HDD Interface Technologies

● Masakazu Kawamoto

(Manuscript received July 1, 2005)

The interfaces of hard disk drives (HDDs) have been changed to serial interfaces to meet the demands for high-speed data transfer. The main interfaces involved are the Fibre Channel (FC), Serial ATA (SATA), and Serial Attached SCSI (SAS). Up to now, the HDD market segments corresponded to the serial interfaces that were available. Because of the recent demands for cost reduction, however, requests are being made for serial interfaces optimized for the use of the HDD units. As a result, the HDD market segments have become fractionalized, and the correspondence between the market segments and serial interfaces has also been lost. For example, products in which a conventional low-end SATA and a large-capacity HDD are combined have begun to appear on the market as enterprise HDDs. This paper describes the features of these serial interfaces and the interface technologies of the HDD controller that enables serialization of the interfaces. It also discusses the technological problems of future HDD interfaces and their solutions.

1. Introduction

All of the interfaces used in computer systems have been changed from parallel transmission to serial transmission. The main cause of this change has been the demand for faster interfaces. This applies even to the interfaces of hard disk drives (HDDs). In the latter half of the 1990s, a Fibre Channel (FC)[1)] having a transfer rate of 1 Gb/s (100 MB/s) was put into practical use as the serial interface of HDDs. Then, a Serial ATA (SATA)[2)] having a transfer rate of 1.5 Gb/s was produced in 2003, and a Serial Attached SCSI (SAS)[3)] having a transfer rate of 3 Gb/s was produced in 2004. Since it produced its first FC HDD in 1998, Fujitsu has been developing and providing SATA and SAS HDDs to meet the demands of its customers.

This paper describes the features of these serial interfaces and the interface technologies of the HDD controller that enables serialization of the interfaces. It then discusses some of the future problems in this field. Note that in this paper, we use the terms "interface" and "HDD body" to distinguish between the interface of an HDD and the HDD's platters, heads, motor, control circuit, and remaining components.

2. Interfaces for HDDs

HDDs are used in many different places for many different purposes. Different interfaces have therefore been developed to meet this diversity. The AT Attachment (ATA) interface[4)] was developed for the internal HDDs of desktop PCs, while Small Computer System Interface (SCSI)[5)] was developed for servers and large-scale storage systems. Now, these parallel interfaces are being replaced to meet the demands for higher speeds. SCSI is being replaced with FC and SAS, while ATA is being replaced with SATA. Moreover, because the uses of HDDs are also expanding, interfaces to meet these new uses are also being developed. **Table 1** lists the main HDD interfac-

Table 1
HDD interfaces.

Interface name˜	ATA/IDE˜	SATA˜	SATA-2˜	SCSI˜	SAS˜	FC-AL˜
Command˜	ATA˜	ATA˜	ATA/ATAPI˜	SCSI˜	SCSI˜	SCSI˜
Specification/date˜	ATA-7, 2001/11˜	Rev.1, 2001/8 ˜	Rev.1, 2002/8˜	SPI-5, 2002/5˜	Rev.5, 2003/7˜	Rev.7, 1999/4˜
Volume production date˜	Now˜	2002˜	2005˜	Now ˜	2004˜	Now˜
Transfer speed˜	66, 100, 133 MB/s˜	1.5 Gb/s˜	1.5, 3 Gb/s˜	160, 320 MB/s˜	3, 6 Gb/s˜	1, 2, 4, (8) GB/s˜
Cable/link displacement˜	0.5 m˜	1 m˜	7 m˜	25/12 m˜	0.5 m/10 m˜	30 m/10 km˜
HOT pluggable˜	Not available˜	Available˜	Available˜	Not available˜	Available˜	Available˜
Connector˜	40 pin˜	7+15 pin˜	7+15 pin˜	80 pin˜	7+15 pin˜	40 pin˜
Number of drives˜	2˜	1˜	1 ˜	16˜	16256 ˜	126˜
Topology˜	String˜	Point-to-point˜	Point-to-point˜	String˜	Star (Expander)˜	Loop,Star (Switch)˜
TRX power consumption˜	Lowest˜	300 mW˜	300 mW˜	High˜	300 mW˜	300 mW˜
Speed expandability˜	Limited˜	Available˜	Available˜	Limited˜	Available˜	Available˜
Multi-initiator˜	Not available˜	Not available˜	Not available˜	Available˜	Available˜	Available˜
Queuing limit˜	32˜	32˜	32˜	No limitation˜	No limitation˜	No limitation˜
Dual-port configuration˜	Not available˜	Not available˜	Not available˜	Available˜	Available˜	Available˜
Arbitration˜	No˜	No˜	No˜	Yes˜	Yes˜	Yes˜
Impact on other˜ devices at port failure˜	Yes˜	No˜	No˜	Yes˜	No˜	Impact (loop)˜ No (S/W)˜
Interface's˜ characteristics ˜ and application	With In BOX˜ PC/consumer	With In BOX˜ Replace ATA˜ PC/consumer	With In BOX˜ PC/tier storage	In/Out BOX˜ SVR/RAID	In/Out BOX˜ Replace SCSI˜ SVR/RAID	In/Out BOX˜ JBOD/SBOD˜ High-end RAID

es. The following sections describe the features of the four types of serial interfaces, including the CE-ATA[6] currently being developed.

2.1 FC

The FC interface combines the characteristics of a data channel, for example, high-speed transfer, and the features of a network, for example, a wide range of connections. The link transfer rate of the FC interface was originally 1 Gb/s (100 MB/s), increased to 2 Gb/s, and is currently 4 Gb/s. A transfer rate of 8 Gb/s is currently being investigated. This link supports simultaneous bidirectional communications with independent sending and receiving operations. **Figure 1** shows the topologies (connection forms) of FC units.

The basic topologies are point-to-point, star centered around Fabric, and loop, and these three can be combined to construct other topologies. Because the mesh form has multiple routes, load balance and failure avoidance are possible.

Generally, the protocols of serial interfaces have a hierarchical structure (**Table 2**).

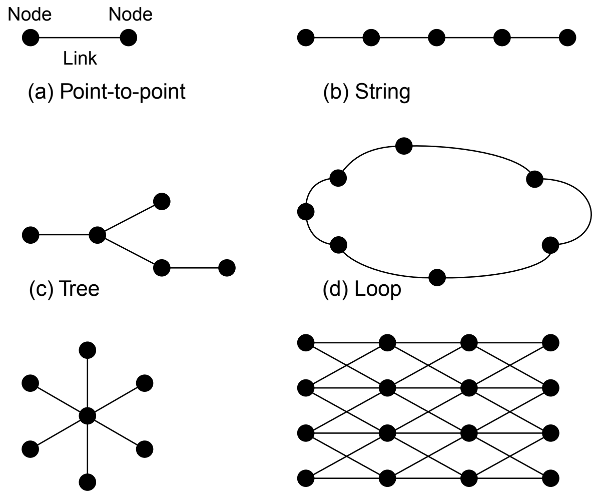
The user protocol layer is positioned at the highest layer of the FC. The FC can support multiple types of protocols such as the SCSI command protocol and Internet protocols (IPs) using a single FC network. SCSI is used for the HDD connections. FC is therefore multifunctional. However, this versatility makes the FC interface complicated. To reduce this complexity, only the FC functions required for the HDD connection are selected and used when an FC interface is used for HDD connections. These selected function specifications are collectively referred to as a profile. PLDA[7] and FLA[8] are representative of such profiles.

The FC interface is used to connect multiple HDDs in high-performance storage systems. Because high reliability is also demanded from these high-performance systems, HDD bodies with an

FC interface are designed with emphasis on high performance and high reliability.

2.2 SAS

SAS is an exclusive HDD interface that combines a simple SATA link and a layer control protocol similar to that of FC (Table 2). The SAS link has a bidirectional speed of 3 Gb/s. There are two types of topologies: point-to-point and star centered on an Expander. Because the SAS link shares part of the link layer Primitive with SATA, SAS can share the link and Expander with SATA devices. The SCSI command set is used for disk control. Because the functions have been restricted from the start for use as an exclusive HDD interface, SAS is a simpler interface than FC. HDD bodies combined with the SAS interface are identical to those of conventional SCSI HDDs.



(a) Point-to-point (b) String

(c) Tree (d) Loop

(e) Star (f) Mesh (network)

Figure 1
Topology.

2.3 SATA

SATA was developed with the single aim of serializing the parallel ATA interface. The SATA is a very simple interface; it supports only point-to-point connections and has limited functions, for example, it has no function for specifying drive addresses. Because the 1.5 Gb/s SATA link handles Primitive interlock control (described later), frames cannot be sent and received simultaneously. The ATA command set is used for disk control.

The SATA interface was therefore developed and produced to replace the ATA interface of ATA

Table 2
Protocol layer structure.

Interface / Layer	SCSI˜	FC˜	SAS˜	ATA˜	SATA˜
Command˜	SCSI-3˜	SCSI-3˜	SCSI application ˜ layer˜ SCSI-3˜	ATA˜	Device command ˜ layer˜ ATA˜
Mapping˜	–˜	FC-4˜ SCSI-FCP˜	Transport layer ˜ Frames˜ COMMAND/TASK/˜ XFER_RDY/DATA/˜ RESPONSE˜	–˜	˜–˜
Protocol˜	Bus phase˜ Bus sequence˜ Bus condition˜	FC-2˜ Primitives / Frames / ˜ Service /˜ Flow control˜	Port layer˜ Frames ˜	Interface operation˜ Register definition˜ Protocol (PIA/DMA)˜	Frame (FIS) ˜
Link˜	Signal line˜ Bus timing˜	FC-1˜ 8B/10B code ˜	Phys & link layer˜ 8B/10B code / OOB˜ Primitives / CRC /˜ Address frame˜	Signal line˜ Bus timing˜	8B/10B code /˜ Scramble ˜ Primitives / CRC˜
Physical	Connector˜ Cable˜ Electrical˜ characteristic	FC-0˜ Connector˜ Cable˜ Electrical ˜ characteristic	Connector˜ Cable˜ Electrical ˜ characteristic	Connector˜ Cable˜ Electrical ˜ characteristic	Connector˜ Cable˜ Electrical characteristic˜ OOB signaling

HDDs that were developed as internal HDDs for use in desktop and notebook PCs. Recently, however, the SATA interface has also begun to be used for enterprise HDDs as described below.

2.4 CE-ATA

CE-ATA supports point-to-point connections using an MMC link[9] and sends serial data and clock signals using separate signal lines. By restricting the length of its cables to several centimeters, the effect of skew on the data and clock signals can be suppressed. Because the data is sent separately from the clock signals, there is no need for a PLL or other circuits to regenerate the clock signals at the receiving side. As a result, the circuits are much simpler. Part of the ATA command set is used for disk control.

The above described HDD bodies and their interfaces and how they are being used. However, recently, it has become necessary to review these relationships.

Conventionally, HDD bodies designed for high performance and reliability have been combined with an FC interface for use in storage systems that require high performance and reliability.

Customers are increasing their demands for higher storage capacities, reduced bit costs, improved space efficiency, and reduced power consumption for the most recent storage systems. To meet these demands, a method of choosing HDD bodies suited to the characteristics of the target data has been started. For example, mission-critical data is placed on a high-performance FC HDD to achieve stable, high-reliability operation for fast FC HDD processing and the high loads of continuous 24-hour operation. Bulk data that is not frequently accessed is placed on a large-capacity SATA HDD that operates at a slightly slower speed. By choosing HDD bodies based on the data characteristics, an entire storage system can be optimized. Storage systems with HDD bodies having different performances are called tier type systems.

In the case described above, the SATA and FC interfaces are not compatible, so an interface converter is needed to connect a large-capacity SATA HDD to an FC network in a storage system. To eliminate the need for a converter, an HDD that was originally designed as a SATA HDD has been modified to replace SATA interface to FC interface for FC tier storage systems. In addition, instead of switching the interfaces, FC-SATA specifications in which SATA frames are embedded in the FC frames are being investigated.

In this way, conventional desktop SATA HDD bodies are also beginning to be used for enterprise storage systems. The conventional SATA is therefore no longer classified as simply for desktop use. From now on, the HDD interface and usage will have to be considered from the viewpoints of the characteristics of the HDD interface and the characteristics of the HDD body. With these differences in mind, the following sections describe the optimum interfaces based on the new uses and market demands and the points and problems that must be considered to realize these interfaces.

3. Market segments and interfaces

HDD bodies are classified into market segments based on their uses. These classifications are meaningful in that they indicate the characteristics and the conditions for preparing an HDD. The HDD market can be divided, for example, into the six market segments listed below. The serial interfaces of the main HDD of each market segment are shown in parentheses.

– Enterprise (FC)
– Server (SAS)
– Desktop (SATA)
– Mobile (SATA)
– Consumer (stationary) (SATA)
– Consumer (mobile) (CE-ATA)

The current combination of conventional interfaces and HDD bodies must be reviewed whenever new uses give rise to new market segments and also when new uses arise for the current market segments. A good example of this is the large-capacity SATA HDD used in the enterprise storage system given in the example above. Actually, the low-speed, large-capacity SATA HDD in the example is not always the same as that used for desktop PCs. This is because, even for a SATA interface, for example, a large-capacity HDD combined with SATA must provide the high reliability demanded by an enterprise system.

To connect multiple host systems to multiple storage systems, an interface that provides network functions, for example, the FC interface, must be used. This type of network is referred to as a storage area network (SAN).

If the internal structure of a storage system is a Just a Bunch Of Disks/Switched Bunch Of Disks (JBOD/SBOD) instead of a Redundant Array of Inexpensive Disks (RAID), the host directly accesses the HDD bodies in the storage system. This means that an FC interface equivalent to the external interface of the storage system is suitable for use as the internal HDD interface.

For a RAID system, however, the user data is first expanded on the buffer of the RAID controller then distributed for a write or collected for a read. Data access is then handled independently between the host and HDD interfaces, and the HDD is not accessed directly from the host. The HDD interface operations have no relationship with the host interface operations. Therefore, the HDD interface only has to be suitable for controlling the closed network consisting of the RAID controller and subordinate HDD group.

That is, the optimum HDD interface depends on the storage system architecture. As such, it is predicted that SAS and SATA interfaces will be used as the internal interfaces for small to medium sized RAID systems in the future.

For example, for small-scale RAID systems for Small to Medium sized Businesses (SMBs), customers demand low-device-installation costs and the ability to improve system performance when their system is expanded in the future. To meet these demands, desktop HDDs and enterprise HDDs must be used together. Also, because the system scale is not that large, SAS is used for the interface within the subsystem, and an SATA HDD that can operate with SAS is used as the drive at installation.

However, for large-scale RAID systems, including tier-type systems, FC is used for the internal interface because of the performance demands and the many HDDs that are needed to meet the required capacity. For these types of system, large-capacity SATA drives are accommodated in an exclusive cabinet and connected to the internal FC network through an interface converter.

For internal drives such as those used for servers and PCs, only one host is connected and complicated addressing functions are not needed. From the viewpoint that the drive conventionally used is a SCSI HDD and the existing software resources are to be used, SAS HDDs are used if SCSI commands are requested, and SATA HDDs are used if ATA HDDs have been used.

4. Serial interface features

This section describes the features of serial interfaces. In particular, it discusses the points and problems that must be considered when serial interfaces are used for HDDs.

4.1 Cables, signals, and connection forms

For serial interfaces, the signal transmission lines (links) are formed using cables and connectors.

For serial interfaces, the connection forms of the links and devices (nodes) are referred to as the topology (Figure 1). Table 1 lists the type of topology used for each serial interface.

For a serial link, the transmitters and receivers are connected point-to-point using cables.

1) For a point-to-point connection, the transmitter of the local node is connected to the receiver of the remote node, and the transmitter of the remote node is connected to the

receiver of the local node. This type of connection form, therefore, enables bidirectional sending and receiving of signals.

2) For a loop connection, three or more nodes are connected to form a loop. The receiver of a node is connected to the transmitter of the next upstream node in the loop, and the transmitter of the node is connected to the receiver of the next downstream node in the loop. That is, the transmitter and receiver of a node are each connected point-to-point to a different node. Each node has logic circuits that send the data received by the receiver to the transmitter so data is relayed in turn. This group of transmitter and receiver connected by the logic circuits within the node is referred to as a port. Each node can have multiple ports.

3) For a star connection, the node that forms the center of the star has multiple ports that are connected to the ports of the other nodes. The ports of the central node are connected to switching circuits within the central node and exchange frames with the other ports. Route switching or frame switching circuits are used for the switching circuits. Route switching circuits maintain the connection routes between the ports, while frame-switching circuits exchange frames between arbitrary ports as the occasion demands.

4) String, tree, and mesh connections can be thought of as combinations of star connections.

The conventions for the signal levels and coding schemes on the link are referred to as the link layer. For serial interfaces, all information such as the clock signals, data, and control signals is transmitted using bit strings sent and received using a single signal line. This is enabled by assigning meanings to special bit patterns or arrays in regards to control. The FC, SAS, and SATA interfaces all use 8B/10B transmission codes.[10)]

4.2 Limitation of data transfer rates

Parallel interfaces such as SCSI and ATA transfer multiple data items and the clock and data strobe signals in parallel. For parallel transfer, the speed of the parallel interface is limited due to the influence of crosstalk and skewing between the signals. Development has stopped at 100 MB/s for the ATA interface and at 320 MB/s for the SCSI interface. Because, in principle, skewing does not occur in serial transmission, the serial transfer rate can be increased by raising the clock speed.

4.3 Technologies for increasing the serial transfer rate

The clock speed of serial interfaces is limited by the operating frequencies of the transceiver and its peripheral circuits. It is also limited by the link characteristics (particularly the frequency-dependent attenuation characteristics) and the effectiveness of the compensation technology. The operating frequency of the transceiver depends on the LSI technology. For CMOS for example, the operating frequency depends on the size of the gate channels, which is based on the design rule. For an HDD, the improved interface technologies described above and the advanced data read/write frequencies of media have increased the data transfer rates of interfaces. This increase is illustrated in **Figure 2**, which shows the roadmap of HDD interfaces.

4.4 Signal quality of serial interfaces

Twisted copper wires suitable for propagating differential signals are used for the serial signal lines of HDDs. In addition, when using HDD connectors that are directly attached to the printed circuit board, the transmission lines are formed as microstriplines on the board.

For all serial interfaces, the stipulated Bit Error Rate (BER) of the transmission line should be lower than 1 bit per 10^{12} bits. For actual products, however, much lower BERs (1 bit per 10^{14} to 10^{15} bits) are demanded.

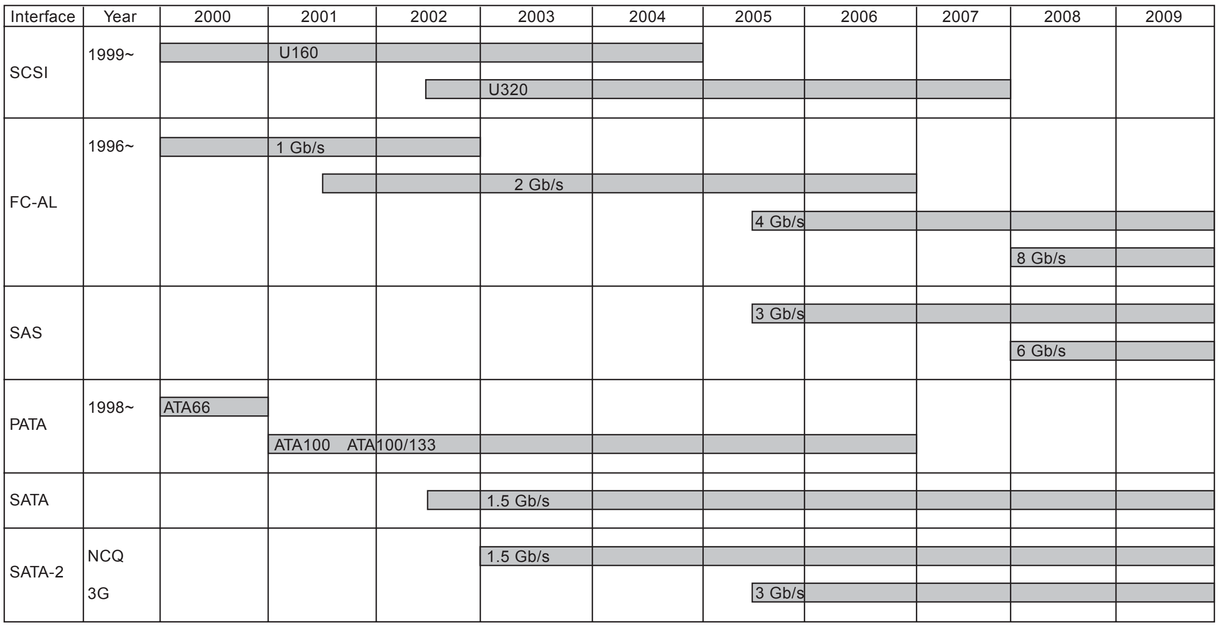


Figure 2
Road map of HDD interfaces.

The BER is closely related to the jitter of the interface signals,[11] which is a fluctuation in the timing of signal transitions. There are various causes of jitter, and it is usually caused by a combination of them. Some typical causes of jitter are the thermal noise of the line, reflection of the transmission line, and electrical noise from outside the circuit. Some causes of jitter cannot be avoided, while some can be avoided to some degree by skillfully adjusting the characteristics and design of the parts.

In addition, receivers can tolerate a certain amount of jitter without introducing errors, and this tolerance is one of the factors that determine the overall capability of a transmission system.

4.5 Interface compatibility

Generally, to interconnect devices via an interface, designers must consider not only the specifications of the interface standard but also various undocumented details of the interface. Otherwise, conflicts in processing will occur that can have a severe impact on the linked operation of the devices. The presence of conflicts is checked when the devices are connected. If there are no conflicts in processing between devices, the devices are said to be interoperable. For serial interfaces, organizations such as the Interoperability Laboratory (IOL) of the University of New Hampshire (UNH) have implemented compliance tests. Connection operations are also checked at events referred to as plugfests, where manufacturers bring their devices and they are tested for compliances. Fujitsu has participated from early on in these organizations and events to ensure the high interoperability of its products.

4.6 Command sets

Serial interfaces have the hierarchical structure shown in Table 2. The highest layer is the command layer, which is where the SCSI and ATA command sets are used. At the command layer, serial and parallel interfaces use the same command sets. FC and SAS differ from the physical layer to the mapping layer. However, they both use the same SCSI commands,[12] so the command sets of FC and SAS can be considered to be the same logical interface.

Because of their original purposes, there are differences between the SCSI command set developed as a system interface and the ATA command set[4] developed as a PC internal drive interface. For example, the byte lengths of SCSI sectors are variable, but the byte length of ATA sectors is fixed to 512 bytes. The differences in the functions of these command sets have become differences in the functions of the respective HDDs.

When FC, SAS, and SATA HDDs are used together, it becomes difficult to handle the functional differences of SCSI and ATA. Until now, this problem has been handled separately. Recently, however, INCITS T10/T13 has been working on standardizing the conversion specifications.[13]

4.7 Data integrity

Data integrity in a storage system means that user data is copied from the host memory to the HDD media and vice versa without error. This section discusses how the integrity of data in the interface and HDD body of serial interface HDDs has been improved.

Serial interfaces perform 8B/10B encoding to encode 8 bits of data into a 10-bit code. If a data bit in one of these codes is changed due to noise or another cause, there is a high probability that the code will become invalid. To check for errors, therefore, the validity of each code is checked at the receiving end. In addition to the code validity check, a CRC code attached to each frame is used to check the frame's validity. Also, a sequence number is added to the header of each frame. These sequence numbers are then checked to detect frame losses.

The movement of user data within the hard disk controller (HDC) body is checked using CRCs and parity bits without gaps. In addition, each data item on the media is given an Error Correction Code (ECC) that includes the Logical Block Address (LBA) of the data, and this information is used to check whether the received data came from the target sector.

All of these checks are handled by hardware.

The number of HDDs that can be connected to a single RAID controller is increasing and depends on the connectivity of the serial interface. For a RAID system, the user data is distributed for HDDs and collected for reconfiguration. To detect incorrect operations that may occur during this distribution and collection, sequence numbers are assigned to the data. These sequence numbers are checked at critical locations in the storage system to also identify the locations of incorrect operations. This method is referred to as End-to-End data protection. Standardization of the End-to-End data protection[14] specifications is moving forward, and Fujitsu is participating in this standardization.

4.8 Security functions

As one of the characteristics of a network, serial interfaces can be used to increase the configuration scalability, and as a result, enable the construction of large-scale storage networks. However, the devices in such a network are prone to illegal accesses. To prevent illegal access therefore, a method has been developed by which 1) groups of devices are logically assigned to zones based on addresses and 2) access is restricted to access within the zones. For FC, Fabric positioned in the center of a star topology manages the addresses of all devices connected to Fabric. As a result, Fabric also performs exclusive control using zones. For SAS, addresses are assigned to the physical port locations of the Expander, and restriction of access based on these addresses is under review.

As described above, each type of interface has specific access restrictions. One method that is independent of the type of interface is Trusted Peripheral (TPer) of the Trusted Computing Group (TCG).[15] This method restricts access to an entire system based on the identification of devices using public keys. By identifying devices using an encrypted ID specific to each device, connection and access to devices can be controlled. Fujitsu is also participating in this standardization.

5. HDD controller

This section discusses the interface technologies of the HDD controller for handling the features and problems of the serial interfaces that have been described so far.

First, we briefly describe the HDD controller. We then describe the host interface controller by comparing the structures of SATA, which provides simple functions, and FC, which provides multiple functions.

The HDD controller is divided into two sections. One section controls the host interface that decodes the commands sent from the host, sends the requested data, and reports the status. The other section controls the drives, for example, it determines the head position, controls media rotation, and controls the data format on the media.

1) Controller structure

Figure 3 shows the block diagram of the FC HDD controller.

A data buffer of several megabytes to several 10s of megabytes, control memory, microprocessor for control, and connectors and ports configured with impedance matching circuits are positioned around the HDC, which is the core of the controller. Read channels for reading/writing the data from the disk media, a servo controller for determining the positioning of the actuator and controlling the spindle rotation, and disk mechanism are positioned around the disk ports. Some of these components are omitted in the figure.

Within the HDC, there are two FC host

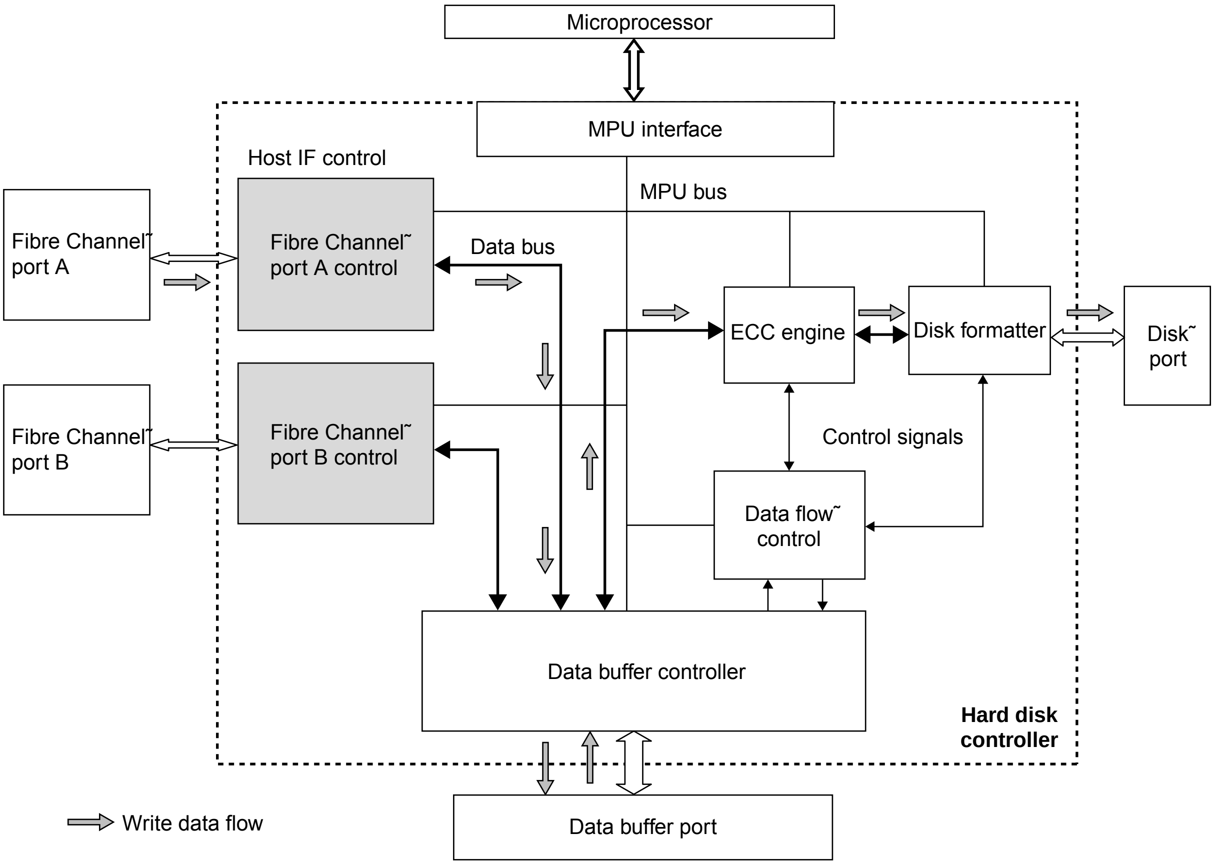


Figure 3
FC HDD controller.

interface controllers; a data buffer controller (BC); an ECC engine that generates ECC, detects ECC errors, and generates error correction data; and a disk formatter (DF) that controls the formatting of data on the media. The figure shows the flow of user data when writing.

The user data sent from the host to the FC interface is written to the data buffer via the FC interface control circuit. When the buffer has accumulated a specific amount of data, the data is read and an ECC is generated and attached to the data. The data is then sent to the disk port after being adjusted to the format of the media by the DF, where it is finally written to the media. Reading is performed in almost the reverse order of writing.

An example of an FC HDC has been given here. For the other interfaces, the basic structure of the controller is the same. The reason for the two host interface controllers is because of the dual port. Except for the FC and SAS dual port, there is only one host interface controller for the other interfaces because there is only one port.

2) Host interface controller

Figure 4 shows the block diagram of the FC host interface controller. The features of the FC include completely independent send/receive and control circuits for constructing a loop topology between both.

The flow of data at the receiving side is described first. A clock signal is generated by the PLL from the serial signal received by the receiver (RX), and a data string is recreated based on the clock signal. The data string is divided into 10-bit characters based on patterns in the data string, 10B/8B decoding is performed, and an 8-bit data code and a control code are detected. The PLL/Decoder performs these character operations. Next, the Primitive signal consisting of one control code and three data code strings (four

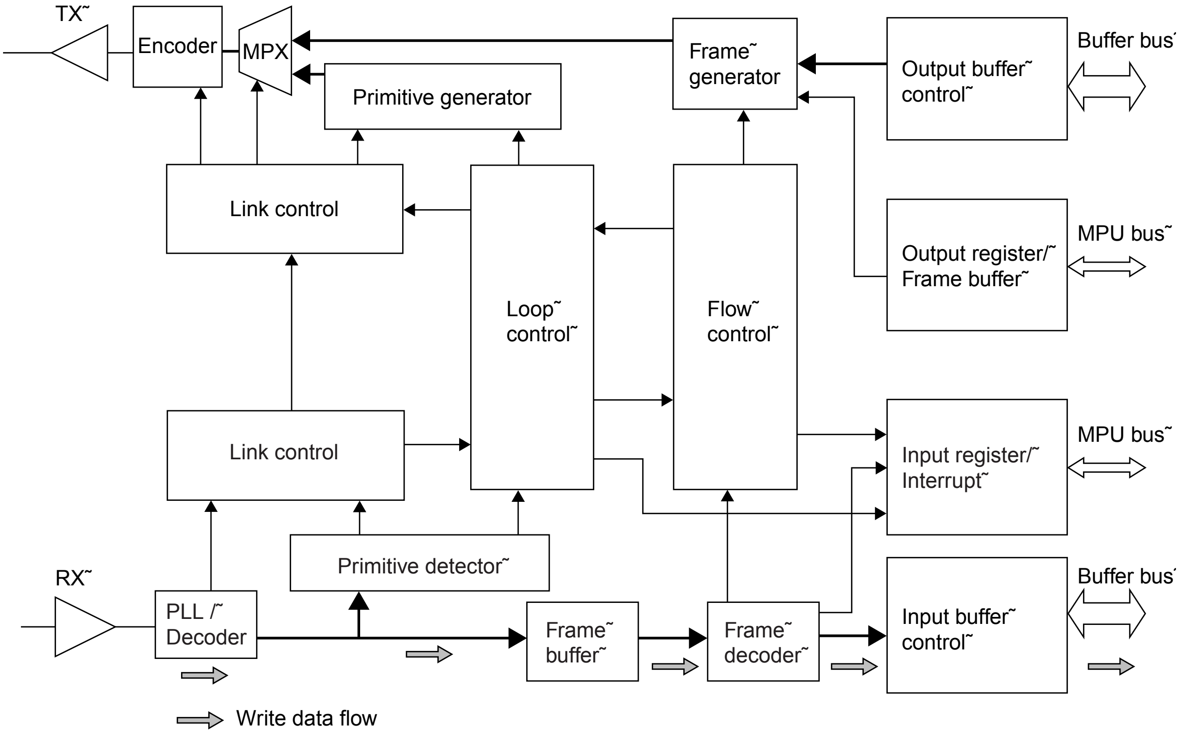


Figure 4
FC host interface control.

bytes, one word) is detected. This Primitive signal indicates the beginning and end of the data frame, and this information is used to detect whether the data frame can be received. If the frame can be received, it is then placed in the frame buffer, where it is validated and classified by the frame decoder. The user data frame is written to the buffer via the buffer control circuit.

The flow of data at the sending side is described next. The data read from the buffer by the buffer control is combined with the control data for the frame header prepared by the output register, arranged into a data frame by the frame generator, and sent to the encoder. After the encoder performs 8B/10B encoding on the data frame, the data is arrayed serially and sent by the transmitter.

The flow control maintains the frame-receive enable status from the host and handles control so the allowable number of frames are sent.

The link control monitors the Primitive signal and signal states of the receiving side and controls the link status. The link control of the sending side inserts a Primitive signal for control in the interframe gap.

For a comparison with FC, **Figure 5** shows the block diagram of the SATA host interface controller.

The flow of data at the SATA receiving and sending sides is nearly identical to that of FC. However, there is only one control section because SATA reception and sending are linked. For example, while a frame (FIS) is being received, the sending side continues to send a “receive in progress” (R_IP) Primitive to indicate that a frame is being received. In this way, sending and receiving are linked (using Primitive interlock) and executed as a single processing operation. As a result, there is only one type of control status transition for sending and receiving frames and Primitives, and therefore only one control circuit. In addition, during frame transfer, whether data is being read to or written from the buffer, the action is performed as a single operation. Therefore, there is only one buffer control circuit. Unlike FC, SATA does not have loop control.

6. FC interface problems

This section describes some of the technological problems of the FC interface used for HDD controllers and the solutions to these problems.

A loop topology called the Fibre Channel Arbitrated Loop (FC-AL)[1] is used for FC so that a large number of HDDs can be connected using a

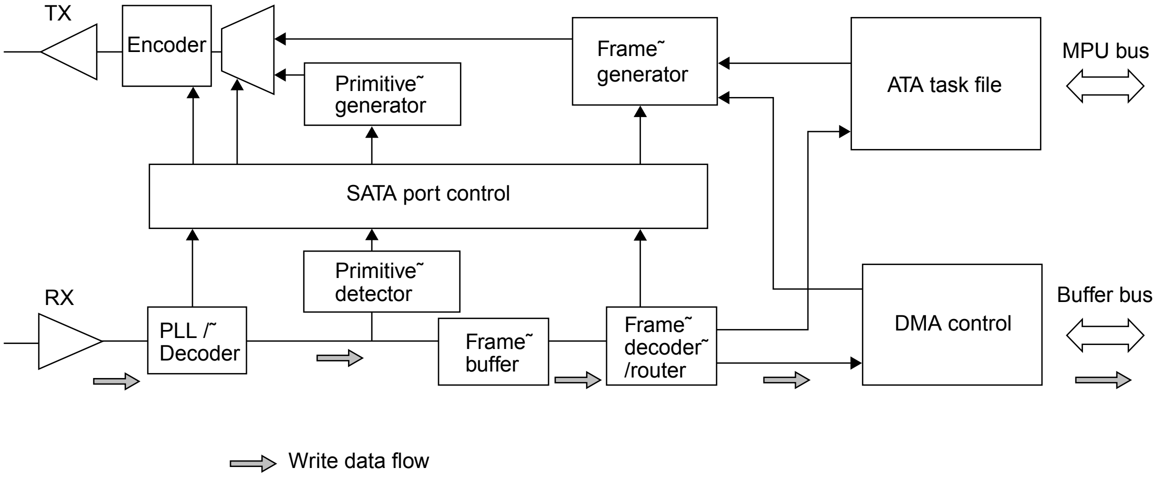


Figure 5
SATA host interface control.

small number of transceivers. To realize this loop topology, the FC ports need a control function for the loop topology. This function is called the loop control (Figure 3).

6.1 Loop control

During loop control, each port within the loop decodes the Primitive sent from the upstream port of the data flow in the loop, determines whether to forward the received Primitive as is or replace it with a new one that it generates, and then sends the Primitive downstream. Therefore, by relaying Primitives around the entire loop, every port in the loop can 1) recognize the loop status and requests from other ports, 2) be linked with each other, and 3) control the loop.

The loop is initialized after the power supply of each port in the loop is turned on, and the loop address of each port is decided during this initialization. Next, while exchanging Primitives embedded in the loop addresses with the other ports, competition for possession of the loop (loop arbitration) is controlled, and one of the ports that are competing for possession of the loop obtains the exclusive right of loop control. The port that obtains the right selects a partner port and establishes a point-to-point connection with that port. Frames and Primitives are then exchanged between these two ports. While this connection remains established, the other ports are prevented from sending their own Primitives. If a difference in speed between the sending and receiving clocks must be absorbed, Primitives are inserted in or removed from the frame gaps. When processing is completed, the port with the exclusive right releases the connection.

Because several ports are linked and operate as described above, interoperability (including the synchronization of operations) must be strictly observed for systems that are constructed using HDDs of different manufacturers. It is not an exaggeration to say that the FC plugfests are held to achieve this.

Turning to the Primitives and frames, loop control determines whether a port that receives a frame from an upstream port accepts the frame, forwards it to a lower port, or sends its own frame to a downstream port. Therefore, loop control acts like a frame router. Moreover, if loop arbitration is performed with all of the ports in the loop linked and uniform services are achieved, loop control acts like a distributed Fabric switch.

6.2 Optimizing the distribution of buffer transfer capability

This section lists the circuits that use the data buffer when FC is used and gives an example of the data transfer rate required by the main circuits.

1) FC interface: 425 MB/s × 2 (4 Gb/s × 2)
2) Media read/write: 100 MB/s
3) ECC generation/correction: 100 MB/s
4) MPU access
5) RAID exclusive OR operation: 100 MB/s
6) Form control constant table
7) Refresh control

As the above example shows, the maximum buffer load is more than 1150 MB/s.

However, the buffer transfer capability is generally lower than this. For example, assume a four-byte width to match the internal bus of the FC controller and assume that DDR RAM operates at 150 MHz (because of the good availability of 150 MHz DDR RAM). Here, the transfer capability is 1200 MB/s at burst, but the sustain speed is around 800 MB/s due to switching of the addresses and read/write operations. This capability will be insufficient for the two FC interfaces to operate simultaneously at 4 Gb/s.

To handle this insufficiency, methods such as increasing the data width of memory or using high-speed memory to raise the buffer transfer capability are available. However, these methods have problems because changing the bus width or operating clock signals within the controller significantly affects the overall system.

Therefore, allocation of the limited buffer data transfer capability to each process is controlled to optimize distribution of the buffer transfer capability.

The buffer use requests for the above operations have different characteristics.

For media reading/writing, a full service is required without interruption. Otherwise, read/write interrupts can cause rotational delays resulting in a significant reduction of performance. Therefore, only as much transfer capability as required is allocated unconditionally for media reading/writing.

For the FC interface, because frame transfer cannot be interrupted during transfer, a full service without interruption is also required. However, because the start and continuation of data transfer can be controlled, the load can be adjusted using the firmware.

ECC requests are made to correct data errors in the buffer. The frequency of these requests is low, and the time constraints are also moderate.

Other requests have time constraints, but their loads are low.

The method currently used for optimizing the distribution of transfer capability is as follows. First, the order of priority is divided into two stages using a two-stage round-robin arbiter (patrol service type competition arbitration). Then, the maximum continuous access time of each data route is made controllable based on the amount of data in the First In First Out (FIFO) on the route. This service time distribution can be adjusted using software. In this way, the distribution of transfer capability is adjusted based on the FC interface speed and media read/write speed in order to effectively use the limited buffer transfer capability.

6.3 Technologies for handling increased interface speed

Increasing the speed of the FC interface deteriorates the signal quality of the transmission line, and this deterioration needs to be addressed.

To increase the link speed, the clock speed must be increased. The jitter tolerance of the receiver, however, becomes a problem at this time. Clock signal regeneration using a Phase Locked Loop (PLL) to increase the jitter tolerance can handle speed fluctuations; however, it is easily effected by outside noise. To handle the noise, data regeneration by sampling the high-speed clock is used.

In the same way, pre-emphasis at the transmitter side and equalization at the receiver side are used to compensate for the frequency characteristics of the transmission line. However, these characteristics greatly depend on the installation conditions of the customer's system. Therefore, adjustment of the compensation characteristics is made programmable so the compensation characteristics can be adapted to a wide range of customer systems.

7. Future problems

The wide range of requirements that system manufacturers and end users have regarding HDDs include increased speed, increased storage capacity, power saving, and quiet operation. This section discusses the market requirements regarding interfaces and the problems in realizing them.

Because HDDs have already been produced for hierarchical type storages, we believe that HDDs for storage systems will differentiate into those that offer higher performance and those that offer higher storage capacity. The demands are that both types of HDDs support multiple interfaces to handle the internal interfaces of storage systems. The reason for these demands, which exceed the conventional use of HDDs, is assumed to be due to the fast data transfer rates of serial interfaces. An efficient development method that meets these demands is to modularize the interface controllers. This method improves the development efficiency and also ensures compatibility at the SCSI command level.

In addition, demands are being made to further increase the transfer rates of the interfaces. To meet these demands, the FC transfer rate will be increased from 4 Gb/s to 8 Gb/s, the SAS transfer rate from 3 Gb/s to 6 Gb/s, and the SATA transfer rate from 1.5 Gb/s to 3 Gb/s. These increases will be achieved by doubling the clock speeds of the circuits. For the transfer capability of the internal buffer of a dual port, demands are being made to double the clock speed of FC transmission lines from 4 to 8 Gb/s for the two ports; that is, to quadruple the overall clock speed. However, this could cause a buffer bottleneck; therefore, as discussed in this paper, continuing optimization of the distribution of buffer transfer capability is essential.

Moreover, to minimize the effect that increasing the speed has on the transmission line, the transmission line's characteristics are being investigated. We believe it is essential to make the appropriate response while watching the trends of standardization.

The specifications for a CE-ATA interface for use in consumer HDDs have recently been established,[9] and this interface will use a reduced ATA command set. The new interface was designed to provide drives with the optimum cost and performance for new uses. Moreover, the interface is being optimized from the specification stage based on where and how new drives will be used. This interface should be seen as the culmination of the simplification of the HDD serial interface.

In addition, various types of HDDs can be used in the same network because of the network functions that a serial interface provides. For a RAID system, the possibility of selecting complicated data arrangements and data transmission routes has increased even more. In response to these trends, how to strengthen the data integrity functions of not only the data transmission routes but also the entire system has become a problem. With this in mind, standardization of the Trusted Computer Group specifications[12] as the End-To-End Data Protection[11] and drive security functions will continue to increase in meaning.

8. Conclusion

This paper described the serial interfaces used as the controllers of current HDDs and introduced the technological problems of future HDD interfaces and their solutions.

Our customers have demanded optimum drives to meet the new uses and market demands of HDDs, and we will continue to use our vast accumulation of technologies to develop timely, stable products to meet customers' needs.

References

1) X3T11/Project1133D: Fibre Channel Arbitrated Loop (FC-AL-2). Rev 7.0, 1999/4/1.
2) Serial ATA WG: High Speed Serialized AT Attachment. Rev 1.0, 2000/11/15.
3) X3T10/Project1601D: Serial Attached SCSI-1.1. Rev 9, 2005/3/18.
4) X3T13/Project1532D: AT Attachment with Packet Interface-7 volume1-Register Delivered Command Set, Logical Register Set, Logical Register set. ATA/ATA-7V1, 2004/4/21.
5) SCSI Parallel Interface-4 (SPI-4). Rev 10, 2002/05/06. *http://www.t10.org/ftp/t10/drafts/spi4/spi4r10.pdf*
6) CE-ATA WG: CE-ATA Storage Interface Specification. Rev 1.001, 2005/6/14.
7) X3T11/Project1162DT: Fibre Channel Private Loop SCSI Direct Attach (FC-PLDA). Rev 2.1, 1997/9/22.
8) INCITS T11/Project 1235-DT/Rev 2.7: Fibre Channel - Fabric Loop Attachment (FC-FLA). Rev 2.7, 1997. 8. 12.
9) MMCA Technical Committee: The Multi Media Card. Ver 3.31, 2003/3. p.10.
10) A. X. Widmer and P. A. Franaszek: A DC-Balanced, Partitioned-Block, 8B/10B Transmission Code. *IBM J. Res. Develop*, **27**, 5, p.440-451 (1983).
11) INCITS T11.2/Project1316-DT/Rev 12.1: Fiber Channel — Methodologies for Jitter and Signal Quality Specification — MJSQ. Rev 12.1, 2003. 12.7, p.26.
12) X3T10: SCSI Block Commands-2 (SBC-2). Rev 16, 2024/11/13. *http://www.t10.org/ftp/t10/drafts/sbc2/sbc2r16.pdf*
13) INCITS T10 SAT Working Group: SCSI/ATA Translation(SAT). Rev 4, 2005. 5. 17. p.xv Foreword.
14) Jim Coomes: SBC 32 Byte Commands for End-to-End Data Protection. Rev 7, 2004/4/21. *http://www.t10.org/ftp/t10/document.03/03-307r7.pdf*
15) TCG Peripherals Work Group: TPer & MCTP Requirements. Ver 1.0 Rev 0.03, 2004/8/13.

Masakazu Kawamoto graduated National Defense Academy of Japan in 1969. He joined Fujitsu Ltd. in 1970, where he has been engaged in the development of file controllers. He has also been engaged in research and development of storage interface technology since 1990.