AP-633

APPLICATION NOTE

Miniature Card PC Connectivity Guide

October 1996

Order Number: 292189-001

CONTENTS

ADVANCE INFORMATION

REVISION HISTORY

Number	Description
-001	Original version

1.0 INTRODUCTION

The importance of connectivity to a personal computer (PC) continues to grow with the emergence of new digital consumer applications like digital cameras, digital audio recorders and hand-held PCs (see Figure 1). These new devices provide a mechanism to record data while away from your PC. With the right PC connectivity solution, a consumer can easily transfer data back to the PC to be processed, manipulated, transferred over a network, or archived. The Miniature Card PC Connectivity Guide focuses on solutions that make data transfer to the PC a reality.

Miniature Card is a new digital media that provides a low-cost, low-power, high-performance memory interface. It has a very small form factor with an elastomeric connector interface that provides a very rugged and reliable consumer friendly connection.

This document provides guidelines to using this new memory technology to transfer data to a PC. The following sections discuss the software and hardware issues to assist the Miniature Card-to-PC connection.

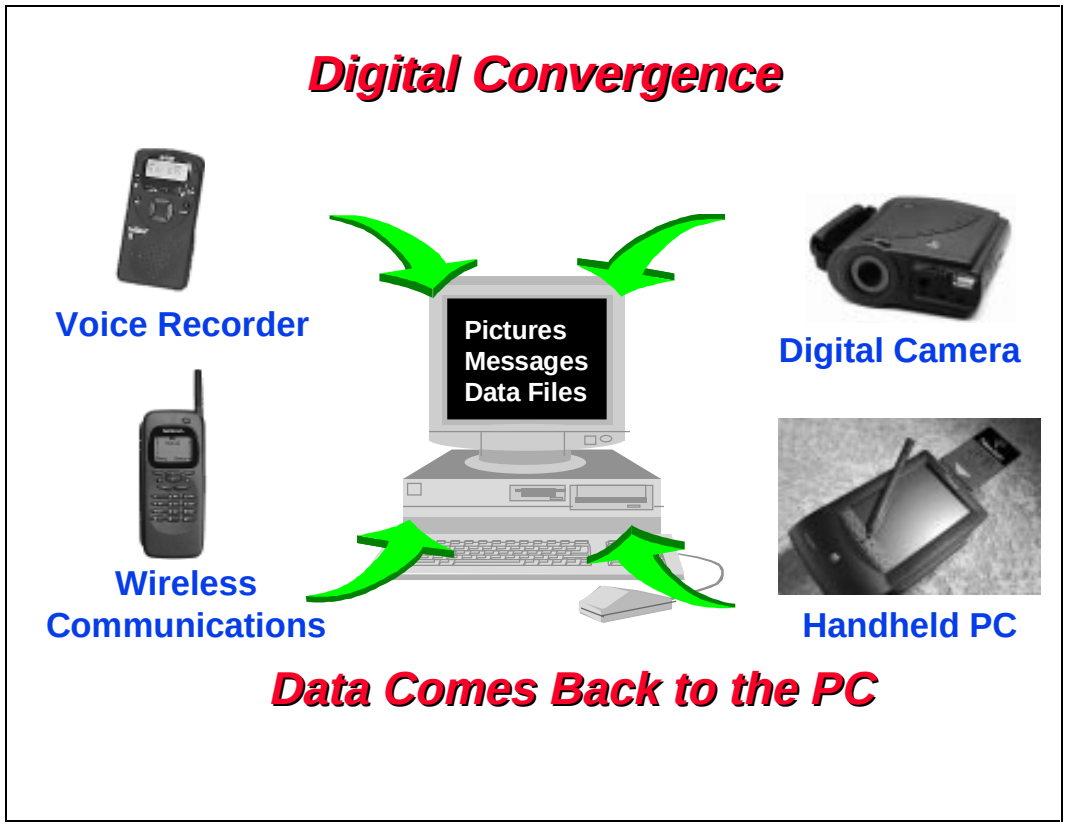

Figure 1. Digital Convergence

2.0 PC CONNECTIVITY SOLUTIONS

In order to provide a seamless connection to the PC, a common media format between the PC and embedded devices must exist, and a hardware interface to the PC must be available.

Common Media Format: All data must be recorded in a common data file format, and this format must also be recognized by the PC. This common format exists today and is called the Flash Translation Layer (FTL). FTL is an industry specification that was recently accepted into the PC Card standard. With the FTL standard, users can record data on embedded applications and transfer that data back to the PC where it can be manipulated or archived. This application note discusses FTL in Section 3.0.

Hardware Interface: There are several hardware solutions for PC Connectivity. In each case, there is a specific hardware and software stack associated with the solution.

- PC Card Connectivity: All notebook PCs come equipped with PC Card slots. The PC Card provides a convenient and fast method of data transfer. PC Card slots can also be added to desktop PCs using an ISA add-in card. PC Cards are the fastest connectivity solution. PC Card Connectivity is covered in Section 4.0.
- Universal Serial Bus (USB) Connectivity: USB is a high speed, plug and play serial bus. A USB reader/writer provides a convenient and fast method of data transfer to a desktop PC. The USB reader/writer can be a stand-alone device or combined with another device like a keyboard. USB Connectivity is covered in Section 5.0.
- Legacy PC Connectivity Solutions: For the existing installed base of desktop PCs, there are other connectivity solutions. These solutions include a serial cable or a parallel port reader/writer. Legacy PC Connectivity Solutions are covered in Section 6.0.

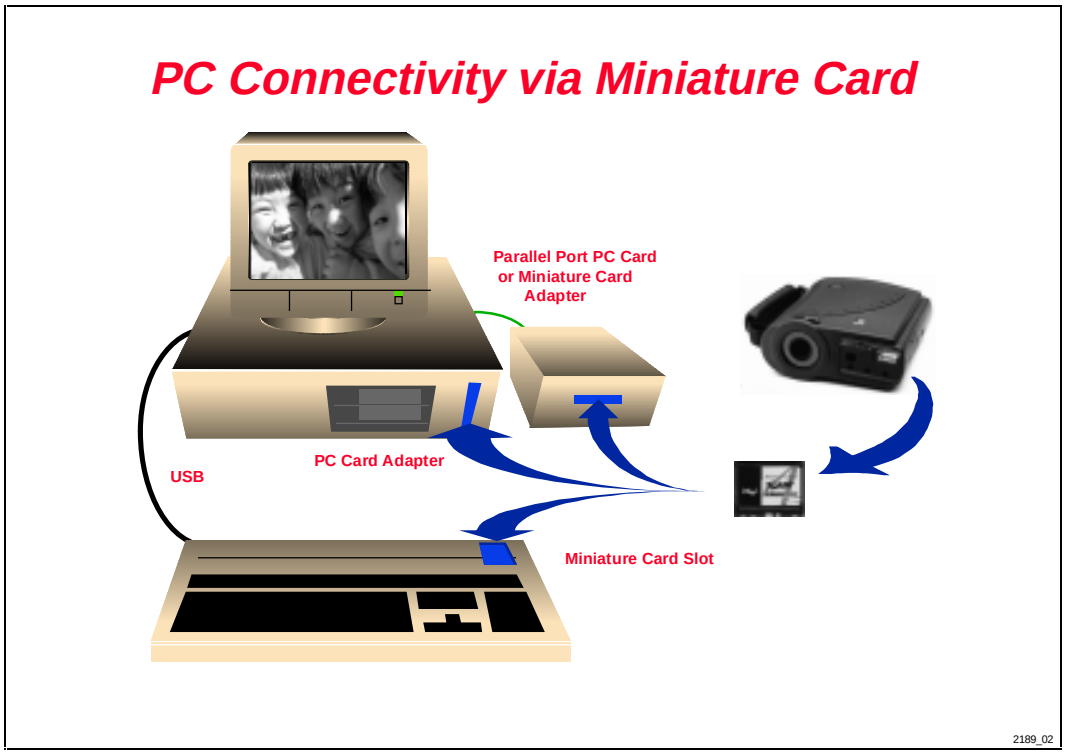

Figure 2. PC Connectivity Options

3.0 FLASH TRANSLATION LAYER

FTL is an industry-standard data format for flash media that was accepted into the PC Card Standard by the PCMCIA standards body (March, 1996 Update). FTL flash file systems are software which provide flash media management that uses and maintains the FTL format on the flash media. By translating FAT sector calls to virtual flash sectors, FTL software makes a flash array look just like a hard disk drive. FTL provides an industry-standard media format for PC compatibility. Data can be recorded on one PC and then transferred to another PC. Digital electronic devices that use the FTL media format are also compatible with the PC. FTL provides a common media format for a seamless connection to the PC. Refer to Appendix A for a list of third-party vendors of FTL software.

3.1 FTL Basics

The FTL filing software is a sector-based media manager that uses an existing sector-based file system, such as DOS FAT, to provide the upper level file handling capabilities. By translating received requests from DOS, an FTL driver appears as a normal sector-based drive to the upper layer software (see Figure 3). Upper layer software expects to be able to modify these sectors at any time. Flash requires that a block be erased before files within that block are modified. When the upper layer software tries to re-write a sector, FTL filing software remaps the request to a free area of flash. These remapped sectors are treated as logical read/write blocks rather than physical sectors. FTL subdivides each flash block into smaller read/write blocks. Each read/write block is the same size as a sector (typically 512 bytes, see Figure 4).

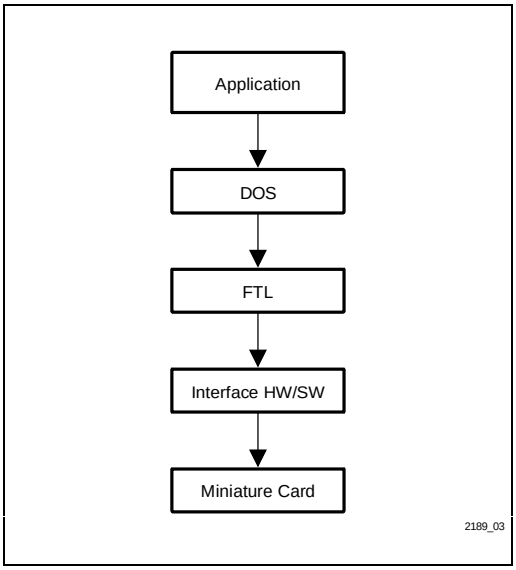

Figure 3. FTL Architecture

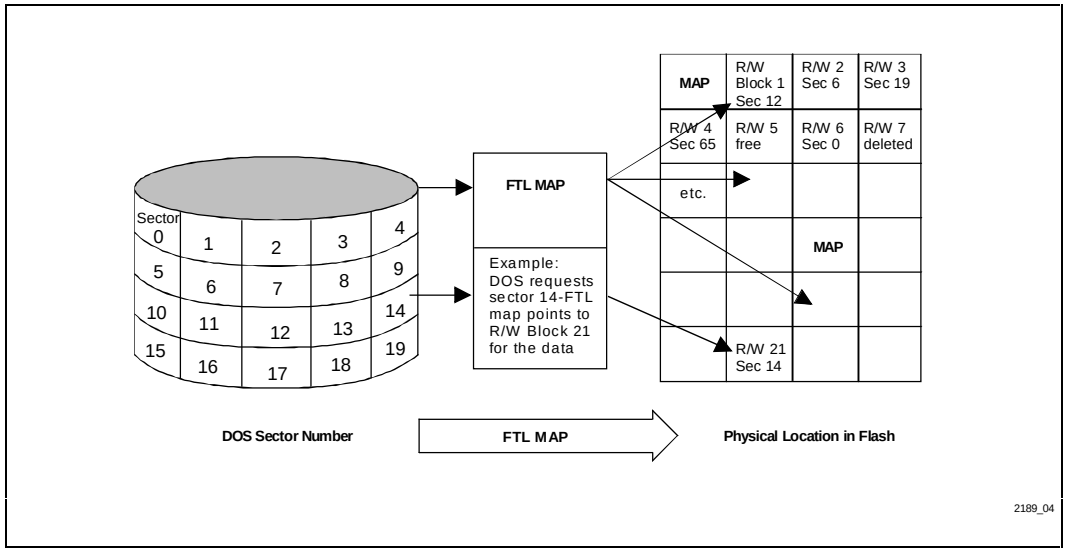

Figure 4. FTL Mapping

Some FTL filing software implement other design features like wear-leveling, background erase, fault recovery methods, and integrated flash chip algorithms. These features are not part of the FTL standard, but they are features of FTL filing software providers.

3.2 FTL-Lite

Some applications do not implement a file system (like DOS FAT) which is required for FTL, and some applications have special design requirements for filing (like file append and insertion). These applications cannot use FTL. However, FTL-Lite is an alternative filing system and flash media manager that provides the flexibility for embedded applications while maintaining FTL format compatibility.

FTL-Lite is specifically designed as a source code product which is easy to customize and integrate into specific embedded applications. FTL-Lite combines a condensed FAT and a FTL file system for a small footprint (typically 14 KB) media manager that records data using the FAT file format on top of the FTL media format (see Figure 5). Applications that use FTL-Lite to record data will maintain file format compatibility with FTL on the PC.

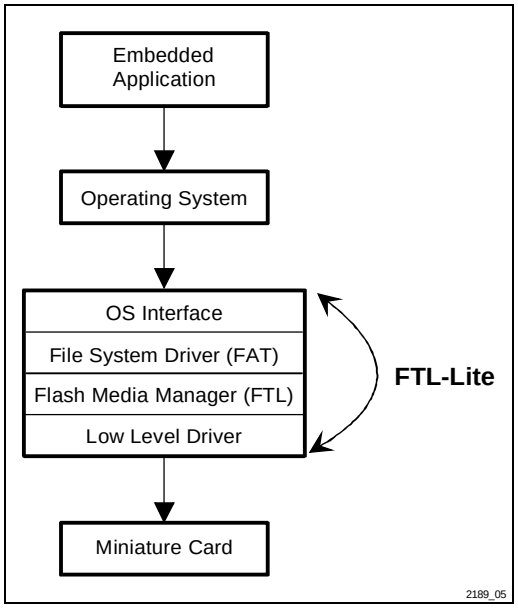

Figure 5. FTL-Lite Software Stack

4.0 PC CARD CONNECTIVITY

Connectivity to the PC is a key feature of Miniature Card. The areas that need to be addressed in order to use Miniature Cards in a PC Card implementation are:

- PC Card Adapter–The hardware needed to connect the electrical signals on the Miniature Card to the PC Card host.
- PC Card Software–A compatible file system to read and write the data on the Miniature Card.
- Compatible Metaformat (CIS)– Ensuring the proper Card Information Structure (CIS) in the Miniature Card's memory area so that it can be properly identified and used by the PC Card configuration software.

Refer to Appendix A for a list of third-party vendors for PC Card connectivity solutions.

4.1 PC Card Adapter

A PC Card adapter for Miniature Card provides the hardware mechanism to take data recorded on a Miniature Card, and transfer that data to a PC or notebook through a PC Card slot. The Miniature Card bus is very similar to the PC Card bus so the PC Card adapter primarily consists of signal routing. However, If the supported Miniature Card voltage(s) does not match the supported PC Card slot voltages, a charge pump and level shifters will be required in the PC Card adapter circuitry.

The PC Card signals and Miniature Card signals are compatible except in the following three areas:

- PC Card signal levels are TTL compatible; Miniature Card uses CMOS levels. Since almost all PC Card slots drive their signals to CMOS levels this is not an issue.
- PC Card's RESET is active high; Miniature Card RESET# is active-low. This incompatibility is easily fixed by adding an inverter to the PC Card adapter. This inverter will convert the PC Card RESET signal from active high to active low.

- PC Card does not support Miniature Card's DRAM signals (RAS#/CAS#), or the serial EEPROM (SCL, SDA) signals. The DRAM-based Miniature Card was not intended to be removed from the embedded system. Since DRAM is a volatile memory, all data will be lost when the card is removed. The Miniature Card specification does define a special DRAM key so the PC Card adapter will reject DRAM Miniature Cards from plugging into the Miniature Card socket.

The Miniature Card is 3.5 mm thick, so the PC Card Adapter requires the type II form factor (5.0 mm thick). Refer to Figure 6 for a conceptual drawing of a PC Card adapter for the Miniature Card.

Figure 6. PC Card Adapter

4.1.1 ADAPTER VOLTAGES

A PC Card adapter can be designed in three ways, depending on the expected operation parameters of the PC Card host and Miniature Card, and the desired cost target:

- 5V PC Card adapter for 5V-capable Miniature Cards
- 3.3V PC Card adapter for 3.3V-capable Miniature Cards
- Universal PC Card adapter to support 5V and 3.3V Miniature Cards

Intel's Series 100 Miniature Card implements SmartVoltage technology enabling it to function in all three adapters described above.

The Miniature Card specification supports three different operational voltages: 5V, 3.3V, and x.xV. The x.x voltage is lower than 3.3V and will be determined at a future date. An adapter that supports x.xV is not covered in this text.

The VS1#, VS2# Signals indicate to the host system the card's operational voltage. If VS1# and VS2# are both open, the card only operates at 5V. If VS1# is grounded, the card can operate at 3.3V. If VS2# is grounded, the card is capable of x.xV operation. Both VS1# and VS2# grounded indicates that the card can operate at either voltage.

Table 1. Voltage Sense Signals

Miniature Card Power-Up Voltage	VS1#	VS2#
5V only	Open	Open
3.3V (1)	GND	Open
x.xV	Open	GND
3.3V and x.xV	GND	GND

NOTES:
1. A 3.3V/5.0V card uses the same sense configuration as the 3.3V card.

In addition to the voltage sense pins, there are also mechanical voltage keys on the Miniature Card that ensure the card can only be inserted into host systems that can supply the proper voltage levels to the card. Therefore, if the Miniature Card fits into the host socket, the host will power-up the card at the voltage defined by the VS1# and VS2# signals.

4.1.2 ADAPTER ADDRESSING

The PC Card standard defines 26 address lines A[25:0]. A_0 is used only by 8-bit hosts to read individual bytes from the PC Card. The PC Card specification requires that PC Cards implement a technique called byte swapping that moves the high data byte to the lower order data signals D[7:0]. A 16 bit host would not use A_0 on a PC Card. The Miniature Card does not support byte swapping and defines A_0 differently. On a Miniature Card, A_0 selects between even and odd words which is equivalent to the PC Card A_1 signal. The PC Card adapter for Miniature Card shifts the address signals by

one because they are defined differently. The adapter connects A[25:1] on the PC Card to A[24:0] on the Miniature Card.

4.1.3 ADAPTER RESET

The PC Card standard defines the RESET signal as a active high signal. However, older versions of the specification did not define RESET. The PC Card standard states that PC Cards may pull RESET up to V_{CC} on cards that require RESET. This ensures that the card remains in RESET until the host drives this signal low. To ensure the Miniature Card remains in RESET, a pull up should be placed in the adapter (on the PC Card side of the adapter).

The Miniature Card specification defines RESET# active low. The adapter requires an inverter to match the functionality of the PC Card standard. This inverter is the only active component in the single voltage adapter.

4.1.4 5 VOLT ADAPTER

The 5V adapter will fit into all PC Card sockets. On 5V hosts, the Miniature Card will be powered-up at 5V. On host that support 3.3V and 5V, the Miniature Card will power-up at 5V if the card's VS1#/VS2# pins are open. If VS1# is grounded (indicating the Miniature Card supports dual voltages), the card will power-up at 3.3V.

If this PC Card adapter is plugged into a 3.3V (or lower) host system, and the Miniature Card only operates at 5V (VS1# and VS2# pins open), the host will not power-up the system (according to the protocol specified in the PC Card Standard).

Requirements of the 5V PC Card Adapter:
PC Card Key: 5V Key (See PC Card Standard)
Miniature Card Adapter Socket:
5V Tab
VS1#, VS2# routed to Miniature Card

4.1.5 3.3 VOLT ADAPTER

This adapter will only accept Miniature Cards capable of operating at 3.3V (indicated by the presence of the 3.3V Notch in the card) and will only plug into PC Card slots that provide 3.3V.

Requirements of the 3.3V PC Card Adapter:
PC Card Key: Low Volt Key
Card Adapter Socket:
3.3V Tab
VS1#, VS2# routed to Miniature Card

NOTE:

The 5V and 3.3V PC Card Adapters are identical except for the following:

- PC Card Connector :
  5V Key vs. Low Voltage Key
- Miniature Card Adapter Socket:
  5V Tab vs. 3.3V Tab
- RESET Inverter Voltage Operation

4.1.6 UNIVERSAL VOLTAGE ADAPTER

The Universal PC Card Adapter provides the flexibility to use both 5V and 3.3V Only Miniature Cards in PC Card hosts that support 5V PC Cards. The adapter contains signal level shifters that translate the host's 5V signals to the Miniature Card's 5V or 3.3V levels.

The adapter provides the following features:

- Detection of Miniature Card's voltage requirements (using VS1#)
- Signal translation from host's 5V levels to Miniature Card's 5V or 3.3V level
- 3.3V V_{CC} generation for 3.3V Miniature Cards
- Host power-up delay and voltage control

Requirements of the Universal PC Card Adapter:
PC Card Key: 5V Key (See PC Card Standard)
Card Adapter Socket: 3.3V/5V Key
VS1# open, not connected to Miniature Card VS1#
VS2# open, not connected to Miniature Card VS2#

NOTE:

The VS1# and VS2# signals on the socket do not connect to the Miniature Card voltage sense signals (the PC Card Adapter will always power-up at 5V at the PC Card socket).

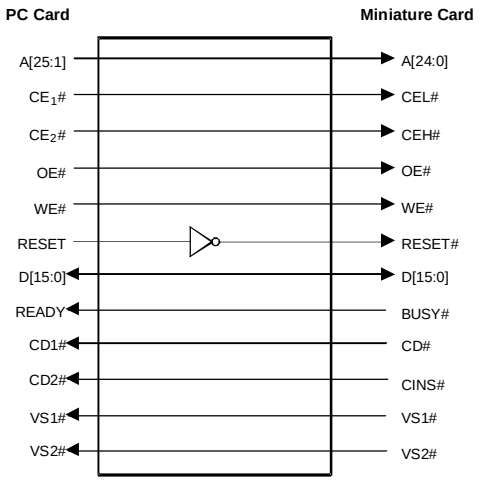

Figure 7. Single Voltage Adapter

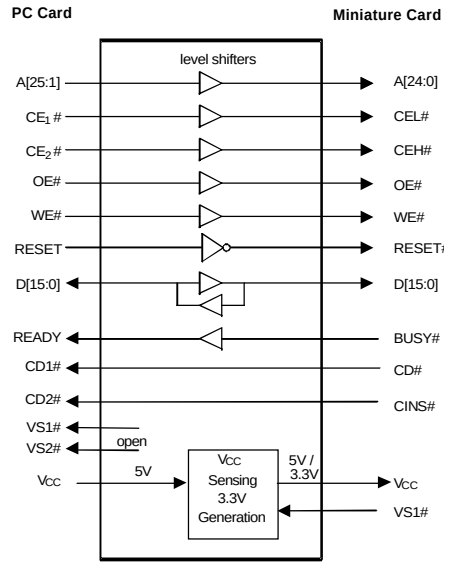

Figure 8. Universal Voltage Adapter

4.1.7 PC CARD ADAPTER SIGNAL ASSIGNMENTS

The following signal assignments apply only to the single voltage adapter (see Figure 7). Refer to the PC Card specification and the Miniature Card specification for pin locations.

Table 2. PC Card Adapter Signals

PC Card Pin	Miniature Card Signal
Pin 1: GND	GND Contact
Pin 2: D_3	Pad 56: D_3
Pin 3: D_4	Pad 26: D_4
Pin 4: D_5	Pad 57: D_5
Pin 5: D_6	Pad 58: D_6
Pin 6: D_7	Pad 28: D_7
Pin 7: CE_1#	Pad 42: CEL#
Pin 8: A_{10}	Pad 7: A_9
Pin 9: OE#	Pad 18: OE#
Pin 10: A_{11}	Pad 37: A_{10}
Pin 11: A_9	Pad 8: A_8
Pin 12: A_8	Pad 39: A_7
Pin 13: A_{13}	Pad 35: A_{12}
Pin 14: A_{14}	Pad 34: A_{13}
Pin 15: WE#	Pad 49: WE#
Pin 16: READY	Pad 48: BUSY#
Pin 17: V_{CC}	V_{CC} Contact
Pin 18: V_{PP1} (not connected)	No Miniature Card Equivalent
Pin 19: A_{16}	Pad 33: A_{15}
Pin 20: A_{15}	Pad 3: A_{14}
Pin 21: A_{12}	Pad 6: A_{11}
Pin 22: A_7	Pad 9: A_6
Pin 23: A_6	Pad 10: A_5
Pin 24: A_5	Pad 41: A_4
Pin 25: A_4	Pad 11: A_3

PC Card Pin	Miniature Card Signal
Pin 26: A_3	Pad 12: A_2
Pin 27: A_2	Pad 43: A_1
Pin 28: A_1	Pad 13: A_0
Pin 29: A_0 (not connected)	No Miniature Card Equivalent
Pin 30: D_0	Pad 24: D_0
Pin 31: D_1	Pad 55: D_1
Pin 32: D2	Pad 25: D_2
Pin 33: WP (tied to GND)	No Miniature Card Equivalent
Pin 34: GND	GND Contact
Pin 35: GND	GND Contact
Pin 36: CD_1#	Pad 46: CD#
Pin 37: D_{11}	Pad 52: D_{11}
Pin 38: D_{12}	Pad 21: D_{12}
Pin 39: D_{13}	Pad 20: D_{13}
Pin 40: D_{14}	Pad 50: D_{14}
Pin 41: D_{15}	Pad 19: D_{15}
Pin 42: CE_2#	Pad 5: CEH#
Pin 43: VS1#	Pad 38: VS1#
Pin 44: RFU (not connected)	No Miniature Card Equivalent
Pin 45: RFU (not connected)	No Miniature Card Equivalent
Pin 46: A_{17}	Pad 2: A_{16}
Pin 47: A_{18}	Pad 32: A_{17}
Pin 48: A_{19}	Pad 1: A_{18}
Pin 49: A_{20}	Pad 31: A_{19}
Pin 50: A_{21}	Pad 60: A_{20}

PC Card Pin	Miniature Card Signal
Pin 51: V_{CC}	V_{CC} Contact
Pin 52: $V_{PP}2$ (not connected)	No Miniature Card Equivalent
Pin 53: A_{22}	Pad 47: A_{21}
Pin 54: A_{23}	Pad 17: A_{22}
Pin 55: A_{24}	Pad 16: A_{23}
Pin 56: A_{25}	Pad 15: A_{24}
Pin 57: VS2#	Pad 53: VS2#
Pin 58: RESET (pulled to V_{CC}, then inverted)	Pad 36: RESET#
Pin 59: WAIT# (tied to V_{CC})	No Miniature Card Equivalent
Pin 60: RFU (not connected)	No Miniature Card Equivalent
Pin 61: REG# (not connected)	No Miniature Card Equivalent
Pin 62: BVD2# (tied to V_{CC})	No Miniature Card Equivalent
Pin 63: BVD1# (tied to V_{CC})	No Miniature Card Equivalent
Pin 64: D_8	Pad 54: D_8
Pin 65: D_9	Pad 23: D_9
Pin 66: D_{10}	Pad 22: D_{10}
Pin 67: CD2#	CIN# Contact
Pin 68: GND	GND Contact

Other Miniature Card Signals:

Pad 4: V_{CCR} Not Connected
Pad 14: RAS# Not Connected
Pad 27: RFU Not Connected
Pad 29: SDA Not Connected
Pad 30: SCL Not Connected
Pad 40: BS8# Tied to V_{CC} in adapter
Pad 44: CASL# Not Connected
Pad 45: CASH# Not Connected
Pad 51: RFU Not Connected
Pad 59: RFU Not Connected

4.2 PC Card Software

PC Card software currently shipping on notebook PCs can transfer data to and from the Miniature Card. The Socket Services layer, Card Services layer, and FTL are all PC Card defined specifications (see Figure 9). These software layers are shipping on all major Notebook PC manufacturers.

Card Services is the layer responsible for allocating system resources for a PC Card. With the appropriate client drivers (like FTL), card services manages the support for any memory or I/O card. Card Services provides a standard interface for upper layer software and is hardware independent.

Socket Services is hardware dependent providing the interface between Card Services and the PC Card controller. The PC Card standard defines specific socket services functions for controlling and configuring the PC Card controller.

PC Card Controller is the hardware that provides the PC Card interface. This controller provides the logic necessary to control the PC Cards. For example, the PC Card controller controls the OE# and WE# signals to read and write to memory PC Cards.

For more details on PC Card software, refer to the PC Card specification.

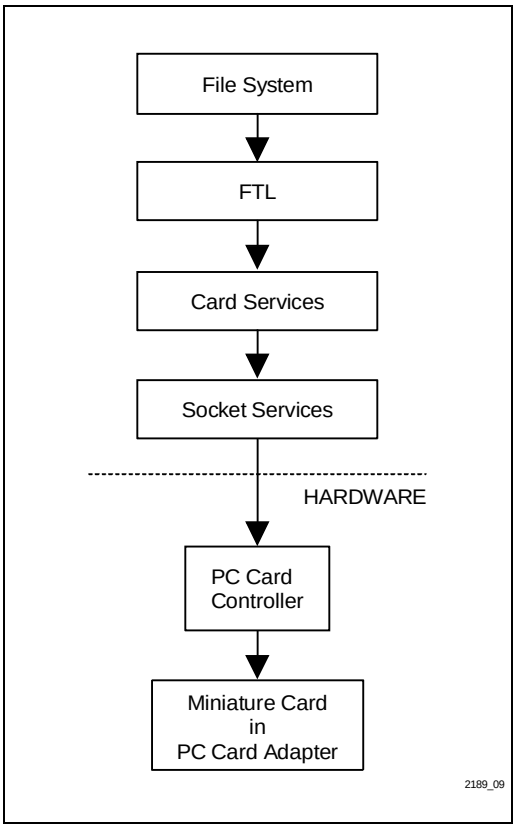

Figure 9. PC Card Software Stack

4.3 Card Information Structure

The CIS provides compatibility with the PC Card standard. This area on the Miniature Card allows the card to be identified and used (through a PC Card adapter) in a PC Card host system. The CIS provides basic information about the Miniature Card for PC Card software to identify and configure the Miniature Card for operation in a PC Card Socket. The CIS consists of data about the card that resides at the beginning of the cards memory array.

Although the PC Card standard requires the presence of a CIS in all cards, PC Card software supports memory cards without a CIS. This is important for customers that do not want to sacrifice any array memory for the CIS.

4.3.1 PC CARD CIS FORMAT

The CIS is a linked-list of data blocks called tuples. The CIS format complies with the PC Card Metaformat Specification and is required for PC Card compatibility (consult the PC Card Standard for details). All tuples have the format shown in Table 3.

Table 3. PC Card Tuple Format

Bytes	Data
0	Tuple Code: CISTPL_XXX
1	Tuple Link: TPL_LINK Number of bytes in tuple body
2	Tuple Body

4.3.2 CIS MEMORY MAP

Table 4 provides an example CIS for a Miniature Card. The format of these tuples meet the PC Card Metaformat Specification, Volume 4.

The first tuple must always be the Device Tuple. Null Tuples fill the space between the Device Tuple and the Miniature Card Tuple (this is required because the AIS must be located at a specific address).

The remaining tuples are either required or recommended for memory cards by the PC Card Metaformat Specification.

For an example of how the data appears in Table 4, refer to the *Intel Series 100 Flash Memory Miniature Card Datasheet* (order number 290581).

4.3.3 TUPLE DESCRIPTIONS

CISTPL_DEVICE (01H): Common memory device information. This tuple contains information such as Device ID, Device Size, Device Type, and Speed for 5V operation.

CISTPL_NULL (00H): Null tuple.

CISTPL_MINI (80H): Miniature Card AIS tuple. This tuple contains the Miniature Card's Attribute Information Structure (AIS) and uses the Vendor Unique (vendor specific information) tuple code as defined in the PC Card Metaformat Specification (February 1995 printing). For more information on the definition of the AIS, refer to the Miniature Card specification.

CISTPL_DEVICEGEO (1EH): Device geometry information for common memory devices. This tuple identifies the devices erase block size, read block size, write block size, partitions, and interleaving.

CISTPL_MANFID (20H): Manufacturer ID. This tuple specifics the manufacturer of the Miniature Card.

CISTPL_FUNCID (21H): Function ID. This tuple identified the card as a memory card.

CISTPL_LONGLINK_C (12H): Longlink to common memory. This tuple dictates a jump to the common memory location specified to continue tuple parsing. This tuple is provided to jump to block 1 to read the tuples recorded by an FTL format.

CISTPL_VERS_1 (15H): Version 1 Product. This tuple identifies the metaformat version level and product information.

CISTPL_JEDEC_C (18H): JEDEC Tuple. This tuple identifies the specific memory devices on the Miniature Card.

CISTPL_DEVICE_OC (1CH): Common memory device other conditions information. This tuple contains information like device ID, size, type, and speed for 3.3V and x.xV operation.

CISTPL_END (FFH): The end-of-chain tuple.

Table 4. CIS Memory Map

Tuple Name	Description	Tuple Code	Address Location
CISTPL_DEVICE	Device Information	01H	0H - 04H
CISTPL_NULL	Null (Ignore)	00H	05H - 0DH
CISTPL_MINI	Miniature Card AIS (Vendor Unique)	80H	0EH - FFH
CISTPL_DEVICEGEO	Device Geometry Information	1EH	100H - 107H
CISTPL_MANFID	Manufacturer Identification String	20H	108H - 10DH
CISTPL_FUNCID	Function Class Identification	21H	10EH - 111H
CISTPL_LONGLINK_C	Longlink to Common Memory	12H	112H - 117H
CISTPL_VERS_1	Level 1 Version/Product Information	15H	118H - 167H
CISTPL_DEVICE_OC	Other Operating Conditions Device Info.	1CH	168H - 16CH
CISTPL_JEDEC_C	JEDEC Identifier Tuple	18H	16DH - 170H
CISTPL_END	The End-of-Chain Tuple	FFH	171H - 172H

5.0 USB CONNECTIVITY

USB is a new plug and play, high-speed serial bus that is capable of transfer rates up to 12 Mbits/sec. USB ports will become standard on PCs by the end of 1996. USB peripherals like keyboards, joysticks, and monitors will also be available by the end of 1996. The PC Connectivity Guide provides an overview of the host USB software stack, and the logic/firmware necessary to create a USB reader/writer for Miniature Cards. Refer to Appendix A for a list of third-party vendors for USB connectivity solutions.

5.1 USB Hardware

There are many ways to implement a USB reader/writer. However, one can be easily implemented using an Intel 8X930AX USB Controller and some minimal logic (see Figure 10). The 8X930AX is an ideal device because it can support multiple devices (keyboard + reader/writer).

The 8X930AX is an 8-bit multiplexed controller with 256 KB of addressable memory. The Miniature Card is a 16-bit nonmultiplexed memory card with up to 64 MB of addressable memory. These architectural differences require the reader/writer design to implement two latches.

Demux latch: One 8-bit latch is required to demultiplex the address/data bus. At the beginning of an access this design latches A[7:0] from the 8X930AX. A[7:1] from the latch connects to A[6:0] of the Miniature Card. A_0 is used to properly assert the Miniature Card's CEL# and CEH# signals.

Page Latch: A second 8-bit latch is required to page another 8 addresses to the Miniature Card. The second latch holds A[23:16] (note that to address all 64 MB of the Miniature Card, this latch must hold A[24:16]). These 8 bits represents a 64-Kbyte "page" of memory. The 8X930AX determines which page the data addressed is associated with and writes that page number to this latch.

The firmware is about 11 KB and can reside on the 8X930AX internal masked ROM or an external nonvolatile device. An external flash device must be used if the USB reader/writer requires support for future updateable memory technology drivers (MTDs).

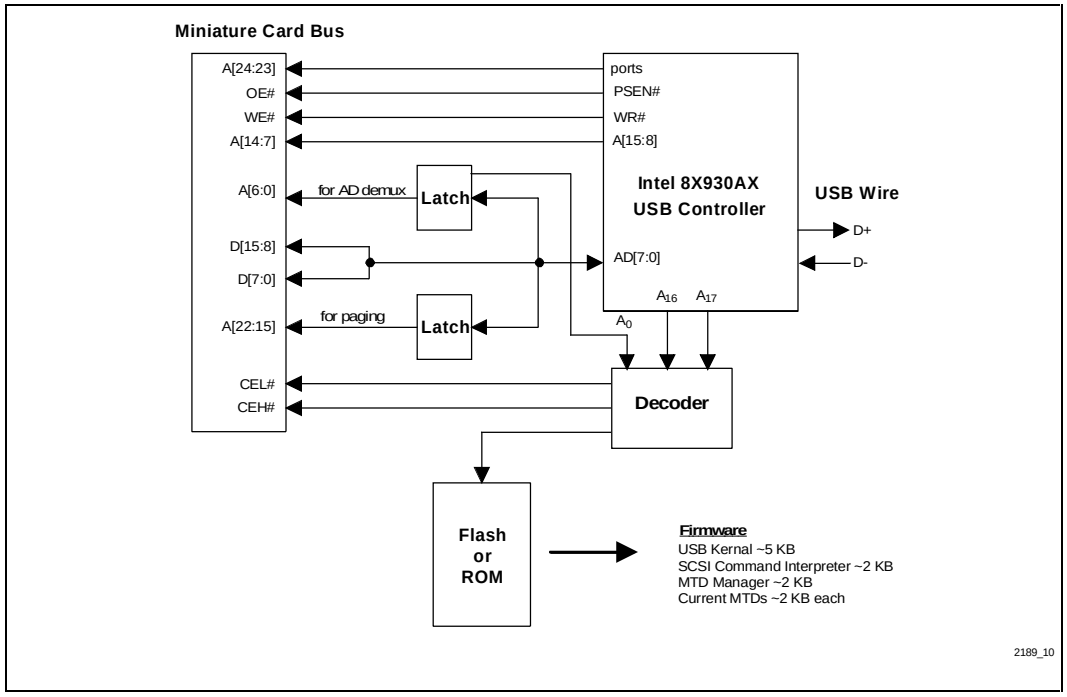

Figure 10. USB Reader/Writer Block Diagram

5.2 USB Reader/Writer Firmware

Refer to Figure 11 for a block diagram of the USB Reader/Writer Firmware Architecture.

5.2.1 FIFOS

The FIFOs are located within the 82930A. They temporarily hold the data that is transmitted to and from the USB wire.

5.2.2 USB KERNEL

The USB kernel consists of two elements.

1. Enumerator/Power Manager: The Enumerator informs the host of the reader/writer's capabilities on insertion or power-up. The host uses this information to configure the reader/writer and prepare itself to use the reader/writer. The Power Manager makes sure the reader/writer powers down when not in use.
2. Pipe Manager: The Pipe Manager monitors the USB pipe. It checks the byte count that is transmitted to and from the FIFOs and receives SCSI commands.

5.2.3 SCSI COMMAND INTERPRETER

The SCSI Command Interpreter is responsible for interpreting the SCSI commands using the MTDs. The interpreter also puts data into packets to be sent over the USB wire and informs the Pipe Manager that the data is ready to be sent.

5.2.4 MEMORY TECHNOLOGY DRIVERS (MTD)

The MTDs are the drivers that include the programming and erase algorithms for the different flash technologies.

5.2.5 UTILITIES

The Utilities manage the paging of the flash array to the 8X930AX. The utilities determine which page the addressed data is located and makes sure the page latch contains the correct page.

5.2.6 MTD MANAGER

The MTD Manager picks the correct MTD for the flash memory on the card that has been inserted into the reader/writer. The MTD Manager is also responsible for handling undesirable conditions caused by a power loss during an MTD update.

5.3 USB PC Software

Refer to Figure 12 for a block diagram of the USB software stack. This software stack implements the Microsoft defined Win32 Driver Model (WDM).

5.3.1 FTL

As indicated in Section 3.0, FTL is responsible for the file management of the data in the Miniature Card. This driver passes requests like read and write to the Mass Storage Device Driver.

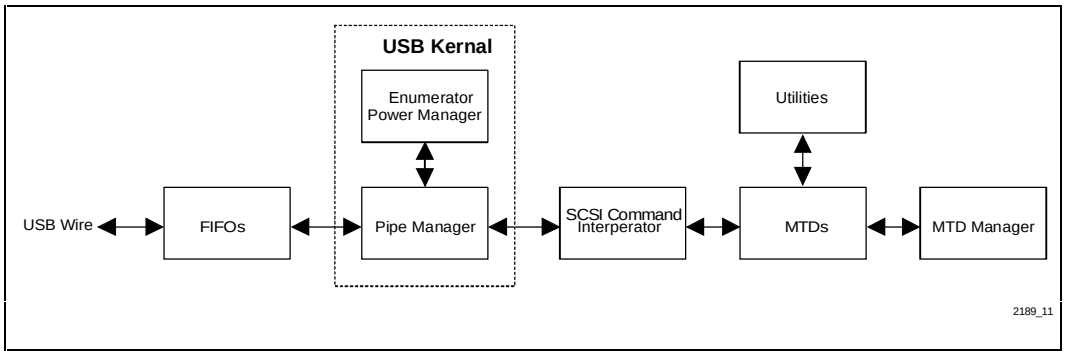

Figure 11. USB Firmware Architecture

5.3.2 LINEAR FLASH CLASS DRIVER (LFCD)

This driver implements a SCSI-2 interface converting the requests from FTL into the SCSI-2 twelve byte command block and then routes the command to the USB Driver. This driver only supports linear flash cards.

5.3.3 USB DRIVER (USBD)

The USBD provides the command and pipe interface services to the host system through the USB Host Controller. Some of these services include retrieving a list of all the USB devices on the bus and processing USB Request Block (URB) structures.

5.3.4 HOST CONTROLLER DRIVER (HCD)

The HCD provides the lowest layer of software control over the hardware host controller. The HCD functions as a hardware abstraction layer enabling upper layer software to write to the HCD layer without having to worry about which hardware control logic the PC designer implemented.

6.0 LEGACY PC CONNECTIVITY SOLUTIONS

The PC Card adapter described earlier is targeted at notebook PCs since most desktops do not have PC Card slots. The USB connectivity solution described earlier is targeted at future PC since there is a large installed base of PCs without USB slots. For the installed base of PCs, other means of PC Connectivity are necessary. This section briefly describes other PC Connectivity solutions for the installed base of PCs, referred to as legacy PCs.

6.1 Serial Cable

All PCs have serial ports (comm ports). These ports meet the industry-standard RS232 specification. The maximum transfer rate of this solution is 57.6–115.2 Kbits/sec. Many embedded applications today support a serial connection directly on the device. These devices come with proprietary software to download the data back to the PC. This solution can be tedious due to the length of time required to download data and the time required to search around the back of the PC for an available serial port.

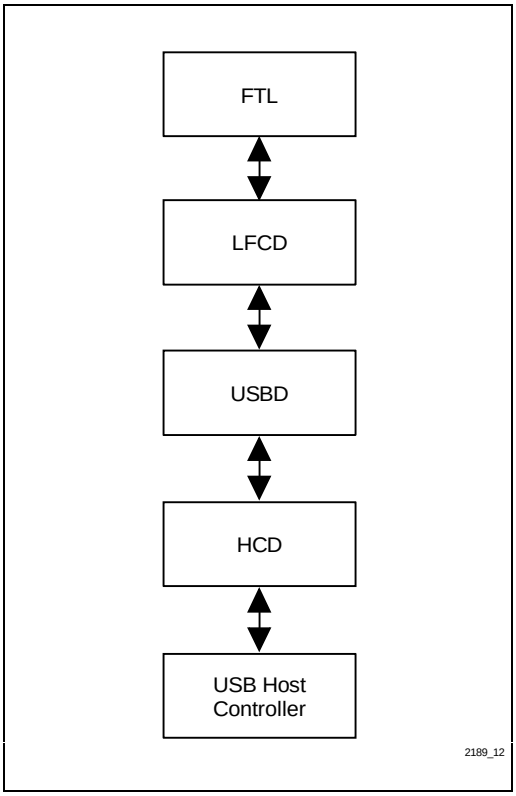

Figure 12. USB Software Stack

6.2 Parallel Port

The parallel port transfers data 8 bits at a time compared to 1 bit for a serial cable. Older parallel ports only support a nibble mode which can sustain read throughput of 300–800 Kbits/sec. However, newer PCs have an enhanced parallel port that can sustain a read throughput of 1.6 Mbits/sec.

6.3 Add-in Card

Another solution is to add PC Card or USB functionality to the installed base of PCs that require connectivity to Miniature Cards. There are several vendors that provide PCI or ISA add-in cards enabling consumers to add PC Card or USB functionality to their PC.

APPENDIX A
PC CONNECTIVITY
THIRD-PARTY VENDORS

Product	Company
FTL	M-Systems
	Microsoft Windows*95
	SystemSoft
	SCM Microsystems
	Phoenix Technologies Ltd.
	Kanematsu Electronic Components Corp.
	Annabooks
FTL-Lite	M-Systems
	Phoenix Technologies Ltd.
PC Card Adapter	Mitsubishi Plastics, Inc.
	SCM Microsystems
USB Reader/Writer	SCM Microsystems
USB Keyboard with Miniature Card Slot	Cherry Mikroscholter
Parallel Port Reader/Writer	SCM Microsystems

Company	Contact Numbers
M-Systems	USA (408) 654-5820 Israel 972-3-647-7776
Microsoft	USA (800) 426-9400
SystemSoft	USA Santa Clara, CA (408) 988-6756 USA Natick, MA (508) 651-0088
SCM Microsystems	USA (408) 370-4888 Germany 49 (8441) 896-0 France 33 (76) 25 87 48 Singapore 65-841 52 33
Phoenix Technologies Ltd.	USA (408) 452-6528 Japan (03) 3374-6555
Kanematsu Electronic Components Corp.	Japan (03) 3779-7866
Annabooks	USA (800) 462-1042

APPENDIX B
ADDITIONAL INFORMATION

References

The following are other sources of information referenced in this guide.

PC Card Standard (February, 1995)

Miniature Card Specification (Release 1.0, February 29, 1996)

Acknowledgments

Many thanks to Thomas J. Newman, Mission Peak Designs, for his help writing the PC Card Connectivity section of this document.

Intel-Related Documentation(1,2)

Order Number	Document
290581	*Series 100 Flash Memory Miniature Card Datasheet*
272797	*8X930AX USB Microcontroller Datasheet*
272811	*8X930AX USB Microcontroller User Manual*

NOTE:

1. Please call the Intel Literature Center at (800) 548-4725 to request Intel documentation. International customers should contact their local Intel or distribution sales office.
2. Visit Intel's World Wide Web home page at http://www.Intel.com for technical documentation and tools.

Specifications

Specification	Web Site
Miniature Card Specification	http://www.mcif.org/
USB Specification	http://www.teleport.com/~usb