DEBUG

Requests that TCXCOPY operate in debugging mode. Considerable output is produced while the operation is performed.

-NODEBUG

Suppresses debug output. This is the default.

-PRECHECK

Requests that TCXCOPY make sure that everything will fit before starting the copy process. This is the default.

-NOPRECHECK

Requests that TCXCOPY disable its normal checks that determine whether the input data will fit on the output card.

-PROGRESS

Requests that TCXCOPY give progress reports while working. This is the default.

-NOPROGRESS

Disable the progress report.

`-OVERRIDE`

Requests that the user-supplied card type and size information override the information obtained from the card by `TCXCOPY`, no matter what the source. Do not use this option normally.

`-NOOVERRIDE`

This is the default. `TCXCOPY` will refuse to operate if the user's data conflicts with information supplied by information embedded in the card.

`-SIZE` `CARD-SIZE`

Specifies the size of the card, in bytes. If the card has CIS, or already uses the `-CARD` option, or if adding more files to the card, don't use this option.

The simplest way to specify card sizes is in hexadecimal. For reference, some useful values are:

128KB	`0x20000`
256KB	`0x40000`
512KB	`0x80000`
1MB	`0x100000`
2M	`0x200000`
4M	`0x400000`

`-SLASH`

Tells `TCXCOPY` to look for a "/" preceding each file-directive option. This is the default.

`-NOSLASH`

Requests that `TCXCOPY` disable its use of "/" to introduce file-directive options. The hyphen (-) is used instead to introduce file-directive options.

`-STATS`





Requests that TCXCOPY display statistics on programming efficiency at the end of a programming cycle.

-NOSTATS

Suppresses statistical display. This is the default.

-SYS

Requests TCXCOPY to initialize the card and place DOS system files on the card.

For Flash cards, this option is equivalent to the -SYS option of TCFORMAT. Note that the command must be invoked from the boot drive.

-TYPE CARD-TYPE

Specifies the type of card being worked with. Supported types are RAM and FLASH. If the card has CIS, or is already using the -CARD option, don't specify this option.

-V

Requests verbose operation. If -V is specified, TCXCOPY produces more output regarding its progress.

-NOV

Suppress verbose output. This is the default.

File-Directives

File-directives specify the files to be copied to the card, in one of the following ways:

- by specifying the name of the file to be copied
- by specifying a wild-card pattern giving the names of files to be copied
- by specifying the name of a file containing more file-directives

Use as many file-directives as will fit on TCXCOPY's command line.

File-directives all have a common form: they consist of a file name optionally followed by one (or more) file-directive-options. Each file-directive-option is written as a forward slash ("/"), followed by a single letter (unless the NOSLASH option is

specified with the TCXCOPY command). The following file-directive options are recognized:

/S

Search subdirectories. If the preceding file specification designates a directory, the directory and all subdirectories are copied to the card. Otherwise, the file specification is split into its directory and file-name portions. The directory and all subdirectories are searched for files that match the file-name; if any are found, they are copied to the card.

/NOS

Suppress subdirectory searches (default).

/E

Copy empty subdirectories. if used with /S, and if the preceding file specification designates a directory or all files in a directory, causes empty subdirectories to be copied. Otherwise has no effect.

/NOE

Do not copy empty subdirectories (default).

/H

Copy hidden files.

/NOH

Don't copy hidden files (default).



A A

/F

The previous file-name designates a file containing a list of file-directives. Each file-directive may be separated from the previous directive by spaces, tabs, or by a new line. If /S, /H, or /E are also specified, they set the defaults for processing the file.

/I

Read a list of file-directives from the standard-input device. If /S, /H, or /E are also specified, they set the defaults for processing the list of directives. The list is terminated by end of file (^Z when entering the list from the console). If this file-directive option is specified, no file specification is present. Any previous options are taken as applying to the previous file-name (if any); subsequent options, up to the next file name, are taken as applying to the /I option.

Note that file-directive options need not be separated from the preceding file specification or from each other. However, a blank space may be used for clarity.

Global File-directive Options

If file-directive options appear after the drive specifier, they implicitly specify default modes for TCXCOPY's operations.

Copying a Single File

A single file can be copied simply by specifying its name. Example:

```
TCXCOPY TMM.DAT D:
```

The file "TMM.DAT" is copied to the card in drive D:.

Copying a Single Directory

The contents of a single directory may be copied simply by specifying the name of the directory. The contents of the current directory may be copied using the special name "." Example:

```
TCXCOPY . D:\MYDIR
```

The directory D:\MYDIR is created, if necessary. Then all files in the current directory are copied to D:\MYDIR. By default, subdirectories are not copied. Use the /S option to include subdirectories while copying.

Copying Multiple Files

Copy a few files by simply listing them. Example:

```
TCXCOPY FILE1.DAT APP.EXE D:
```

Multiple files can also be copied in a single specification using the standard DOS wild-cards "*". To copy all .DAT files to a Flash card in drive D:

```
TCXCOPY *.DAT D:
```

To copy a file called TMM.BAT, all .WK2 files, and all .PRN files to a subdirectory called \TMM on a Flash card:

```
TCXCOPY -TYPE FLASH TMM.BAT *.WK2 *.PRN D:\TMM
```

To copy all .EXE files from \BIN and its subdirectories to a subdirectory called \TMM\BIN on an SRAM card:

```
TCXCOPY -TYPE RAM \BIN\*.EXE/S D:\TMM\BIN
```

To copy all .EXE files from \BIN and all its subdirectories, and to copy all files named TMM.BAT in the current directory and all subdirectories:

```
TCXCOPY -TYPE RAM \BIN\*.EXE TMM.BAT D:\TMM /S
```

All files matched from \BIN and from the current directory are copied into directory \TMM on the card; subdirectories are created below \TMM as needed. To read a list of instructions from FILES.LIS, implicitly searching all subdirectories; and to search all subdirectories of the current directory for TMM.BAT, copying all matching files:

```
TCXCOPY FILES.LIS/F TMM.BAT D: /S
```

Copying Files With Unusual Names



To copy a file whose name begins with "-", either specify an explicit drive or an explicit directory. The following command copies all files named "-FILE-.*" from the current directory and its subdirectories to the card in drive D:.

```
TCXCOPY .\-FILE-.*/S D:
```

If the ".\" were omitted, TCXCOPY would try to interpret "-FILE-.*" as a command option, and would print an error message.

Using UNIX-style Command Syntax

For special applications, the file-directive option specifier ("/") may be changed with the -NOSLASH option. -NOSLASH has two effects:

1) The forward slash ("/") no longer introduces options. It can then be used as a directory separator and is interchangeable with "\" within file names.

2) File-directive-options are introduced with the character "-", just like command options. However, these options must be separated from file names, and from each other, by spaces or tabs.

For example, the following two commands are equivalent:

```
TCXCOPY \BIN\*.*/S/H \123\*.WKS D:

TCXCOPY -NOSLASH /BIN/*.* -S -H \123\*.WKS D:
```

Environment Variables

TCXCOPY's default options can be modified by setting up a variable in the DOS environment named TCXCOYPFLAGS. Any options contained in TCXCOPYFLAGS are scanned before the options from the command line. For example, add the following command to the AUTOEXEC.BAT file:

```
SET TCXCOPYFLAGS=OPTIONS
```

For example, to tell TCXCOPY that it should expect to use Flash cards by default, and that it should always display statistics, use the following command:

```
SET TCXCOPYFLAGS=-STATS -TYPE FLASH
```



NOTES



Appendix B. Configuration File Keywords

The configuration file format consists of a KEYWORD field followed by a VALUE field. Keywords and value fields are not case sensitive.

It is not necessary to specify all of the fields. `TCINIT` queries the card in an attempt to add any needed information to the CIS data. Refer to Section 4.4.3, *Pre-defined Configuration Files* for typical information to specify.



WARNING!

Create a different configuration file for each unique card type to be initialized. Initializing an SRAM card with CIS information intended for a Flash EPROM type card could result in the SRAM card being damaged by excessive voltages during the programming.

Keyword Format

All keywords consist of one or more words, connected by hyphens and ending with a colon:

```
KEYWORD:
MULTI-WORD-KEYWORD:
```

In the following sections, each legal keyword is given, along with the allowable values for the keyword. If the value must be chosen from a select group of words or numbers, the possible choices are shown enclosed in brackets and separated by a vertical bar:

```
KEYWORD: [CHOICE1|CHOICE2|CHOICE3]
```

If the keyword expects either a number or a text string, the choice is represented as:

```
KEYWORD: number
```

or

```
KEYWORD: string
```

Grouped Keywords

In some cases keywords are pulled together into groups. In this situation, the group keywords only have a special meaning when inside the group. All groups start with a special keyword that indicates a group start:

```
GROUP-START-KEYWORD:
GROUP-KEYWORD1: value
GROUP-KEYWORD2: value
```

The groups are logically arranged in the following levels:

```
GENERAL GROUP
PHYSICAL LAYOUT GROUP
SPACE GROUP
PARTITION GROUP
GEOMETRY GROUP
BATTERY GROUP
```

Delimiters

All white-space characters (space, tab, carriage return, line feed) are treated as keyword and value delimiters. In the case of a keyword, the colon character that always ends a keyword also acts as a delimiter. This means it is unnecessary to add a white space character to separate a keyword from its value. Multiple keywords and values are possible on any line in the file. For clarity, all the sample configuration files contain only one keyword per line. The sample configuration files also make use of indentation to group associated keywords together.

Comment Lines

To add a comment line, begin the line with a pound (#) sign. This character must be present in the first column, as shown in the following example.

```
#####################################
# Sample configuration for:
#
#  Intel 1 Meg Flash Cards
#####################################
```

In this case, each line in the block is treated as a comment because each line begins with the comment character (#) in column 1. The comment entries can be useful to add descriptive information to the configuration file, such as the card type.

Strings

The value for a keyword may be a string value. If the string consists of a single word then the word alone may be used:

```
KEYWORD: WORD
```

If multiple words are required, separated by space or tab, enclose the string in double quotes:

```
KEYWORD: "MULTIPLE WORD STRING"
```

In some cases the double quote characters themselves may be in the string. To construct such a string, precede each double quote in the string with a backslash:

```
KEYWORD:"This \"string \"contains a quoted word"
```

Keywords - General Group

`BATTERY:`

This keyword indicates the start of a **battery group**. All keywords encountered after this keyword are assumed to belong to the battery group until the next general group keyword is encountered. See section *Keywords - Battery Group*.



CARDSIZE: NUMBER[K|M]

This keyword specifies the size of the entire physical card. The number can end with a K, in which case the number is assumed to indicate the size in kilobytes, or the number can end with an M, in which case the size is assumed to indicate the size in megabytes. The number specified is a decimal value.

CARDTYPE: STRING

This keyword specifies the type or name of the card. The value supplied here should be the same as the value used with the -TYPE parameter for the TC commands.

The supported types are:

FLASH|RAM

Where FLASH is for Flash memory cards and RAM is for SRAM memory cards.

CIS:[NONE|BASIC|FULL]

This keyword is used to specify the level of CIS data to be recorded on the card.

NONE

If specified, then no CIS information is written to the card. Use this option when preparing cards for systems that don't support CIS.

BASIC

This value specifies that the CIS information written to the card only contains Level 1 CIS data. This data describes the physical properties of the card but does not describe the data recording format.

FULL

This value specifies that the value of CIS information written to the card contain all known recording format information as well as the card's physical properties.

CIS-CHECKSUM: [YES|NO]

YES

To ensure that the data interpreted by the TC commands is valid, specify YES so that error detection is turned on, and inadvertent changes to the CIS are detected.

NO

Specify NO if error detection is not required. It is recommended that the YES value be used with this keyword since accidental modification of the CIS data could result in damage to the card.

CIS-RESERVE: number

This keyword specifies the number of 512 byte blocks of card space to be reserved for the CIS information. In most cases the number should be set to 1. A number other than 1 allows space to be reserved for future expansion of the CIS data.

ECHO: string

Use this keyword to know if the content of a particular configuration file has been read in. The entire string is echoed to STDERR.

PARTITION:

This keyword indicates the start of a **partition group**. All keywords encountered after this keyword are assumed to belong to the partition group until the next general group keyword is encountered. See section *Keywords-Partition Group* for all keywords in this group.

PARTITIONS: number

This keyword specifies the number of separate partitions to be created on the card. The PCMCIA standard allows a card to be created that contains multiple partitions that support the same or differing file systems or one execute-in-place (XIP) partition. See *Multiple Partition Memory Cards* for more information on creating a multi-partition card.



PHYSLAYOUT:

This keyword indicates the start of a **physical layout group**. All keywords encountered after this keyword are assumed to belong to the physical layout group until the next general group keyword is encountered. See section *Keywords-Physical Layout Group* for all of the keywords in this group.

Keywords - Physical Layout Group

ATTRIBUTE-SPACE:
COMMON-SPACE:

These keywords specify the start of a **space group**. The ATTRIBUTE-SPACE keyword indicates that the following space group applies to attribute space while the COMMON-SPACE keyword indicates that the group applies to common space. See section *Keywords-Space Group*.

JEDEC-ID: number number

This keyword allows the specification of the JEDEC manufacturer and device ID, respectively. The number values specified should be two-digit hexadecimal values. Refer to the manufacturer's documentation that came with the card.

REG-SPACE: [COMPLIES|ALIASED|NONE]

This keyword indicates how the card complies with the PCMCIA standard regarding the support of attribute space.

If the card contains true attribute space that is separate and distinct from common space, the COMPLIES value should be specified. If the card maps the attribute space addresses over the common space then ALIASED should be specified. (Intel Flash memory cards are aliased). If the card does not support attribute space, then NONE should be specified. When the level of support is not known, use NONE .

Keywords - Space Group

DEVICE-COUNT: number

This keyword is used to indicate the number of chips that comprise this physical section.

DEVICE-SIZE: number

This keyword is used to specify the size of the chips in this physical section of the card.

DEVICE-TYPE: [NULL|RAM|FLASH]

This keyword indicates the chip type in this physical section of the card.

SPEED: number

This keyword is used to specify the access speed of the chips in this section of the card. This number is stated in nanoseconds.

BIT-WIDTH: number

This keyword indicates the bit width that should be used to access the card. In most cases, this value should be 16.



Keyword - Partition Group

BASE-ADDRESS: number

This keyword specifies the starting card address of the partition. When supplying the card size via the keyword CARDSIZE**,** don't specify the base address of the partition; only the partition group keyword SIZE needs to be present. Up to three values may be supplied with this keyword.

ERROR-DETECT:[NONE|CHECKSUM|CRC|PARTITION]
[INLINE|TABLE]
[SIZE:[0|1|2]]

Use one of these values to indicate the type of error detection that should be used with the partition. If a DOS file system is to be used then any one of four types NONE, CHECKSUM, CRC or PARTITION may be specified. NONE indicates that no error detection is used. CHECKSUM uses a one-byte checksum for each block in the partition. CRC uses a two-byte SDLC error detection for each block in the partition. PARTITION uses a one-byte checksum for the entire partition.

INLINE|TABLE

Use one of these values to indicate how the error detection data should be stored. INLINE records the data immediately following a block. TABLE causes a table of data to be created at the end of the partition. Don't specify this value if error detection is set to NONE**.**

SIZE

If SIZE is specified, it should match the checksum sizes given above. SIZE may also be used with NONE to reserve room for error detection code.

CHECKSUM executes faster than CRC but CRC provides a much more reliable error detection algorithm. Use of PARTITION should only be used when the system isn't expected to write (or have only limited writes) to the partition.

If access speed is a concern, specify TABLE for the data location as this allows the partition data blocks to remain aligned with 16-byte address boundaries for optimum data access.

FILE-SYSTEM: [DOS|FLASH]

This keyword indicates the type of file system that the card will contain. If this keyword is specified then the ORGANIZATION value should be DISK**.**

DOS

Use this value to use TCFORMAT on an SRAM card, or TCXCOPY on Flash.

FLASH

Use this value to use TCFORMAT on a Flash card. Note that the EXM-17 software does not include Flash File System (FFS) drivers.

GEOMETRY:

This keyword indicates the start of a **geometry group.** If this keyword is specified, then the ORGANIZATION value should be DISK and the FILE-SYSTEM value should be DOS**.**

ORGANIZATION:[DISK|MEMORY|APPLICATION]

This keyword indicates the basic type of recording format to be used. If the card is to be used with the PCFLASH driver, this value should be set to DISK.

SIZE: number [K|M]

This keyword is used to specify the size of the partition. The number can end with a K**,** in which case the number is assumed to indicate the size in kilobytes, or the number can end with an M**,** in which case the size is assumed to be size in megabytes. If neither is specified then the number is assumed to indicate the size in bytes of the partition. The number is specified in decimal notation.



Keywords - Geometry Group

CYLINDERS: number

This keyword specifies the number of simulated cylinders for this partition. The value should be a number between 1 and 65,535.

NOTE:

This value is one-based as opposed to the DOS equivalent that is zero-based. Therefore, this value should always be one greater than the corresponding DOS value.

SECTORS: number

This keyword specifies the number of simulated sectors per track for this partition. The value should be a number between 1 and 255.

TRACKS: number

This keyword specifies the number of simulated tracks per simulated cylinder. The value should be a number between `1` and `255`.

Keyword - Battery Group

`INSTALLED:` mm-dd-yy

This keyword is used to indicate the installation date of the card battery. The format of the date is two digits specifying the month, followed by a hyphen and two digits specifying the day of the month, followed by another hyphen and two digits specifying the year.

`REPLACE-BY:` mm-dd-yy

This keyword is used to indicate the date by which the card battery should be replaced. The format of the date is two digits specifying the month, followed by a hyphen and two digits specifying the day of the month, followed by another hyphen and two digits specifying the year.

Sample Configuration Files

The following configuration file can be used to format a 512KB SRAM card that does not have attribute memory space.

```
# This section specifies the
# card's physical properties.
CARDTYPE: RAM
CARDSIZE: 512K
# This card has no attribute
# memory.
PHYSLAYOUT:
 REG-SPACE: NONE
 COMMON-SPACE:
 DEVICE-TYPE: RAM
# Format 1 partition for the card.
# Use DOS FAT for file system
# Use CRC error detection and
# store it in a table.
PARTITIONS: 1
PARTITION:
 FILE-SYSTEM: DOS
ERROR-DETECTION: CRC TABLE
# Record this information in the CIS.
CIS:FULL
CIS-CHECKSUM: YES
CIS-RESERVE: 1
```



The following configuration file can be used to prepare an Intel 1MB Flash card to have DOS FAT. When using TCXCOPY to format and put files on the card, there must be no error detection.

```
# This section specifies the
# card's physical properties.
CARDTYPE: FLASH
CARDSIZE: 1M
# This card maps attribute space
# addresses over the common space.
PHYSLAYOUT:
 REG-SPACE: ALIASED
 COMMON-SPACE:
 DEVICE-TYPE: FLASH
 SPEED: 200
 BIT-WIDTH: 16

# Format 1 partition for the card.
# Use DOS FAT for file system.
# No error detection.

PARTITIONS: 1
PARTITION:
 FILE-SYSTEM: DOS
 ERROR-DETECTION: NONE

# Record this information in the CIS.
CIS: FULL
CIS-CHECKSUM: YES
CIS-RESERVE: 1
```

NOTES



Appendix C. PCMCIA Card Codes

Appendix C contains a list of manufacturers, part numbers, and codes for configuring available PCMCIA memory cards.

Be sure to check any `README` files included with the EXM-17 software to find out about updates to this table.

C C

Manufacturer	Part Number	Code To Use
Dupont	90100-504	90100504
	90100-505	90100505
	90100-507	90100507
	90100-509	90100509
Epson	AWB065SD20	A065SD20
	AWB065SD30	A065SD30
	AWB065SD40	A065SD40
	AWB101SD20	A101SD20
	AWB101SD30	A101SD30
	AWB101SD40	A101SD40
	AWB129SD20	A129SD20
	AWB129SD30	A129SD30
	AWB129SD40	A129SD40
	AWB201SD20	A201SD20
	AWB201SD30	A201SD30
	AWB201SD40	A201SD40
	AWB257SD20	A257SD20
	AWB257SD30	A257SD30
	AWB257SD40	A257SD40
	AWB513SD20	A513SD20
	AWB513SD30	A513SD30
	AWB513SD40	A513SD40
	BWB101SDX0	B101SDX0
	BWB101SDY0	B101SDY0
	BWB201SDX0	B201SDX0
	BWB201SDY0	B201SDY0
	BWB257SDX0	B257SDX0
	BWB257SDY0	B257SDY0
	BWB513SDX0	B513SDX0
	BWB513SDY0	B513SDY0

Manufacturer	Part Number	Code To Use
Epson	OWB101SDX0	O101SDX0
	OWB101SDY0	O101SDY0
	OWB201SDX0	O201SDX0
	OWB201SDY0	O201SDY0
	OWB257SDX0	O257SDX0
	OWB257SDY0	O257SDY0
	OWB513SDX0	O513SDX0
	OWB513SDY0	O513SDY0
	HWB101SDX0	H101SDX0
	HWB101SDY0	H101SDY0
	HWB201SDX0	H201SDX0
	HWB201SDY0	H201SDY0
	HWB257SDX0	H257SDX0
	HWB257SDY0	H257SDY0
	HWB513SDX0	H513SDX0
	HWB513SDY0	H513SDY0
Fujisoku	JS64G3-C-11	JS64G3
Fujitsu	MB98A60811-25	60811
	MB98A60812-25	60812
	MB98A60813-25	60813
	MB98A608A1-25	608A1
	MB98A608A2-25	608A2
	MB98A60911-25	60911
	MB98A60912-25	60912
	MB98A60913-25	60913
	MB98A609A1-25	609A1
	MB98A609A2-25	609A2
Fujitsu	MB98A61011-25	61011
	MB98A61012-25	61012
	MB98A61013-25	61013
	MB98A610A1-25	610A1
	MB98A610A2-25	610A2

C C

Manufacturer	Part Number	Code To Use
Fujitsu, continued	MB98A61103-25	61103
	MB98A61111-25	61111
	MB98A61112-25	61112
	MB98A61113-25	61113
	MB98A90601-20	90601
	MB98A90602-20	90602
	MB98A90603-20	90603
	MB98A90701-20	90701
	MB98A90702-20	90702
	MB98A90703-20	90703
	MB98A9080	9080
	MB98A90801-20	90801
	MB98A90802-20	90802
	MB98A90803-20	90803
	MB98A9090	9090
	MB98A90901-20	90901
	MB98A90902-20	90902
	MB98A90903-20	90903
HP	F1002A	F1002A
	F1003A	F1003A
Intel	iMC001FLKA (1M)	IMC001
	iMC002FLKA (2M)	IMC002
	iMC004FLKA (4M)	IMC004
Macro	451F-3	451F-3
	451S-1	451S-1
	451S-2	451S-2
	451S-3	451S-3
	451S-4	451S-4
	451S-5	451S-5
	451S-6	451S-6
Mitsubishi	MF32M1	MF32M1
	MF35M1	MF35M1
Poqet	SC9001-162H1	SC9001
	SC9011-H011	SC9011-H
	SC9011-P011	SC9011-P

Index

A

A: drive, 9, *28-30*
address, base, *5-6*
aliased cards, *B-6*
application keyword, *B-9*
attribute space, *19*, *B-6*
AUTOEXEC.BAT file, *7*, *8*, *28-30*, *A-5*, *A-8*, *A-14*, *A-18*, *A-27*
Award/Databook, *10*

B

BASE-ADDRESS keyword, *B-7*
battery in SRAM cards, *14*
battery group of keywords, *B-2*, *B-3*, *B-10*
BIOS setup, *4-6*
BIN (binary image), *A-12*
BIT-WIDTH keyword, *B-7*
booting considerations, *6*, *8*, *9*, *28-32*
BUFFER option, *A-3*, *A-11*, *A-16*

C

C-page of memory, see Restrictions
Card Information Structure (CIS), *15-30*
  store format information, *15*
  to format SRAM cards, *16*
  to format Flash cards, *16-17*
  specifying CIS, *18*
  overview of CIS, *19*
  physical and logical layouts, *19*
  built with DOS text editor, *20*
  with pre-defined files, *20-21*
  using options when missing, *22*
  placing onto memory cards, *23-24*
  used with multi-partitions, *26-28*
  with TCINIT, *A-9 to A-10*
card codes, *C-1 to C-4*
card enable bit, *5-6*
-CARD option, *18*, *22-23*, *27*, *A-3*, *A-4*, *A-6*, *A-9*, *A-17*
CARD-SIZE, *A-5*, *A-13*, *A-17*, *A-21*, *B-4*
CARD-TYPE, *25*, *A-5*, *A-7*, *A-13*, *A-14*, *A-22*, *B-4*
.CFG file, see Configuration files
changing AUTOEXEC.BAT, 7-8
changing CONFIG.SYS, *7-13*, *24-32*
CHECKSUM keyword, *B-7*, *B-8*, *B-11*, *B-12*
checksum errors, *25*
CIS-RESERVE keyword, *B-5*
COM port, *13*
comment lines in keywords, *B-3*
COMMON-SPACE keyword, *B-6*
-CONFIG option, *20*, *23*, *27*, *A-4*, *A-7*, *A-10*, *A-12*, *A-17*, *A-20*
CONFIG.SYS, *7-13*, *24-32*
configuration data, *4*
configuration files, *15 - 28*
  formatting cards with, *15*
  formatting SRAM cards, *16*
  formatting Flash cards, *16-17*
  to build CIS, *18*
  overview, *19*
  pre-defined, *20-21*
  sample files, *B-11 to B-12*
  to specify part numbers, *22*, *C-1 to C-4*
  multi-partitioning, *25-28*
see also -CONFIG, -CARD, TCINIT
COPY command, 7, *29*
CRC error detection, *25*

I

CTRL-ALT-ESC keys, *4*
CYLINDERS keyword, *B-9*

D

D-page of memory, 6, 32, see *Restrictions*
Databook, *10*
DEBUG option, A-4, *A-12, A-20*
default card configuration, *19*
default drive D:, *14*
default TC command options, *20, A-2*
device drivers, see *CONFIG.SYS, PCICCARD.SYS, EMAP.SYS, EMAP.SYS, PC!C16.SYS*
DEVICE-COUNT keyword, *B-7*
DEVICE-TYPE keyword, *B-7*
DEVICEHIGH command, *9, 32*
devices supported, *1*
DIR command with Flash cards, *17*
DOS drive, EXM-17 as, *1*
DOS Partition, *24 (See also FAT, FFS)*
Dupont PCMCIA cards, *C-2*

E

ECHO string, *B-5*
EMAP.SYS, *32*
embedded systems, *A-15*
EMC computer, *31-32*
environment variables, *A-2, A-5, A-8, A-14, A-18, A-27*
EPC computers, *4, 31-32*
EPSON PCMCIA cards, *C-2*
erasing Flash cards, *16 (see also TCERASE)*
ERROR-DETECT keyword, *B-7*
errors during software installation, *10-12*
error messages, *10-12*
ESD damage, *3*
Ethernet LAN cards, *1, 13-14*
EXM-1, conflicts with EXM-17, *32*
EXM-10, conflicts with EXM-17, *32*
EXM-17 as I/O, *32*
EXM ID, *5*
EXM setup screen, *4*
EXM slot numbering, *5*

F

fax+data modem cards, *13*
FEFS.SYS driver, *27*
File Allocation Table (FAT), *20, 21, 25-28, B-11 to B-12*
FILE-SYSTEM keyword, *B-8*
Flags, *A-2 (see also Environment variables)*
Flash cards, *14-30*
  overview, *14*
  formatting, *15-17*
  - with CIS, *18-19*
  - with configuration files, *19-20*
  - with pre-defined files, *20-21*
  - with part numbers, *22*
  - with generic information, *22*
  - with multi-partitions, *24-30*
  - bootable, *28-30*
  as supported card type, *A-14*
  in embedded systems, *A-15*
  as CARD-TYPE, *B4*
  (see also Appendix B, TCERASE, TCINIT, and TCXCOPY
Flash File System (FFS), *20, 21, 26-30, B-8*
formatting Flash cards, *16-17, 26-30*
formatting SRAM cards, *16, 25-26*
formatting multi-partitions, *24-30*
front panel, *3*
Fujisoku PCMCIA cards, *C-2*
Fujitsu PCMCIA cards, *C-2 to C-3*

G

generic parameters, *22-23*
GEOMETRY group of keywords, *B-9 to B-10*
global file directive options, in TCXCOPY, *A-24*

H

hardware installation, *3-6*
hot insertion, *1, 13*
HP PCMCIA cards, *C-4*
humidity specification, *2*

I

IBM Token Ring network card, *13*
I/O cards, *13*
I/O space, *6, 31-32*
ID configuration, *5*
IFS.SYS driver, *27*
initializing Flash cards, *17*
inserting the EXM-17 into an EXM carrier, *3*
INSTALL.BAT, *7*
installation error messages, *10-12*
installing hardware, *3-6*
installing software, *7-12*
Intel ASCII file, *A-11*
Intel OMF, *A-11*
Intel PCMCIA cards, *C-4*

J

JEDEC-ID, *B-6*
JEIDA standard, *10, 14, 22, 26, A-4, A-7, A-11, A-17, A-20*

K

keywords, *15, 24, B-1 to B-12*
- battery group, *B-10*
- comment lines, *B-2*
- delimiters, *B-2*
- format, *B-1 to B-2*
- geometry group, *B-9*
- general group, *B-3 to B-6*
- grouped keywords, *B-2*
- partition group, *B-7 to B-9*
- physical layout group, *B-6*
- sample files, *B-11, B-12*
- space group, *B-6*
- strings, *B-3*

L

LASTDRIVE, *11, 24*

M

MACRO PCMCIA cards, *C-4*
manufacturers listing, C-*1 to C-4*
manufacturer's part number on PCMCIA cards, *22*
memory, *6*
memory cards, *14-30 (see also Flash cards, SRAM cards)*
memory card information structure, *18-24 (see also CIS)*
memory conflicts, *31-32*
Mitsubishi PCMCIA cards, *C-4*
modem cards, *13-14*
modem and fax interfaces, *1*
modem drivers, *9*
multi-partitions, *24-32*

N

network cards, *13-14*
NODEBUG option, *A-12, A-20*
NOE option, *A-23*
NOH option, *A-23*
NOOVERRIDE option, *A-4, A-21*
NOPRECHECK option, *A-20*
NOPROGRESS option, *A-4, A-13, A-21*
NOS file directive option, *A-23*
NOSLASH option, *A-21*

NOSTATS option, *A5, A-22*
NOV option, *A-7, A-14, A-17, A-22*
NOW option, *A-7*

O

OB1 configuration, *4-6*
OB2 configuration, *4-6*
OVERRIDE option, *A-4, A-12, A-21*
options, TC commands, *A-1 to A-2*

P

PARAMETERS, *A-1*
PARTITIONS keyword, *B-5*
PATH statement, *8*
PCFLASH.SYS driver, *A-27*
PCIC16.SYS driver, *32*
PCICCARD.SYS driver, *8-12, 13, 24-30, 31-32*
PCMCIA card manufacturers, C-*1 to C-4*
PHYSLAYOUT keyword, *B-6*
Poqet PCMCIA cards, *C-4*
power specification, *2*
PRECHECK option, *A-20*
predefined configuration files, *8, 20-22*
preserving I/O space, *6*
programming memory cards, *See SRAM cards, Flash cards, formatting*
PROGRESS option, *A-12, A-20*

Q

Quick Exchange Services, *33*

R

RadiSys address, *36*
RadiSys fax number, *33*
RadiSys phone number, *33*
RadiSys Technical Support, *33-36*
REMAP.SYS, *9, 28-30*
Repair Service, *33*
ROM base address, *5-6*

S

select bit, *5-6*
shock specification, *2*
-SIZE option, *22, 27, A-5, A-13, A-17, A-21*
software installation, *7-12*
specifying generic card information, *22-23*
specifying generic information, *22*
specifying part numbers, *22*
specifications, *2*
SLASH option, *A-21*
SLOT configuration, *4*
SRAM cards, *14-30, C-1 to C-4*
  overview, *14*
  formatting, *15-16*
  - with CIS, *18-19*
  - with configuration files, *19-20*
  - with pre-defined files, *20-21*
  - with part numbers, *22*
  - with generic information, *22*
  - with multi-partitions, *24-30*
  - bootable, *28-30*
  supported card type (RAM), *A-14*
  in embedded systems, *A-15*
  as CARD-TYPE, *B-4*
  (see also Appendix B, TCFORMAT and TCXCOPY
STATS option, *A-5, A-13, A-22*
subplane, *3*
-SYS option, *29, A-7, A-22*

T

TC commands, *15-30, A-1 to A-26*
General Information, *A-1*
Specifying defaults, *A-2*
TCERASE, *16 -17, 22-28, A-3 to A-5*
Command syntax, *A-3*
Options, *A-3*
Environment variables, *A-5*
TCFORMAT, *19-21, 25-30, A-6 to A-8*
Command syntax, *A-6*
Options, *A-6 to A-7*
Environment variables, *17, A-8*
TCINIT, *17-19, 22-24, 26-28, A-9, A-10*
Command syntax, *A-9 to A-10*
Options, *A-9 to A-10*
TCPROG, *A-11 to A-15*
Command syntax, *A-11*
Options, *A-11*
Supported card types, *A-14*
Environment variables, *A-14*
Programming cards for embedded systems, *A-15*
TCREAD, *A-16 to A-18*
Command syntax, *A-16*
Options, *A-16*
Environment variables, *A-18*
TCXCOPY, *28-30, A-19 to A-27*
Command syntax, *A-19*
Options, *A-19*
File-Directives, *A-22*
Global options, *A-24*
Copying a single file, *A-24*
Copying a directory, *A-24*
Copying multiple files, *A-25*
Copying files with multiple names, *A-26*

Using UNIX-style syntax, *A-26*
Environment Variables, *A-26*
Technical Support, *33-36*
Technical Support phone number, *33*
temperature specification, *2*
thumbscrews, *3*
Token ring LAN cards, *1, 13, 14*
Troubleshooting, *31-32*
(see also Error Messages)
-TYPE option, *18, 19, A-5, A-7, A-13, A-22*
Type I and Type II cards, *1, 7, 13, 14*

U

UNIX-style command syntax, *A-26*

V

V option, *A-5, A-7, A-13, A-17, A-22*
VALUE field, with Keywords, *B-1*
vibration specification, *2*
verifying software installation, *10-12*
VME, *5, 31-32*
VXI, *5, 31-32*

W

W option, A-7,
warranty information, *iii, 33-36*
Which memory card to use? *15*

X

XCOPY, *A-19*

NOTES

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP