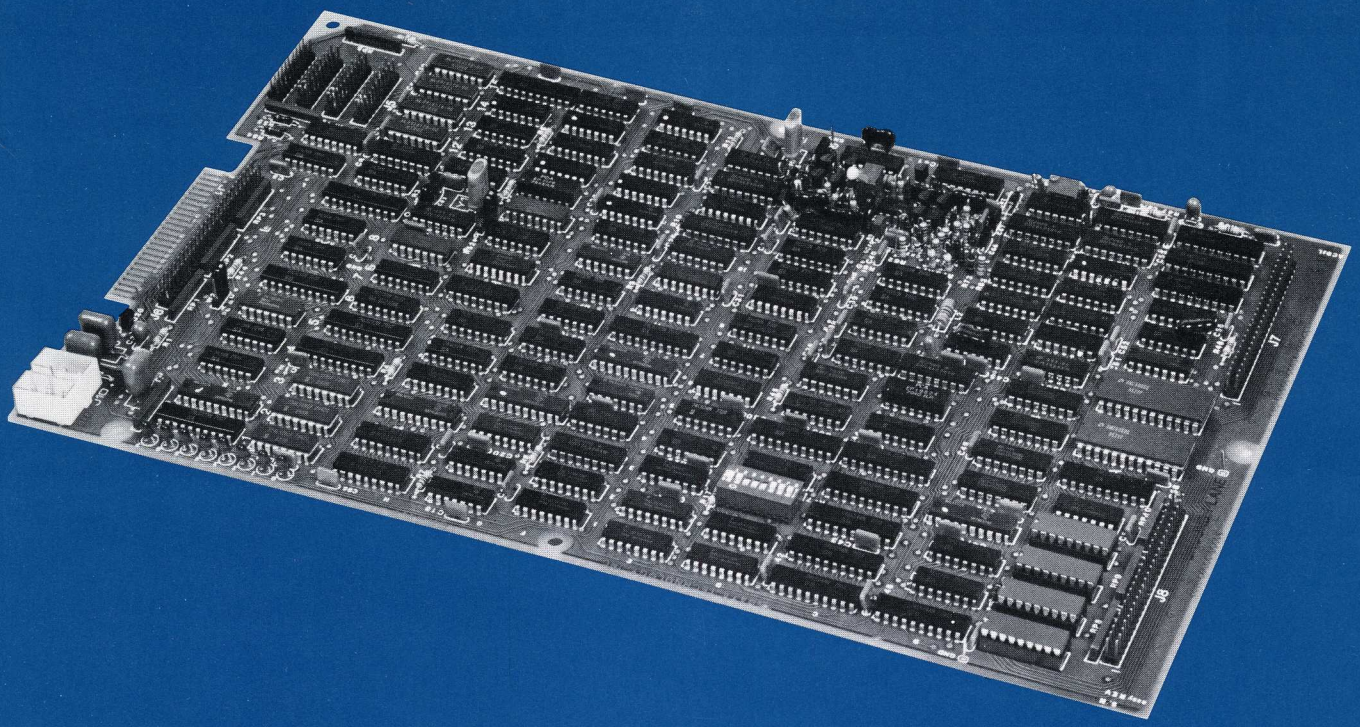
Shugart

Intelligent Disk Drive Controllers SA1400 Series



Description

The Shugart SA1400 series of intelligent controllers handle various combinations of floppy disk, fixed disk and streaming tape cartridge drives. All models include the industry standard Shugart Associates System Interface (SASI™) to the host, and industry standard drive level interfaces to the peripheral.

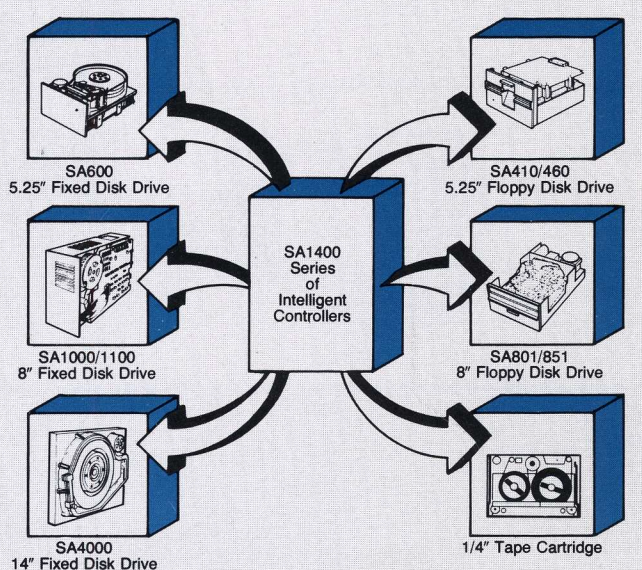
Each model of the SA1400 series contains a bit-slice microprocessor, sector buffer, data separation logic, error detection and correction capability, and plug-in diagnostic PROMs with an accompanying LED display for diagnostic read-outs.

Use of SASI as an interface between the SA1400 series and the host CPU facilitates design of hardware host adapters. In terms of software, SASI is "logical," rather than "physical," to minimize host involvement in disk, cylinder, head and sector selection. With SASI the host CPU need only send a read or write logical address and the number of blocks to be written or read, and the SA1400 series controller does the rest.

The implied seek function is performed, as necessary, at the beginning of each read and write command.

Key Features

- Controls Shugart 5.25, 8 and 14-inch Winchester disk drives alone or in combination with floppy disk drives or streaming tape cartridge drives
- Single printed circuit board
- Sector buffer
- Implied seek with verify
- Automatic soft error retries
- Error correction on fixed disk drives (up to 4-bit errors)
- Automatic switching of head and cylinder
- Sector interleaving
- Plug-in operating firmware (PROM's)
- Plug-in diagnostic PROM's
- IBM compatible floppy format as option
- ¼-inch streaming tape cartridge drive as an auxiliary
- Shugart's industry standard SASI™ (which will be used with all present and future drives)

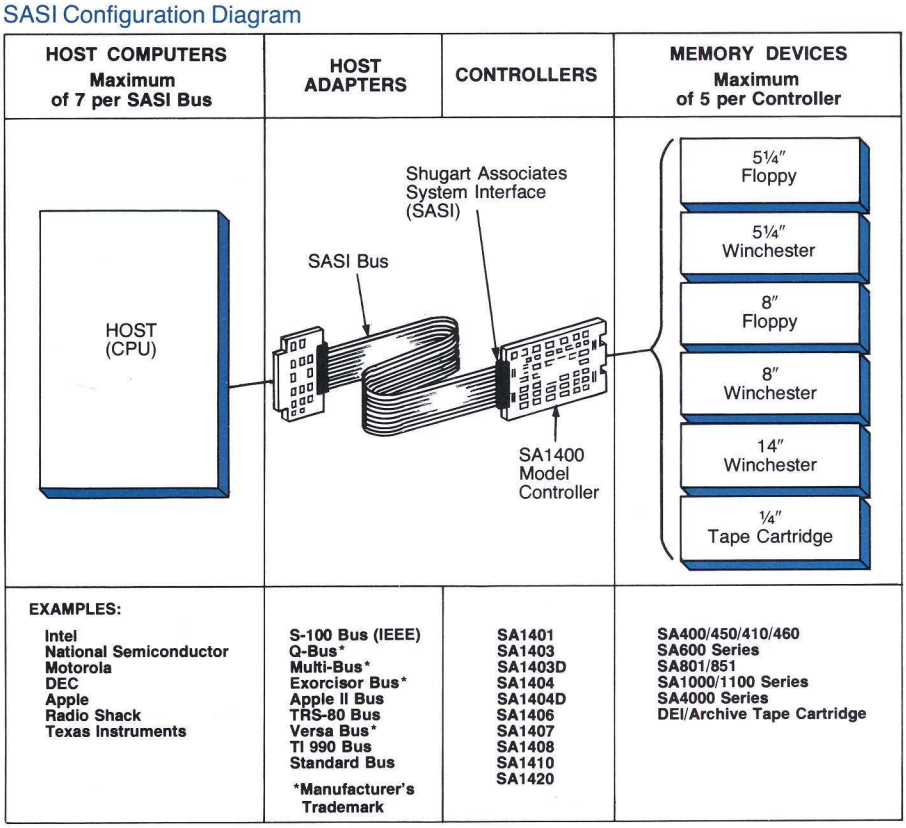


Specifications

Models and Features

	Winchester Drives			Floppy Drives		Tape Cartridges		Floppy/Tape	
	14″	8″	5.25″	8″	5.25″	¼-INCH	¼-INCH		
Drive / Con-troller	SA4000	SA1100 SA1000	SA600	SA801 SA851	SA410 SA460	DEI	Archive	Format/ Error Code	Formatted Capacity
SA1401		•							
SA1403		•		•				32 Sectors 256 bytes/sector ECC	0.6/1.2 MBytes
SA1403D		•		•				IBM 1D/2D CRC	Same as IBM 3740 & System 34 Formats
SA1404	•								
SA1404D	•			•				IBM 1D/2D CRC	Same as IBM 3740 & System 34 Formats
SA1406		•				•		OEM LRC	10/20 MBytes
SA1407	•					•		OEM LRC	10/20 MBytes
SA1408		•					•	OEM LRC	10/20 MBytes
SA1410			•						
SA1420			•		•			16 Sectors 256 bytes/sector CRC	0.4/0.8 MBytes

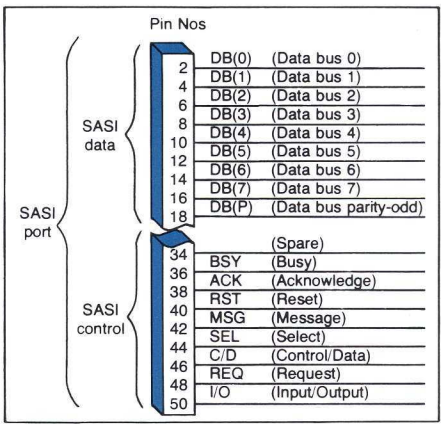
SASI Configuration Diagram



Drive Capacities

DRIVE TYPE	MAXIMUM CAPACITY (Formatted)
Floppy Disk Drives	
SA400	— 204.8 Kbytes
SA450	— 409.6 Kbytes
SA410/460	— 819.2 Kbytes
SA801	— 630.0 Kbytes
SA851	— 1,212.0 Kbytes
Fixed Disk Drives	
SA602	— 2.6 Mbytes
SA604	— 5.2 Mbytes
SA606	— 7.8 Mbytes
SA1002	— 4.2 Mbytes
SA1004	— 8.4 Mbytes
SA1104	— 16.0 Mbytes
SA1106	— 26.6 Mbytes
SA4004	— 12.4 Mbytes
SA4008	— 24.8 Mbytes
Cartridge Tape Drives	
DEI	— 20.0 Mbytes
ARCHIVE	— 20.0 Mbytes

SASI Interface Pin Assignments



The SASI bus structure consists of nine control lines and nine data lines on a 50-pin bus. All bus lines are implemented as low-true signals.

Line Definitions

DB O-P — These eight bi-directional lines plus optional parity are used to transfer 8-bit parallel data to and from a host adapter.

BSY — BSY is asserted in response to the SEL line from the host; it indicates that the SASI bus is in use.

ACK — ACK is asserted in response to the REQ line from the host; it is an acknowledgement by the controller that the host wants to perform a REQ/ACK handshake.

RST — RST resets the controller by returning the microprocessor in the controller to the beginning of its microcode.

MSG — Asserted by the controller to indicate that a command is complete. It is always followed by the assertion of I/O and REQ.

SEL — SEL is asserted by the host in order to select a controller.

C/D — When asserted, the information transmitted across the bus will be the command or status bytes (i.e., "control" information.) When de-asserted, the information will be data or sense bytes.

REQ — REQ is a request for a data transfer; it operates in conjunction with I/O, C/D, and MSG. When asserted and I/O is asserted, REQ will mean that data on the host bus is driven by the controller; when asserted and I/O is deasserted, REQ will mean that the data is driven by the host adapter.

I/O — Asserted by the controller to indicate the direction of information transfer on the data bus lines during a REQ/ACK handshake. I/O is "input" to the host when asserted.

Shugart

Headquarters: 475 Oakmead Parkway, Sunnyvale, California 94086
Telephone (408)733-0100 TWX: 910-339-9355 SHUGART SUVL
Sales/Service Offices:—West: Milpitas, CA (408)263-2600; Costa Mesa, CA (714)979-1935
Central: Richardson, TX (214)234-3568; Minneapolis, MN (612)574-9750
East Coast: Framingham, MA (617)879-1700; Saddle Brook, NJ (201)368-8445; Atlanta, GA (404)955-8968
Eastern Canada: Markham, Ontario (416)495-2655
Europe:—Paris: Telephone (1)686-00-85; **Munich:** Telephone (089)80-60-55

9/81 Printed in U.S.A.

10M S/N 207000

ElectronicDesign®

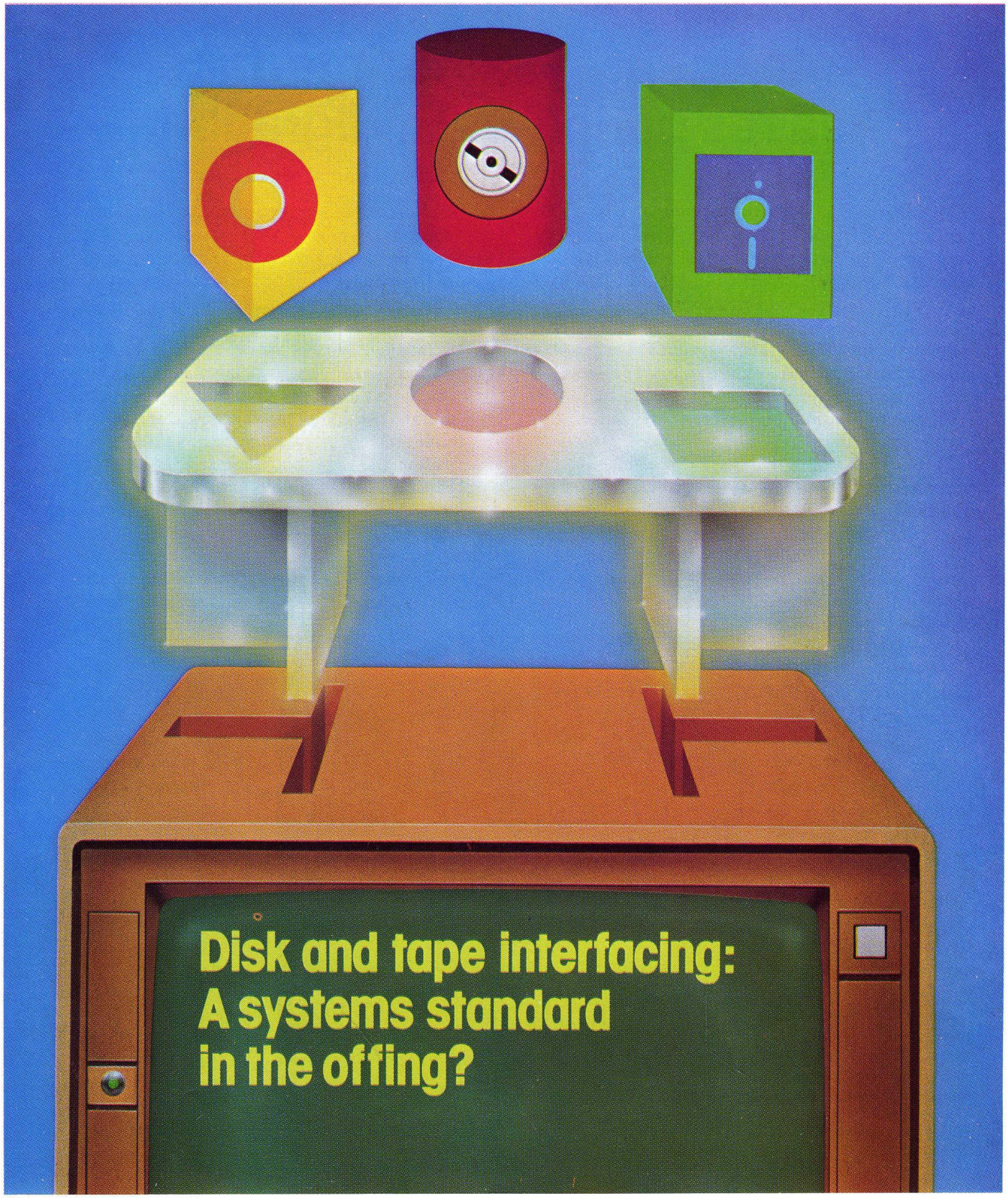
FOR ENGINEERS AND ENGINEERING MANAGERS—WORLDWIDE

AUGUST 20, 1981

Special Report: Winchester drives still going strong

Speech generation with a software development system

Ease memory management of large data bases with one IC



Design

It's no trouble to add a wide range of disks, tapes and future optical storage devices to small computer systems—thanks to an intelligent interface that allows for variations in peripheral type, speed, and performance.

Intelligent systems interface eases peripheral integration

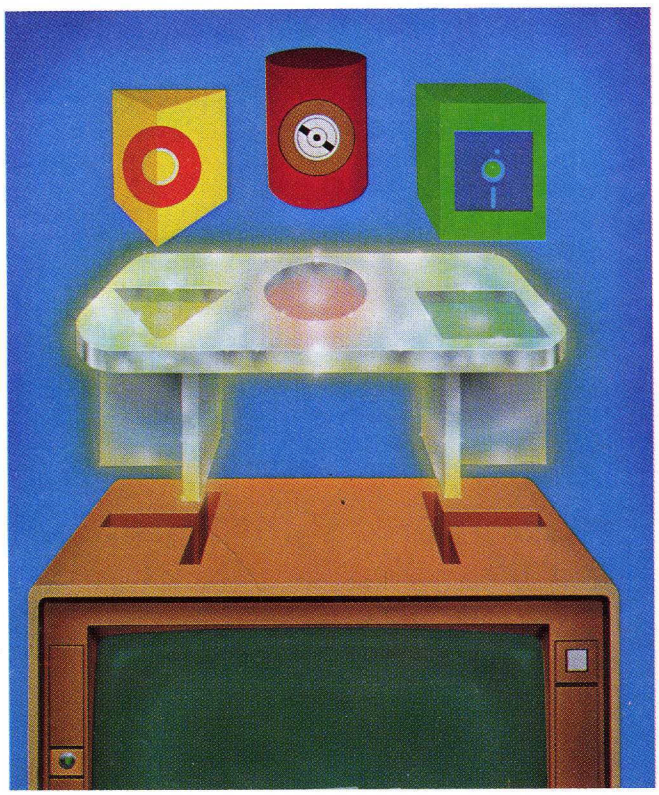
No longer is it difficult to update, mix, or interchange different types of peripheral storage devices on small computer systems. No longer are separate hardware interfaces, I/O drivers, and changes in applications software required. And no longer does the system designer or integrator have to wait for developing ANSI or IEEE-system level standards to surface. The Shugart Associates System Interface (SASI) attacks all the costly and cumbersome stumbling blocks of interfacing peripheral interfaces.

In fact, the SASI has been implemented and optimized on the SA1400 series of disk controllers. The field-proven SASI interface makes custom LSI controller chips a reality, which will, in turn, make it possible to include controller logic in the drive. Consequently, SASI, coupled with LSI chips, will debut next year in selected Shugart disk products. The SASI interface will ease the job of adding LSI chips just as it has already eased the job of adding floppy, Winchester, streaming tape drives, and future optical disks; it does this because it clearly defines the role of each device controller.

SASI is designed for systems integrators—a class of disk user that tends to concentrate its expertise on end-user applications, computer programming, and sources of hardware. It provides critical system-level solutions for error handling, protocols, and performance upgrades from floppy to Winchester technology— problems that are otherwise difficult, time-consuming, and costly to solve.

This analogy can be made: SASI is to systems integrators what high-level languages are to programmers—a design resource that moves upstream the minute details of machine code and interface protocols. What's more, SASI means faster systems integration and early market entry, thus increasing a product's life. SASI also benefits disk manufacturers across the board because with it they have a design target for interfacing controllers, one which allows access to cost-saving gate arrays, cell arrays, and fully custom LSI parts for the device controller.

Another benefit: faster market acceptance for new-product introductions of all sorts since interface protocols are already established, error-correcting firmware is already in place and OEM customers need not generate documentation.



Over the past year, SASI has been field-proven and refined in Shugart's 1400 series disk controllers. Like other standards, SASI is a unifying concept which has led to technological advances.

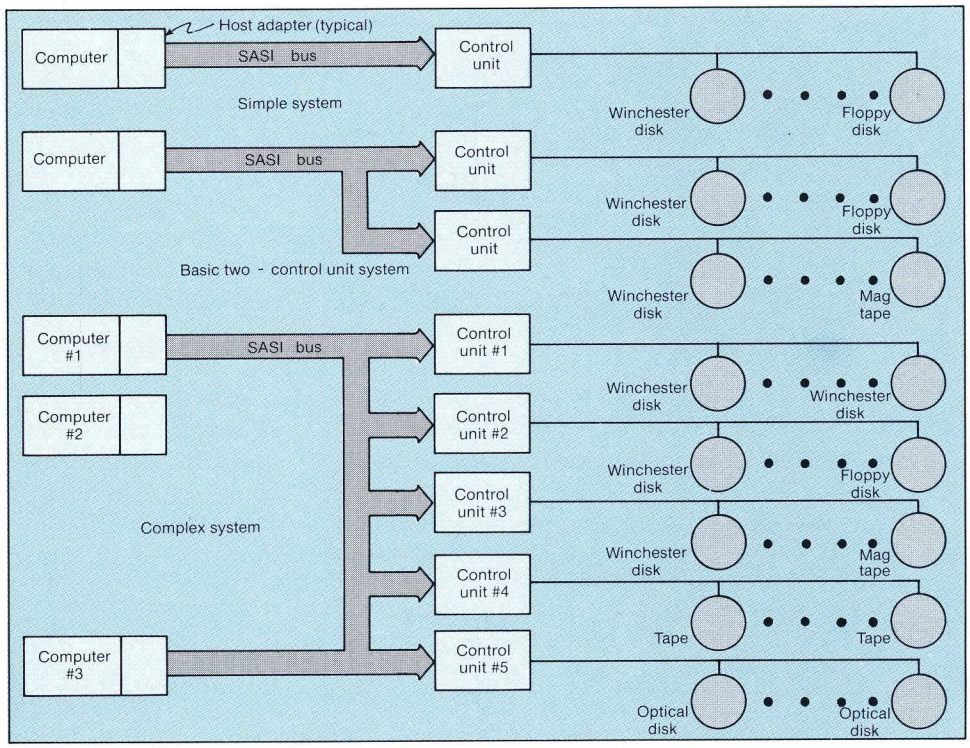
The key to SASI's intelligent interface architecture, which uses logical block addressing, is that it allows a host

Henry Meyer, Product Line Manager
James Korpi, Senior Advisory Engineer
Shugart Associates
475 Oakmead Parkway
Sunnyvale, CA 94086.

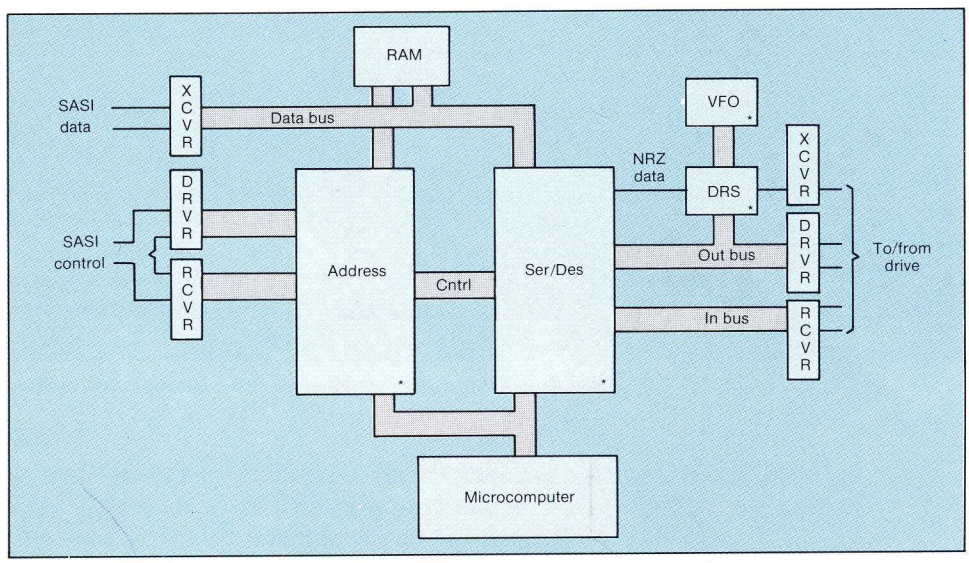
CPU to simply specify the first block address, and the number of data blocks to be transferred. The physical selection of the cylinder track and sector are all performed by the controller. No alterations to the CPU are required when upgrading or changing peripheral storage devices (Fig. 1). All housekeeping operations—including addressing, mapping, and data integrity—are transparent to the host processor.

With SASI on the SA1400 controllers, OEM systems integrators could quickly and easily upgrade their peripheral storage devices—for example, single floppy disks to Winchesters with streaming-tape backup—while remaining independent of the host processor. Whenever a system upgrade is called for, it can be done faster, with less chance of design error, miscalculation or oversight.

The SA1400 controller's high-speed bidirectional



1. Up to 8 devices can be supported by the SASI bus. The devices can be any combination of host CPUs and/or intelligent controllers.



2. Four custom LSI controller chips in the interface include the address chip, serializer/deserializer chip, data-reduction system chip, and a VFO chip.

buffer provides asynchronous data transfers between fast super-minicomputers and the slowest floppy disk, or between the slowest MOS microprocessor and the fastest laser-disk device (yet to be introduced). The host need not be concerned with matching the peripheral's data transfer speed, be it higher or lower. What's more, current implementations of SASI feature data transfer rates to 500 kbytes/s; subsequent performance levels will increase to over 1 Mbyte/s.

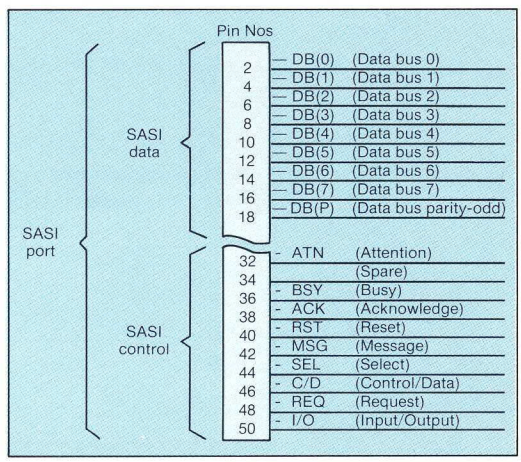
Table 1 SASI control lines

Attention (ATN):	This line is driven by the host to indicate the attention condition. This condition is used by the host to inform a controller that the host has a message ready. The controller then can get the message at its convenience.
Busy (BSY):	BSY indicates when the SASI bus is in use.
Acknowledge (ACK):	This line is driven by a controller to indicate an acknowledgement for a REQ/ACK data transfer handshake.
Reset (RST):	This line clears all bus devices from the bus.
Message (MSG):	This line is driven by the controller to indicate that data bus information for a REQ/ACK handshake is a message.
Select (SEL):	This control line is used by a host to select a controller, or alternately by a controller to reselect a host.
Control/Data (C/D):	This line is driven by a controller to indicate whether control or data information is on the data lines during a REQ/ACK handshake. This line becomes control when asserted.
Request (REQ):	This line is driven by a controller to indicate a request for a REQ/ACK data transfer handshake.
Input/output (I/O):	This line is driven by a controller to indicate the direction of data transfer on the data lines during a REQ/ACK handshake, which is an input to the host when asserted.

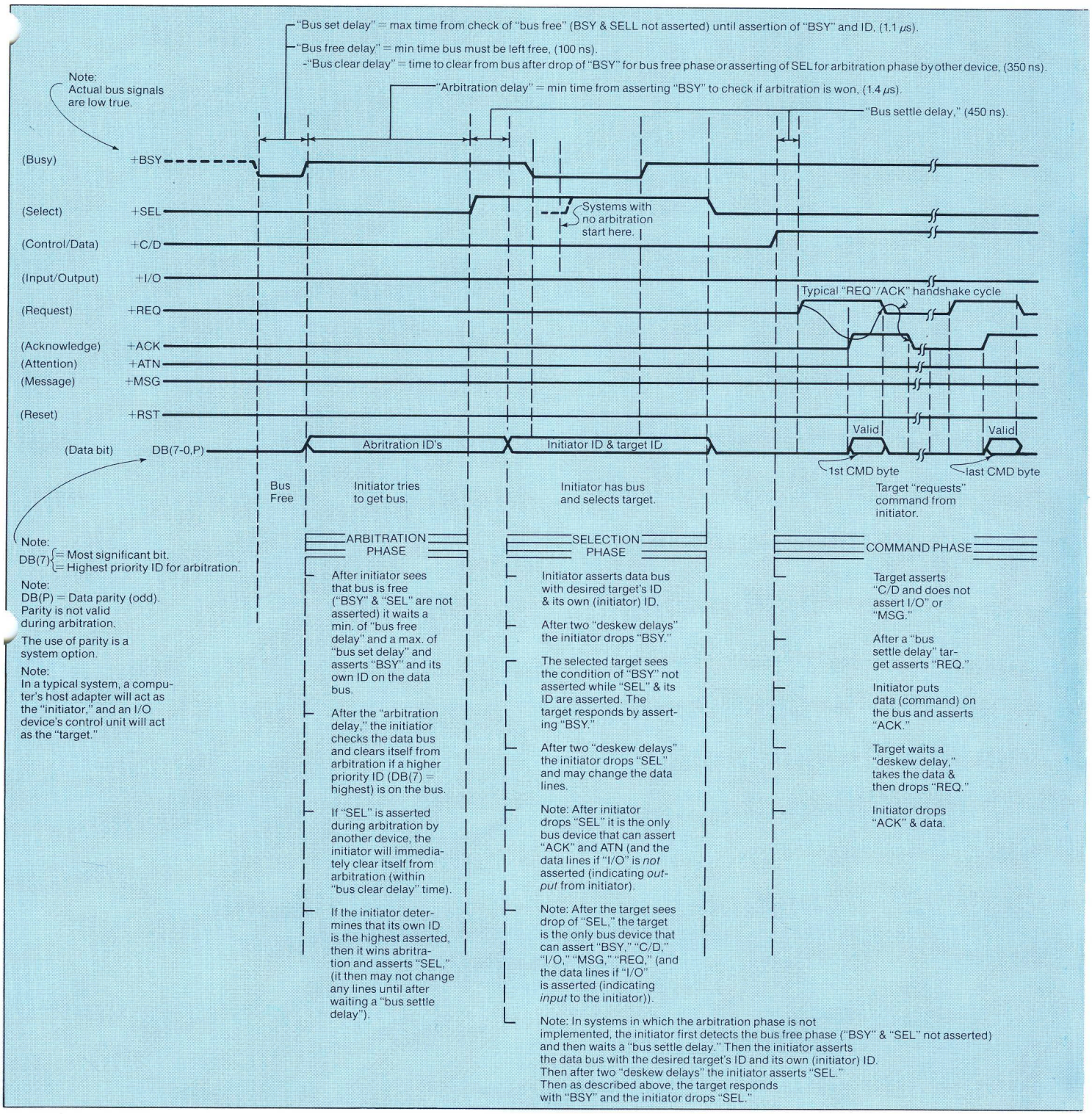
The complement of custom LSI circuits designed to operate with SASI (Fig. 2) includes a pair of MOS devices that are responsible for setting up addresses, queuing requests and other "infrequently" occurring events. A pair of bipolar devices takes care of high-speed, clock-by-clock events such as data separation, and phase-locked-loop timing generation. The custom parts—address chip, serializer/deserializer, data-reduction system chip, and VFO chip—are designed for economy, thereby slashing controller price to less than one-half of the TTL equivalent. The four chips are put together with a complement of about 30 "glue parts" plus a Z8 single-chip microcomputer. Together they assume responsibility for data flow, error detection/correction and diagnostics. In addition, they lend themselves to DMA implementations in the host. As a result, the CPU is available for more productive real-time jobs, for instance, network management, local interactive processing and distributed processing.

Multiple blocks

Automatic data chaining, automatic command chaining, and resource sharing are other major operating features of SASI. With automatic data chaining, the SASI controller assumes total responsibility for all details of fetching multiple blocks of data. Up to 64 kbytes of data can be



3. The SASI bus structure consists of nine control lines and nine data lines on a 50-pin bus. All bus lines are implemented as low-true signals.

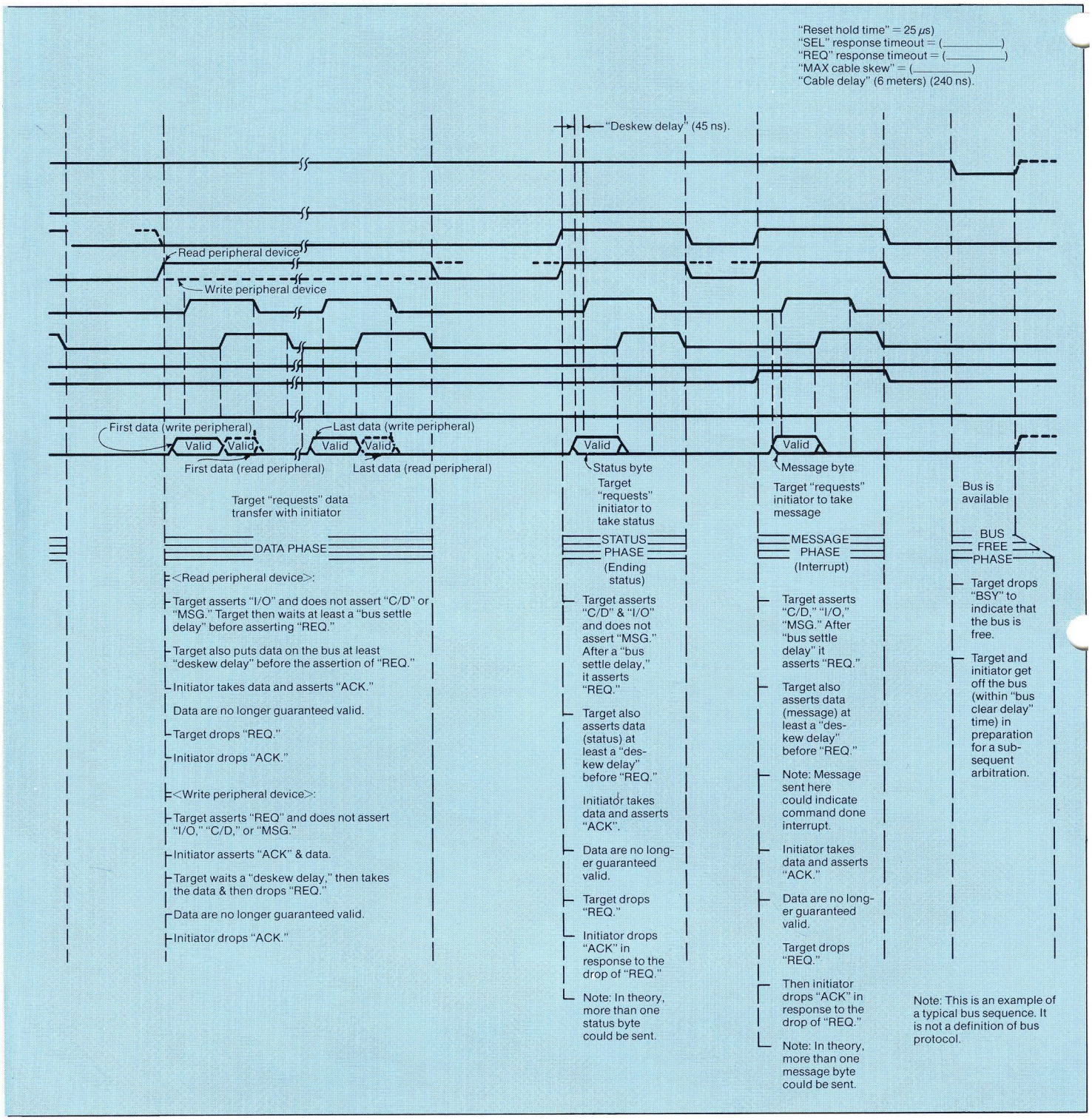


4. The SASI bus operates in any of eight phases at any moment. The bus can never be in more than one phase at a time.

transferred with a single CPU command. What's more, because of automatic data chaining, host operating systems need not be as complex as they are presently.

Automatic command chaining is initiated by the chaining bit set in the control field of the CPU's command: Each command with this bit set is linked to the next command. An unlimited number of commands can be chained in this way.

Another important SASI operating feature provides data-base resource sharing, a scheme to provide multiple host processors with access to a common



storage device. Common data bases need not be replicated, but now they can be shared through the intelligent interface. This is ideal for incorporating small systems into local area networks like Ethernet. What's more, SASI provides a self-arbitrating bus that services multiple host processors and also multiple intelligent controllers.

Keeping track of defective areas on floppy and Winchester disks is normally another task for the host operating system. SASI offloads the CPU here too. The intelligent SASI controller takes care of all disk formatting, which makes the reassignment of bad blocks to spare locations completely transparent to the host processor. SASI's automatic block sparing feature reassigns both individual blocks and full tracks to spare locations.

Automatic disconnect/reconnect is yet another innovative feature that SASI brings to system design. With it, the systems integrator can enjoy true overlapped seek on any number of drives. That is, the first drive to complete a seek will reconnect to the appropriate host for data transfer.

Bus structure

SASI's bus structure is simple and manageable (Fig. 3). It consists of nine control lines and nine data lines on a 50-pin bus. All SASI bus lines are implemented "low true." Thus, the assertion of a control function or a data line will cause the corresponding line to assume a voltage typically less than 0.5 V, while non-assertion will cause the line to assume a voltage typically greater than 2.4 V. The suggested driver/receiver types are open-collector driver 7438 and hysterisis receiver 74LS14.

A 50-connector flat cable or twisted-pair flat cable connects bus ports on host CPUs and controllers. The maximum cable length is six meters. Each bus port has a 0.1-m maximum stub length of any conductor, measured from the bus cable. A 220/330-Ω resistor pair is used for bus termination, at both ends of the bus cable.

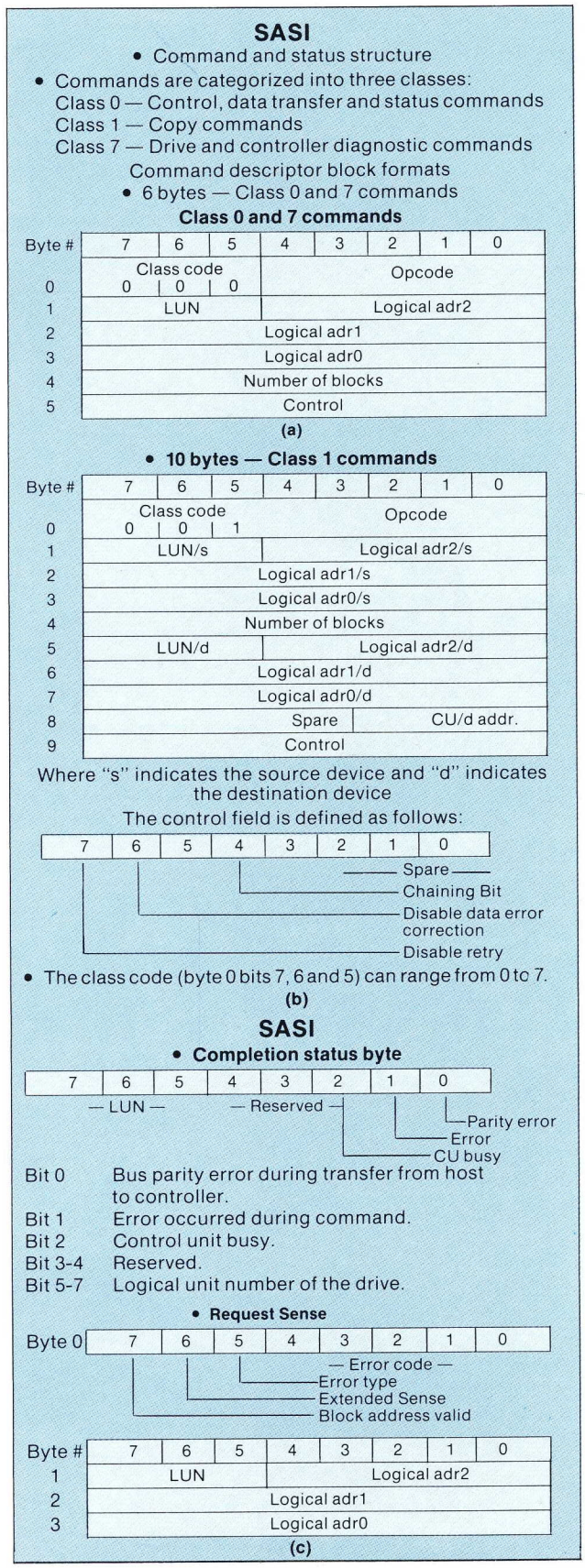
Each SASI device, whether it be a host CPU or intelligent controller, presents a SASI bus port to the 50-wire bus. Up to eight devices (or ports) can be supported on a SASI bus (Fig. 3). These ports can be any combination of host CPUs or intelligent controllers. A device's identification (ID) number is used to establish its priority when arbitrating for access to the bus.

Arbitration is the process by which these devices resolve the conflict which normally results from more than one device trying to use the bus at the same time. SASI also includes an advanced self-arbitrating bus which, although not required in simple systems, is very effective when higher performance is required. With this feature, the system operates out of each SASI port, and does not require the use of a master arbiter.

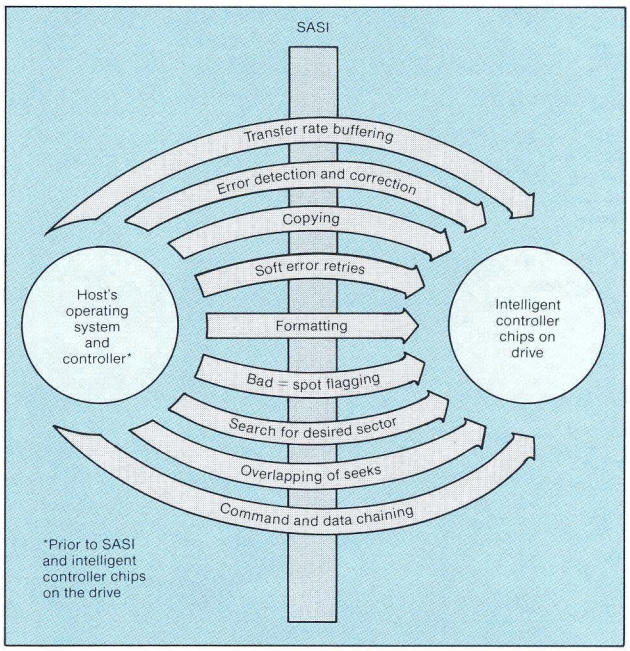
Bus contention

Each device that wants the bus waits for the bus to become free. The device then arbitrates for the bus with every other device in contention. All SASI ports have a logical identification number which is asserted via the corresponding numbered data line. The device with the highest ID number—in this case, number 7 is the highest number and 0 is the lowest —wins the arbitration. To arbitrate again, the other devices have to wait until the bus is again free.

Once a device wins arbitration and connects to the bus, it signals through SASI and connects with another device. The bus is then dedicated to that



5. The structure of commands and status reporting are Class 0 for control, data transfer and status (a), Class 1 for copy (b), and Class 7 for drive and controller (c).



6. Once SASI is implemented, drive and controller functions are handled at a standard system-level interface, instead of the device level interface.

Table 2 Phases of SASI operation

Arbitration phase:	Allows host CPU or controller to gain control of the bus. Implementation of the arbitration phase is a system option. Systems with no arbitration phase can have only one host. The arbitration phase is required for implementations that use the reselection phase.
Selection phase:	Allows a host CPU to select a controller for the purpose of initiating some drive function, like read or write data. However, the selection phase differs from the reselection phase in that during selection, the I/O line is not asserted.
Reselection phase:	Allows a controller to reconnect to a host for the purpose of continuing some operation that was previously started by the host but was interrupted by the controller. (Note: Reselection can only be used in systems which have arbitration implemented.)
Command phase:	Allows the controller to request command information from the host.
Status phase:	Allows the controller to send status information to the host.
Data phase:	Used to send data between the host and the controller.
Message phase:	Used to send messages between the host and the controller.
Bus free phase:	Indicates that bus is not being used. (Select and Busy are not asserted.)

connection until the action between the two devices is completed. When finished, the controller releases the bus. Then, the host CPU must immediately get off the bus. Devices that are waiting to use the bus can then begin arbitration.

The 18 active SASI bus lines are briefly described in Table 1. The simplicity and effectiveness of the interface is demonstrated in the data bus, which consists of eight data lines plus a parity line. Data bus bit 7 is the most significant bit and is also the highest-priority bit during arbitration. Bit number significance and priority decrease as the bit numbers decrease. If data bus parity is used (a system option), parity is odd. Parity is not valid during arbitration.

The C/D, I/O, and MSG lines are used to define eight different types of information to be transferred on the data lines during a REQ/ACK handshake. The types of information include: commands from the host, status to host, data-in to host, data-out from host, message-in to host, message-out from host, while two types are reserved for future use.

The SASI bus operates in one of eight phases at any point in time (Table 2). The bus can never be in more than one phase at any given time. A typical sequence of events on the bus is shown in Fig. 4. The structure of SASI's commands and status reporting is as shown in Figs. 5a through 5c. Other classes of commands may be added.

Future implications

A major transition from standardization efforts directed to the device level (device interface) to a standard at the system level (system interface) means that the system interface standard has been made "device independent." In other words, SASI will do for the microcomputer-based small-system market what the IBM I/O channel standard did for large mainframe computers. Numerous plug-to-plug devices can now be connected to IBM's I/O channel because of the channel standard. Analogously, numerous new peripherals will be connectable to small systems via the SASI standard and standard adapters to connect SASI to the memory buses of host systems. Users can begin with SA1400-type controllers now and progress to Shugart's board implementation of custom LSI chips beginning next year. For example, Fig. 6 shows pre-SASI functions that have been moved out of the realm of host responsibility because of the standard system-level interface.□