

# THE SCSI DRIVE INTERFACE

## Mark E. Donaldson

### SCSI-1 Specifications

- Asynchronous, 2.5MBps, 8-bit data transfer bus
- 7 daisy-chainable devices per bus
- Programmed input/output (PIO) controllers
- Single-tasking operation with one initiator & one target
- Maximum cable length of 6 meters
- No support of Fast SCSI devices
- Requires DB-25 or Centronics-50 to Centronics-50 unshielded cable
- Supports single-ended electrical specification

### SCSI-2 Specifications

- Downward compatible with all SCSI-1 devices (see SCSI-1 standard above)
- Supports "fast" (>10MBps) and "wide" (16 or 32 bit) data transfers
- Supports bus-mastering controllers (on-board CPU & DMA controller)
- Supports multi-tasking environments
- Supports both synchronous and asynchronous data transfers
- Supports "fast" synch data transfers (10Mbps) (SEE NOTE #1 BELOW)
- Supports "wide" 16-bit transfers of 20MBps with two data cables
- Supports "wide" 32-bit transfers of 40MBps with two data cables
- Adopted common command set (CCS) standard for peripherals
- Specific support for CD-ROM, DAT, floptical, removable disk, tape drive magneto-optical, worm drives, and scanners (some may require 3rd party software drivers)
- Requires Centronics-50 or high density SCSI-2 to Centronics-50 shielded cable
- Supports single-ended and differential electrical specifications
- Maximum cable length of 6 meters (single ended) or 25 meters

### SCSI-3 Specifications

- Downward compatible with all SCSI-1 and SCSI-2 devices (see specs above)
- Supports up to 15 daisy-chainable target devices
- Supports SCSI serial interface using fiber-optic cable with 1 GBps data transfers
- Supports "fast" 8-bit synchronous data transfer of 10MBps
- Supports "fast-wide" 16-bit data transfer of 20MBps with one data (P) cable
- Supports "fast-wide" 32-bit data transfer of 40MBps with two data cables
- Supports all known SCSI peripherals
- Requires high density SCSI-2 to Centronics-50 or high density 68 pin SCSI-3 (P) cable nbsp
- The "fast" SCSI-2 transfer rate of 10MBps requires an uncommon differential configuration. Most vendors sell single-ended "fast" SCSI devices that are only capable of 5MBps.
- The "wide" SCSI-2 feature requires special cabling. "Wide" usually refers to SCSI-3 configurations.

# THE SCSI DRIVE INTERFACE

## Mark E. Donaldson

### **SCSI-2**

This is a term describing the latest published ANSI standard (X3.131-1994). This document describes several connectors (both shielded and unshielded) that include 1 byte wide data bus, defines FAST transfer speeds, defines SCSI protocol for wider data transfers, defines the parallel SCSI messages, and command structure. This provides the base on which future SCSI features are compared against.

### **SCSI-3**

This term describes a set of related standards that are currently being developed. The SCSI-2 document is very large (400+ pages) and covers the full range of topics. SCSI-3 split this large document into a series of smaller documents that each covers a "layer" of the interface definition. The basic layers are: physical (connectors, pin assignments, electrical specifications), protocol (description of how physical layer activity is organized into bus phases, packets, etc.), architecture (a description of how command requests are organized, queued, responded to by any protocol), primary commands (description of commands that must be supported by all SCSI devices), and device specific commands (commands that are specific to a particular class of devices; CD-ROMS or WORM drives, for example).

The set of standards needed to do a SCSI-3 parallel interface disc drive implementation is SPI (SCSI Parallel Interface) for the physical layer, SIP (SCSI Interlocked Protocol) for the protocol layer, SAM (SCSI Architecture Model) for the architecture, SPC (SCSI Primary Commands) for the primary command set, and SBC (SCSI Block Commands) for the disc drive specific command set.

The SCSI-3 standards are layered in this manner to allow substitution of parts of the structure as new technology emerges. For example, a comparable set of standards for a SCSI Fiber Channel interface disc drive replaces the physical and protocol layers with new documents but uses the same documents for the other 3 "layers". The main point to remember here is that the terms SCSI-2 or SCSI-3 don't imply any particular performance per se -- it refers to the generation of documents that a product conforms to. Since the newest features are only in SCSI-3 and they tend to be higher performing, however, SCSI-3 devices should be better performing.

### **SCSI FAST**

This refers to timings defined in SCSI-2 for 10 MegaTransfer/sec transfer rate. A "MegaTransfer" refers to the rate of signals on the interface regardless of the width of the bus. For example, 10 MT/sec rate on 1 byte wide bus results in 10 MB/sec transfer rate but on a 2 byte wide bus results in a 20 MB/sec transfer rate.

### **SCSI FAST-20**

This refers to timings defined in SCSI-3 physical document for 20 MT/sec transfer rate. This achieves data rates twice as fast as SCSI FAST rates.

### **SCSI FAST-40**

This refers to timings being defined for a future revision of the SCSI-3 physical documents that achieves 40 MT/sec.

### **Ultra SCSI**

This is an old term for the FAST-20 data rate. This term was dropped by the committee because the company UltraStore owned a trademark on the term Ultra SCSI, so it could not be used by the standards committee to describe its work. If you see this term anywhere, read carefully to see if it refers to the UltraStore trademark (which refers to a SCSI FAST class of product) or if the writer is incorrectly using this term for FAST-20.

# THE SCSI DRIVE INTERFACE

## Mark E. Donaldson

### **SCSI WIDE**

This term usually refers to the two byte wide (68 pin) connector that is defined in the SCSI-3 Parallel Interface (SPI) document. This technically makes it a SCSI-3 feature. The term can be generically applied to any implementation wider than 1 byte, but there are no implementations wider than 2 bytes today. I don't expect wider implementations because faster transfer rates are giving plenty of life to 2 byte transfers until serial interfaces (like Fiber Channel or FireWire) become more popular.

### **SCSI FAST-WIDE**

This refers to a combination of FAST transfer rate with 2 byte wide connector, which results in 20 MB/sec data transfer rate. We had wide FAST-20 (40 MB/sec) products in 1995 and wide FAST-40 (80 MB/sec) products were available in late 1996.

### **Fiber Channel SCSI**

This refers to products that use a fiber channel physical and protocol characteristics with SCSI command set. The interface is completely different than parallel SCSI. It is a serial interface, meaning command and data information is transmitted on one signal. That signal uses a 1 GHz rate, however, so it achieves 100 MB/sec over coaxial cable. Even faster rates are possible if optical fiber is used, but the optical transmitter/receiver is currently too expensive for disk drive use. Information over fiber channel is organized into packets. This interface has more in common with local area networks than with parallel SCSI.

### **ASA-2 SCSI**

This is a Seagate specific term that describes the basic structure of the SCSI firmware included with a Seagate disc drive. It stands for Advanced SCSI Architecture, generation 2. Without going into the gory details, the main idea is that it provides better performance in certain user environments (e.g., sequential 1 block data transfers) than ASA-1 code did. These labels (ASA-1 or ASA-2) can apply to code shipped on a wide range of products. That means that these different products have firmware that come from the same base firmware but have changes to adapt to the particular servo and read/write channel hardware on that product. Many different products from the Hawk line (designed at OKC) and the Barracuda line (designed at TCO) come from the same code base, which is designated by the ASA-1 or ASA-2 label. The performance of a disc drive is still determined mostly by the seek times and data transfer rates, but the code base label provides a level of commonality and lack of bugs in SCSI features for all drives with that label.

The most attractive feature of a common code architecture is that any enhancements made on one drive (eg, a Barracuda 4) can quickly be migrated to other drives (eg, Barracuda 2, Hawk 4, etc.) that is developed from the same code base. This capability started with ASA-1 code (introduced about 2 years ago) and worked well. ASA-2 has a different structure from ASA-1, so there is more re-development needed when migrating an improvement from one code base to the other. All new products starting January 1995 will use the ASA-2 code base and many older products developed with ASA-1 code have been updated to the new structure.

### **SCSI (For Servers and Power Users)**

SCSI provides a standard interface for all types of computers. The IDE disk and the ISA bus are peculiar to IBM-compatible Intel-compatible PC machines. SCSI, however, is used by Macintosh computers, RISC workstations, minicomputers, and even some mainframes. SCSI has always supported a mixture of disks, tapes, and CD-ROM drives. While EIDE disks may go up to one gigabyte, SCSI disks are

# THE SCSI DRIVE INTERFACE

## Mark E. Donaldson

available with 4 to 9 gigabytes of storage.

SCSI is a bus. In the SCSI architecture, the PC (or more precisely, the SCSI adapter card in the PC) is just one device on the bus. Each device is a "peer" of the other devices. In theory, a tape drive could send commands to the PC. In practice, the tape drive isn't smart enough and the PC doesn't respond to commands anyway.

In the **Classic SCSI bus**, there are 25 signals, each represented by a pair of wires (50 wires all together). Nine of the wires hold the eight bits plus parity of a byte of data. The other wires carry control functions. Classic SCSI can transfer data up to **5 megabytes per second**. The **Fast SCSI option of the SCSI-2** standard allows **10 megabytes per second** on the same cable. To run faster, a **Fast Wide SCSI interface** is defined, but it **requires more than the usual 50 wire cable**.

An IDE disk must be mounted inside the computer. There is no provision for the IDE ribbon cable to run to external devices. SCSI devices can also be internal. They are connected to each other and to the adapter card using a flat ribbon cable with 50 wires (OK) or a round bundled cable with 25 twisted pairs of wires (Better). However, SCSI devices can also be external to the computer. They can be mounted in individual boxes, or can be mounted together in larger tower enclosures. The adapter card is connected to external SCSI devices with a round cable containing 25 twisted pairs of wires. Four external SCSI plugs are in common use:

- **Apple has a cheezy, sleazy 25 pin "D Shell"** connector on the back of each Macintosh. This is the same type of plug used on a modem. Obviously you cannot connect 50 wires to 25 pins. The sleazy part is that Apple takes all 25 ground wires and combines them. This eliminates the protection that would be provided by a proper cable with 25 twisted pairs.
- **Older SCSI devices** use the **50 contact Centronix interface** (similar to the cable that plugs into a PC printer. This was the standard in the early years of SCSI, but it is bulky and has fallen out of favor.
- **Newer devices use a smaller special SCSI connector with 2 rows of 25 pins**. The newer **SCSI-2** standard specifies this as the preferred connector. This is **frequently known as a 50 pin mini-"D" shell or MDS50** connector.

On serious large server machines there may be Fast-Wide SCSI cards. They can provide higher data transfer rates, but they may require special cables since they have more than 50 wires.

A new standard has just been adopted (Fire Wire) which has many advantages over IDE or SCSI. It will become important in future desktop systems as soon as the vendors figure out how to manufacture adapters and devices at low cost.

In both IDE and SCSI, the adapter or controller sends a signal to the device over a cable with a separate wire for each bit of data. Data is transferred one or two bytes at a time. This is called a parallel interface. Ultimately, a parallel connection is limited by skew.

At the beginning of a 100 meter dash, the runners all line up evenly at the starting line. But they cross the finish in a ragged pattern. Electrons rush down a wire at almost the speed of light. However, when there are several wires running any distance between several devices, the signals don't all get to the end of the wire at exactly the same time. This is called skew.

# THE SCSI DRIVE INTERFACE

## Mark E. Donaldson

Part of the problem are irregularities in the wire itself. One wire may be thicker than another, and different batches of copper have different amounts of impurities. Part of the problem comes from differences in individual connectors every time the signal passes through an attached device. Skew limits how tightly the bytes can be packed on the cable. All of the bits for one byte must arrive at the end of the cable before any bits for the next byte can arrive. Engineers have to design for the worst case, so they generally leave a big gap between bytes. As the cable gets longer and the number of attached devices grows, the gap has to get wider and the speed of the data goes down. Worse, because skew is caused by low tech issues (refining copper, making wire, and building cheap connectors) it is not possible to solve this problem by building better computer chips.

Advances in chip design now make it possible to pack successive bits very tightly on a pair of copper wires. The cable problems that produce skew don't effect this. Variations in the wire may cause electrons to slow down in one section of the wire, but then the next bit will also slow down at the same point. Today it is generally possible to reliably transmit more than eight bits of data on a pair of wires inside the gap required to solve the skew problem when wires run in parallel. In computer jargon, it is better to use a serial interface (one pair of wires) than a parallel interface (eight pair of wires). Over a short distance, and in a controlled environment, the parallel signal arrangement is best. The CPU, memory, and I/O bus structures use a parallel signal. Even the cable from the EIDE controller to the devices is short and internal to the machine and works well in parallel.

The advantage of SCSI, however, has been the ability to connect to external disk and tape devices over several meters of cable. An external SCSI cable is bulky and fairly expensive. Unplugging a SCSI device when the power is on can crash the system.

**Firewire (1394)** is a serial alternative to the parallel SCSI connection. It was originally developed by Apple Computers, but was then proposed and accepted as a new international standard. Firewire uses two pair of wires for signals in both directions and one pair of wires to provide power to external equipment. Up to 63 devices can be connected in a loop or from the spokes of a hub. Devices can be plugged and unplugged when the system is running. Although the physical connection is different, Firewire uses the same commands as SCSI and may not require major reprogramming for existing device drivers.

Firewire is designed to be simple and low cost. In one of its more visible decisions, it connects to devices using a plug that was originally developed for the Nintendo GameBoy. Data transfer is as fast as SCSI, but the interface is much simpler.

Microsoft has proposed Firewire as part of its SIPC initiative. The SIPC computer comes in a simple basic configuration with no internal card slots or options. The system is extended by connecting external option boxes like hard disks. Unless unexpected problems develop in the engineering, computers should begin to appear with **1394 connectors** sometime in early 1997.

Firewire is specifically designed for desktop and laptop systems. The same laws of physics may drive the redesign of connections for large dedicated file and disk servers, but industrial strength applications may deserve stronger engineering than the Nintendo GameBoy plug. IBM has proposed something called SSA. It has the same basic idea, but can run faster, support more devices, and provide better error recovery than Firewire. IBM uses it on its RS/6000 machines, but it is not clear at this point if it will be accepted for use by high end PC Servers.

# THE SCSI DRIVE INTERFACE

## Mark E. Donaldson

### EIDE or SCSI Today

Today, EIDE disks are a high-volume product, but SCSI remains unusual. A 1, 2, or 4 gigabyte hard disk can cost twice as much with a SCSI connection as a EIDE disk of comparable speed and performance. A SCSI adapter is already a requirement for special devices, like the Iomega 1 gigabyte removable Jaz drive. However, even when a system already has a SCSI adapter card, the cost per disk is so much higher that EIDE is the technology of choice for desktop systems.

- There is no meaningful difference in performance between a single EIDE disk and a single SCSI disk on the same system.
- In many cases, EIDE is faster.
- SCSI is worth the extra cost in a Server.
- EIDE supports two separate I/O operations to two disks (on the two different interface cables).
- SCSI allows all of the disk devices to be active simultaneously. Of course, only one device can be transferring data on the SCSI cable at any given time. However, a disk spends most of its time moving the arm to the right location and waiting for the data to rotate around to the point where it can be read or written. A SCSI controller can have all of its disks moving into position while one disk is actively transferring data.

It goes without saying that a good disk interface on a modern server will connect to the PCI bus. The ISA bus is unreasonably slow (by modern terms), the Microchannel is expensive, and EISA is now obsolete. However, there are both IDE and SCSI adapters that interface to the PCI bus.

An EIDE adapter will always be dumb and cheap. A SCSI adapter can be smart enough to Busmaster. As a Busmaster, the SCSI card can transfer data to or from buffers in memory directly. This frees up the CPU to do other things. To get the full benefit, the computer must be running an operating system (Windows NT, OS/2, Netware, or Unix) that can take advantage of the full capability of the card. Vendors may offer a slightly lower price on a SCSI adapter card that doesn't Busmaster, but the \$30 savings are not worth it if the machine will act as a server or will run a multitasking operating system like Windows NT.

As Firewire becomes available, the difference between the two technologies is likely to blur. Firewire also supports an external connection to plug and play disk modules. Firewire protocol is an extension of current SCSI commands. However, it has not been lost on anyone that IDE disks are much cheaper. So new chips are being designed to translate between IDE disks and the SCSI-like Firewire system. It is expected that as Firewire systems become available, vendors will offer standalone external expansion modules with four bays, a power supply, an EIDE controller logic chip to run the disks, but an external SCSI-Firewire interface. This provides the best of both systems, with the low cost of EIDE devices and the high function and modularity of SCSI connections.

### SCSI Technology

In short, allows up to 63 external devices to be chained together forming a bus, and controlled from a single PC at high speeds. Because SCSI is a well-established standard with a set number of simple, low-level hardware control commands that aren't platform specific, it offers benefits that can't be found in traditional bus architectures:

- up to 63 devices/logical units (LUNs) are independently addressable, versus the 2 to 12 normally found in desktop buses

# THE SCSI DRIVE INTERFACE

## Mark E. Donaldson

- SCSI devices are external, making them easily configurable, portable, and repairable devices can be inserted or removed from a SCSI bus chain simply by plugging or unplugging them -- no removal of a PC's metal casing two or more PCs can enter the SCSI bus chain and share any device(s) that are attached developers are able to write software drivers for new devices in days instead of months, allowing faster development & troubleshooting times.

There are thousands of SCSI devices available, and all can be readily used in any SCSI bus with little regard for the make of the PCs attached:

- polygraphs (proprietary)
- scanners & cameras (HP, Canon, Fujitsu)
- tape backup drives (Colorado)

Because there is no IEEE or ANSI parallel-to-SCSI standard per se, one might call this technology a hybrid discipline that marries both parallel and SCSI knowledge together. Therefore, the following sections detail parallel port and SCSI technology separately, and in more detail. The two last sections, describe popular parallel-to-SCSI applications, how such converters work across different types of platforms, and finally, how this exciting technology promises to continue making connectivity easier for portable PC users.

### **Making the Most of Parallel Port Technology**

IBM co-introduced the parallel port in 1981 as an alternative to the slower serial port that was often used to drive dot matrix printers. Making use of 4 Control Lines, 5 Status Lines, and 8 Data Lines, parallel technology had the ability to transmit 8 bits (1 byte or character) from a PC to a peripheral at a time, whereas the serial port was limited to a single bit (1/8th of a character).

Although the performance of PCs has increased a thousandfold since the early eighties, there were virtually no changes made in parallel port architecture until the last few years. Data transfer rates remained at around 150KB/second. Another problem was that the industry lacked a standard for the electrical interface, which made guaranteeing the compatibility of a printer or other parallel device from one platform to another an impossible task. Finally, a lack of design standards forced a distance limitation of 6 feet for external parallel printer cables.

In 1991, a number of companies, including Lexmark, IBM, Texas Instruments, and others, formed the Network Printing Alliance (NPA) in an effort to develop an intelligent standard for parallel port control. After about three years of study and attempts to influence the IEEE to release an updated printer port standard, the Standard Signaling Method for a Bi-directional Parallel Peripheral Interface for Personal Computers was finally completed. Also known as the IEEE 1284 Parallel Printer Port Standard, the new set of specifications defines:

- 5 modes of parallel data transfer
- a method to negotiate between the different modes
- the physical interface
- electrical & cabling requirements

Parallel ports were thereafter said to operate in five different modes:

- Forward Compatibility Mode

# THE SCSI DRIVE INTERFACE

## Mark E. Donaldson

- Reverse Nibble Mode (4-bits)
- Byte Mode (8-bits)
- Bi-directional Enhanced Parallel Port (EPP)
- Extended Capabilities Port (ECP)

The first two modes, Forward and Reverse are software-driven protocols, which means that a software program or low-level device driver has to execute multiple I/O instructions in order to transfer a single byte of data to or from a parallel peripheral. The two Bi-directional parallel port types, Enhanced and Extended Capabilities, use hardware shortcuts to reduce the number of software commands required to transmit bytes and monitor transmissions, thereby increasing overall parallel port data transfer speeds

by as much as 15:1. The 1284 recommends that a parallel port's memory registers reside as a continuous block of 3 registers from the familiar I/O base addresses of 3BCh, 378h, and 278h. Many PCs manufactured after mid-1993 are equipped with newer implementations of the EPP and ECP parallel ports, with I/O addresses at 378h, 278h, or user-definable locations.

**Compatibility mode** (also known as Centronics mode) is utilized by the most basic parallel ports. Data is sent along the port's data lines, the printer status is checked for errors and that it is not busy, and a data strobe is generated by the software to clock the data into the printer. The output of a single byte (1 character) requires at least four I/O instructions, and at least as many additional instructions if other features are required. The net effect of this limitation on bandwidth is a speed on the order of 150KB/second.

**Nibble mode** is the most common way to achieve "reverse channel data" from a printer or other peripheral to the PC, which is useful for diagnosing problems, monitoring a device's status, etc. Using two data transfer cycles to get a bit on its way, this mode is usually combined with the compatibility mode or a proprietary forward channel mode to create a complete bi-directional channel. Although this mode is guaranteed to work on nearly all parallel port-equipped PCs, it has a severe speed limitation of 50KB/second when used by itself.

**Byte mode** adds the capability to disable the drivers that are normally utilized for driving the data lines, allowing them to become an input read data port (to the PC). This enables a peripheral to send an entire byte of data home in one data transfer cycle by using the 8 data lines, rather than the 2 cycles required by the Nibble mode.

The **Enhanced Parallel Port (EPP)** and **Extended Capabilities Port (ECP)** provide a mechanism for a fully interlocked data transfer within a single ISA I/O data transfer cycle. When an I/O instruction is executed, the port hardware automatically synchronizes the external peripheral and generates the necessary handshake signals required before, during, and after the transmission of the byte signal. Because software is no longer required for data maintenance, transmissions take place in one I/O cycle, and all handshaking is automatic, EPP and ECP ports can reach data rates from 800KB to over 2MB/second, which is fast enough to drive multimedia CD-ROMs and other speed-intensive devices.

### Going SCSI

SCSI is an acronym for Small Computer System Interface . First used as a bus technology on IBM-360 minicomputers, it was scaled down by the Shugart Company in 1981 for use as a universal disk drive interface. In 1986, after nearly four years of committee discussions, ANSI pronounced it an international

# THE SCSI DRIVE INTERFACE

## Mark E. Donaldson

bus standard for interconnecting personal computers and peripheral devices of all kinds together. Simply put, SCSI is a specific hardware/software standard that allows one or more external peripheral devices ("targets") to be chained together into a bus, which is then controlled by one or more PCs ("hosts" or "initiators").

Each device, including the PC's SCSI host adapter that marks the PC's entry into the chain, uses a unique number to identify itself to the rest of the components on the bus. Up to 8 devices can exist on a single bus (including the initiator). Some devices, like CD-ROM changers or printers that can also fax and copy, are said to have more than one "logical unit" or LUNs (each unit being one feature or partition of the device). A single device on a SCSI bus--including the initiator--can branch off from the bus to control up to 7 additional independent LUNs. At its fullest, then, a SCSI bus can contain up to 63 independently addressable units: 7 devices, each with 8 LUNs (including the original device itself), plus the 7 LUNs that could branch directly off from the SCSI host adapter that is attached to the initiating PC itself.

Without SCSI, each time a new peripheral is added to a PC, a highly specialized, low-level device driver has to be written for it. The driver controls not only the device's interface card, but also the device itself. Unfortunately, the development cycle for programming and testing the software can range anywhere in this traditional method from one to six months. As hardware capabilities grow at increasing rates, it takes even longer to develop the computer/peripheral interfaces. As a result, it isn't surprising to see peripherals attached to a PC that are one or more generations behind the PC itself, to say nothing of the software drivers running the peripheral(s).

The development and testing time for SCSI peripherals is extremely short, <U>often lasting only days or weeks . Because the SCSI interface is consistent across all platforms and requires only a few simple commands with which to drive any attached device(s), a developer can often re-write an old driver for use with a new peripheral with a minimum of effort. This technique returns independence to the computer--in other words, all magnetic disks look alike to the SCSI bus except for their capacities. All printers look alike, regardless of whether they're dot matrix, bubble jet, or laser. Within any one kind of device, the SCSI bus and PC system should not need any modifications when removing one manufacturer's device and replacing it with another (of course, there are still small variations from one parallel port to another, and from one manufacturer's interpretation of the SCSI standard to another, which can make interfacing a PC with a SCSI device interesting--see the next section for more information).

There three SCSI standards to date. **SCSI-1**, the original SCSI standard, has been implemented in most SCSI devices manufactured before 1993. Newer SCSI devices follow the **SCSI-2** standard, which supports a greater number of devices, protocols, and speeds. At the time that this text went to press, the **SCSI-3** standard had been defined, but not yet widely implemented in the manufacture of SCSI hardware and software.

### Putting It All Together

Joining the best of the parallel and SCSI worlds together has already provided tens of thousands of portable PC users with a bridge to a virtually unlimited number of SCSI devices , and not at unacceptable speeds, either. In 1993, PC Magazine (August 13) reported a non-cached, sustained data transfer rate by one manufacturer's parallel-to-SCSI converter of about 670KB/seconds, which is phenomenal because the host platform was a 33Mhz IBM PC/AT with a standard uni-directional parallel port--rather slow by 1994 standards.

# THE SCSI DRIVE INTERFACE

## Mark E. Donaldson

Today, a handful of connectivity manufacturers offer parallel-to-SCSI converters that operate at speeds much closer to 1.2MB/seconds when tested on a standard 486/33 computer that is equipped with an EPP-compliant parallel port, and speeds are increasing by about <U>500KB per year . Throw in the possibility of adding up to 62 additional devices without any noticeable speed depreciation (for SCSI ID recognition, not necessarily data reads/writes), 100% compatibility with caching software like PC-CACHE or MS-DOS SMARTDRV, and suddenly no bus alternative appears faster, more portable, or more flexible.

Most parallel-to-SCSI manufacturers build their adapters with various "smart features" that facilitate ease-of-use even further. The latest proprietary parallel and SCSI chipsets allow an external SCSI host adapter to automatically detect and optimize itself to the presence of a uni-directional or bi-directional parallel port, automatically making the most of a PC's maximum port speeds so that users aren't normally required to make BIOS or jumper settings that enable advanced parallel port speeds. Not all parallel port manufacturers build their ports so that they strictly adhere to international parallel port standards, however, which means that "smart features" aren't always successful at optimizing speeds. Various proprietary I/O modes, strange port memory address assignments, and electrical voltage variances sometimes require users to take an active role in making the most of their parallel ports' many features by setting hardware or software switches that enable the most desirable data transfer speeds.

Thus far, the two primary objections to using parallel-to-SCSI technology have been (1) speed, and (2) the hope that PCMCIA technology will one day surpass it. To date, the latter has proven unworthy of worry, and the former, unfounded. Already, the increasing speed of parallel ports and SCSI buses have allowed portable PC users to enjoy even the most complex scanners, multimedia programs, and other speed-intensive SCSI targets. In the coming years, more and more devices will begin following the SCSI-3 standard, which, although adopted formally by international agencies, has only just started to be implemented by manufacturers of SCSI hardware and software.

The so-called 1-Gig standard allows a SCSI bus to reach a data transfer speed of up to 1 gigabyte/second through the use of fiber-optic cabling, and 20MB-40MB/second with a single 68-pin "P" cable (not yet in wide circulation). Many of these factors may change, of course. As the PCMCIA standard is followed more closely by card manufacturers in the future, it should be easier to iron out compatibility and software driver problems. The most likely competitor for parallel-to-SCSI technology will probably be PCMCIA-to-SCSI when it becomes available, as it will provide an overall data transfer increase of 4 to 5MB/second (of course, users without PCMCIA slots will still be out of luck). Of all the expansion solutions available to portable PC users, only parallel-to-SCSI offers true portability, compatibility across varied platforms, the promise of continually increasing data transfer speeds, a variety of low-cost devices, and ease of development