DESIGN CONSIDERATIONS FOR DUAL-DENSITY DISKETTE CONTROLLERS

An expanded outlook of design factors for a dual-density diskette controller involves system integration of the diskette drive and covers bit packing densities, data recovery methods, input and output interfaces, and available methods of accessing data

John Worden Magnetic Peripherals, Incorporated, Oklahoma City, Oklahoma

Implementation of flexible diskette drives as low cost data processing peripheral memory devices is increasing rapidly in terminals and standalone systems, where storage capacity, performance, and capability to random-access data make them suitable for on- and offline information and software applications. Previously limited by the ability to operate using single-density, single-sided media, the storage capacity of flexible diskette drives quadrupled during the past year when units capable of both double-density and 2-sided media read/write operation were introduced. However, since existing single-sided, single-density-encoded diskette drive controllers were incapable of operating 2-sided, double-density-encoded devices, a major effort was needed to develop controllers that were capable of handling the double-density drives.

The most useful diskette controller will operate with either single-sided, single-density-encoded diskettes or 2-sided, single- or double-density-encoded diskettes, ie, a dual-density diskette controller. A controller that would provide IBM compatible format for data interchange using single-density-encoding, and would operate with double-density capability under program control for file use, would furnish maximum diskette flexibility and storage capability and would maintain current media interchange standards.

Implementation of these criteria and considerations for the dual-density controller demands particular attention to the hardware/software interfaces of both the diskette drive and the host computer. Specifically, the controller interface must consider diskette format, read/write-erase head design, data encoding and recovery means, and diskette drive requirements. The host computer interface must be concerned with software command-level interface, logical interface between the two devices, and the timing relationships necessary for contiguous-sector data transfers—the optimum condition for diskette data transfer.

Soft- and Hard-Sectored Formats

The most commonly used diskette format is the single-density, soft-sectored (or IBM 3740 compatible) type, wherein a diskette is preformatted with individual record identification fields followed by the data field (Fig 1). A 2-sided diskette doubles the data capacity of a single-sided diskette; similarly, double-density devices double both capacity and data transfer rate of the single-density devices. Storage capacities for soft-sectored diskettes are specified in the Table. Increasing the number of bytes per sector and decreasing the number of sectors per track actually increases the total storage capacity of the diskette because track capacity is more efficiently utilized. For single-density IBM format of 77 tracks, 74 are used for data storage,

Soft-Sectored Diskette Capacities

	Single-Sided Single Density	2-Sided Single Density	Single-Sided Double Density	2-Sided Double Density
Diameter	7.88" (200.2 mm)			
Package size	8 x 8" (203.2 x 203.2 mm)			
Coating	Magnetic Oxide			
Total tracks/diskette	77	154	77	154
Data tracks/diskette (includes index)	74	148	74	148
Data transfer rate	31.2k bytes/s	31.2k bytes/s	62.5k bytes/s	62.5k bytes/s
Spindle speed	360 rev/min			
Average latency	83.3 ms			
Unformatted capacity				
bytes/track	5208	10,416	10,416	20,832
bytes/disc (74 tracks)	385,392	770,784	770,784	1,541,568
Formatted capacity (bytes/sector for)				
26 sectors/track	128	128	256	256
15 sectors/track	256	256	512	512
8 sectors/track	512	512	1024	1024
4 sectors/track	1024	1024	2048	2048
2 sectors/track	2048	2048	4096	4096
1 sector/track	4096	4096	—	—
Bytes/diskette (74 tracks for)				
26 sectors/track	246,272	492,544	492,544	985,088
15 sectors/track	284,160	568,320	568,320	1,136,640
8 sectors/track	303,104	606,208	606,208	1,212,416
4 sectors/track	303,104	606,208	606,208	1,212,416
2 sectors/track	303,104	606,208	606,208	1,212,416
1 sector/track	303,104	606,208	—	—
Data reliability				
Recoverable errors	1 in 10^9 bits			
Unrecoverable errors	1 in 10^{12} bits			

two as alternates for bad tracks, and one for maintenance purposes. Track 00 is the index track and certain sectors within it are defined for specific data. The controller design must consider bad tracks, variable sector sizes, and single and double densities.

Hard-sectored techniques use equally spaced holes around an inner circumference of the diskette. These holes are sensed by a photocell to define the beginning of each sector location (Fig 1). A hard-sectored diskette with 32 sector holes, 128 data bytes/sector, and 4096

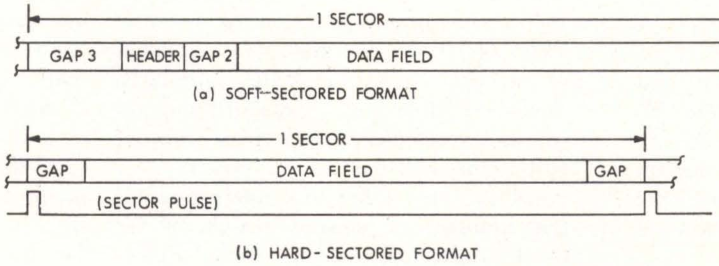


Fig 1 Sectored diskette formats. Soft-sectored diskette consists of two fields (header and data) per sector; hard-sectored diskette may contain only one field (data) per sector. Hard-sectored diskettes allow greater data storage per track because header fields are eliminated

bytes/track, represents 23% more storage capacity than a similarly encoded soft-sectored diskette.

For a soft-sectored diskette, the controller locates a sector by searching for and interpreting the contents of the header field. In 3740 compatible format the header contains track number, sector number of the following data field, diskette side (0 or 1), and length of the following data field. Once the desired header field is located, data contents of this sector may be read or written by enabling the read or write gate at a fixed time interval from the header field.

To read or write a hard-sectored diskette, the controller uses a sector counter register, which is initialized to zero by the singular index pulse from the diskette drive. The sector counter is then updated with each sector pulse, and the controller locates the desired sector by comparing contents of the sector counter with the desired sector value. Data contents of this sector are then read or written by enabling the read or write gate at a fixed time interval from the diskette drive sector pulse.

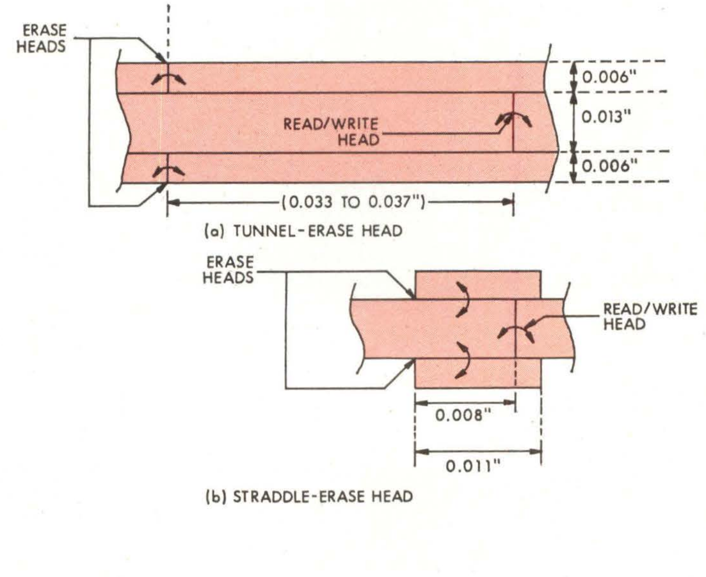
A hard-sectored data field may contain an imbedded header for read verification, whereas a soft-sectored format usually contains only system data. Start of data field bytes may also be different. Media interchangeability and incompatibility of separate hard- and soft-sectored diskette controllers are key design considerations. Since current IBM devices do not use hard-sectoring, compatible designs do not require additional logic to handle hard-sectored operation.

Read/Write-Erase Head Design Impact

Presently, two types of read/write and erase head constructions are prevalent—tunnel erase and straddle erase (Fig 2). Both types consist of a single read/write magnetic transducer and two erase magnetic transducers. The read/write transducer senses or creates magnetic field flux reversals on the diskette media in line with disc rotation. These flux transitions are interpreted as digital data bits by the controller according to the data encoding scheme used. Erase transducers create a separation band between adjacent tracks on the diskette: a magnetic field in line with disc rotation for tunnel erase and at right angles to it for straddle erase. The tunnel-erase design has the advantage of minimizing the noise influence of data stored in adjacent tracks; the tunnel erase geometric construction produces a more clearly defined erase band, thus improving signal to noise ratios.

An advantage of straddle erase heads results from the shorter distance from the read/write head to the end of the erase head. This allows erase current to be turned on and off earlier, while still ensuring erased bands the entire length of the written data fields. (Note that this timing is affected by different head velocities and packing densities at the inner and outer tracks.) Shorter erase times allow shorter gaps between data or data header fields on the diskette, which permits more recoverable data to be written on the diskette. From a cost viewpoint, analysis shows insignificant manufacturing advantages of one head type over the other when yield, material, and labor factors are considered. However, from a business standpoint, the tunnel-erase head has a definite advantage due to its wide acceptance in the industry and its multiple sources.

What type of erase head to use in conjunction with a diskette controller design is mainly a system consideration. Most drives incorporate a fixed delay interval to control erase current turn-on and turn-off; ie, erase-coil control is not accessible by the controller. If IBM format is used, either head type should allow data to be read or written, since IBM equipment uses a tunnel-erase head. However, if nonstandard IBM format is used, with gap times between data and/or data and header fields determined by the controller for the optimum performance characteristics of a



(a) TUNNEL-ERASE HEAD

(b) STRADDLE-ERASE HEAD

Fig 2 Read/write and erase head types. Tunnel-erase and straddle-erase heads are independent conventional magnetic transducers that create magnetic field at diskette surface proportional to current applied through individual read/write or erase coils. Principle is common for both erase head configurations. Major differences are constructional and geometric, which create head views shown looking from diskette surface

straddle erase head, this format can not be read or written if the drive uses a tunnel-erase head.

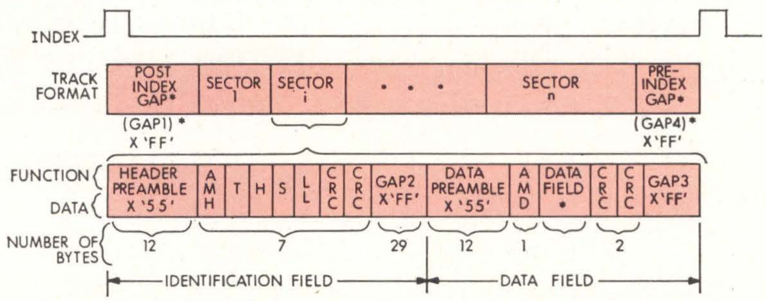
Dual-Density Format Considerations

Modified frequency modulation (MFM), double-density-encoded, soft-sectored diskette format utilized by the CDC 9474 controller (see Fig 3) is generally the same as the IBM 3740 single-density, soft-sectored case except for different gap lengths, preambles, data contents, data transfer rates, and address marks. Each data sector is defined with a header or identification (ID) field and a data field. ID field consists of a preamble (PRE) that allows the controller to differentiate between MFM encoded clock and data bits, lock in a phase-locked loop data detector or synchronize the data recovery detector, and enable an address-mark decode. The unique address ID mark (AMH) allows controller electronics to group each succeeding 8-data-bit group into data bytes that define meaningful binary data, thereby establishing byte synchronization. ID field also contains track (T) and head (H) number for position verification of the head, sector (S) value in the ID field that identifies the data field sector location, and data record length field (LL) that informs the controller how many bytes of data are to be transferred to or from the data field. The 2-byte cyclic redundancy check (CRC) characters are compared to regenerated CRC bytes during reading to verify data.

Header gap 2 (GAP2) field allows the controller designer to establish the timing relationship relative to the location of the data field. It also permits a tolerance gap that enables drive electronics to switch between header read mode and data field write or read mode. Generally header fields are not rewritten after a format operation. Data fields are written, including preamble, address mark, data, and CRC at a location on the diskette determined by a fixed time interval from the header field. Data field will shift slightly between gaps 2 and 3 due to controller and device tolerances, creating "write splices" of unknown data and clock content. Care must be taken in the design to prevent false address mark detection due to these write splices.

Data fields are read by enabling read electronics a fixed time interval after the desired header (ie, past possible write splices). Purpose of the data field preamble is identical to that of the ID field preamble. Data field address mark (AMD) establishes byte synchronization and identifies the data field as valid data or as a control field. For data integrity, regenerated CRC bytes are compared to recorded CRC bytes.

Logic implementation of a hard-sectored diskette controller differs from that of soft-sectored controllers; however, data rate logic is the same. Read/write timing is controlled by occurrence of the sector pulse, and sector location is determined by counting sector pulses relative to the index pulse, or possibly by reading the imbedded header of the previous sector. In practice, the entire sector is written, including preamble char-



AMH = HEADER ADDRESS MARK
DATA = X '0D'
CLOCK = X '50'
T = TRACK VALUE 0 TO 76
H = HEADER NUMBER (0-SIDE 0, 1-SIDE 1)
S = SECTOR (1 TO 26)
LL = DATA RECORD LENGTH
X '01' = 256-BYTE SECTOR
X '02' = 512-BYTE SECTOR
X '03' = 1024-BYTE SECTOR
X '04' = 2048-BYTE SECTOR
X '05' = 4096-BYTE SECTOR
CRC = CYCLIC REDUNDANCY CHECK CHARACTERS $(X^{16}+X^{12}+X^5+1)$

AMD = DATA ADDRESS MARK

	DATA	CLOCK
VALID DATA	X 'OF'	X '50'
CONTROL FIELD	X 'OE'	X '50'

SECTORS PER TRACK (n)	DATA FIELD LENGTH (BYTES)	GAP 1* (BYTES)	GAP 3* (BYTES)	GAP 4* (BYTES)
26	256	146	57	≈ 506
15	512	146	87	≈ 340
8	1024	146	117	≈ 638
4	2048	146	147	≈1238
2	4096	146	177	≈1598

Fig 3 MFM diskette format. Double-density soft-sectored diskette format offers twice data storage capacity of single-density diskette. Format is similar to IBM 3740 format except for differences in gap widths, sector sizes, preambles, and address marks. Address marks are still defined with illegal missing blocks under MFM encoding rules

acters, address mark, imbedded header in the data field for possible read verification, data field, and CRC. For sector reads, preamble and address mark are functionally equivalent to their soft-sectored counterparts; however, hard-sectored address marks usually do not contain illegal missing clocks. Imbedded header of the sector being read can be used only for sector verification as it is really part of the data field.

Encoding Schemes and Data Recovery Methods

Four encoding schemes in current use with flexible diskette drive devices are frequency modulation (FM), also referred to as double-frequency (DF) coding; modified frequency modulation (MFM); modified-modified frequency modulation (M^2FM); and group code recording (GCR). (See Refs 1-5). Single-density IBM 3740 compatible diskette units use FM encoding.[6] Double-density diskette drives and their associated controllers use either MFM, M^2FM, or GCR encoding. Hardware complexity generally increases in the following order: FM, MFM, M^2FM, and GCR. Implementation cost and/or IBM interchange compatibility will greatly affect the double-density encoding scheme chosen. Key factors to be considered for all schemes and methods are data recovery method, peak-shifting for double-density encoding, and coding/decoding.

Data recovery for FM encoding can be accomplished by either a one-shot pulse or a phase-locked oscillator (PLO). Since the latter is mandatory for all double-density encoding schemes, it should be used with both single- and double-density encoded diskettes. For FM encoding, the read head signal frequency components[1] are 125 and 250 kHz—a subset of the 125-, 166.7-, and 250-kHz frequency components of MFM encoding.[1] For M^2FM encoding, frequency components[1] are 100-, 125-, 166.7-, and 250-kHz, which require a wider bandwidth but still have FM frequencies as a subset. However, GCR frequencies[1] are 104.17-, 156.25-, and 312.5-kHz, without any FM encoding frequencies as a subset.

The PLO data recovery method has several advantages over the one-shot method even in single-density controller operation. As a result of its inherent locking onto the data stream frequencies, the PLO averages and negates the effects of phase shift due to different motor speeds, drive bit shifting, and circuit delays. In contrast, a one-shot pulse interval is fixed and is not modifiable to the incoming data stream frequencies. Additionally, a PLO automatically accounts for speed variations in defining window times for data or clock bits, whereas speed variations must be subtracted from one-shot window times.

Peak shifting[1,2] is a characteristic of magnetic discs that results from interference of adjacent bit flux reversals, which cause a flux reversal to be read slightly before or after its nominal time. Although not critical in single-density encoded diskettes, due to wider data window times, peak shifting becomes critical in the inner tracks of double-density encoded media. Critical bit patterns are 011, 1000, 110, and 0001, with pulses occurring according to MFM encoding criteria. In the innermost tracks (43 to 77) peak shifting may be accounted for through "write precompensation[1,2]" where the appropriate bit either delays or precedes the theoretical bit times by 250 ns. This holds true for MFM and M^2FM encoding but is of questionable value for GCR.

In addition, the controller should be able to accommodate two different write currents—a high value for the outermost tracks (00 to 42) and a low value for the innermost tracks (43 to 77)—for better data recovery reliability and compatibility with single-density IBM formats. Controller requirements involving both single- and double-density coding can use MFM advantageously for double-density encoding, with the resultant simpler design also furnishing the FM frequency component subset required for single-density encoding.

Diskette Drive Control Lines

The write data line to the disc drive electronics (Fig 4) is a single serial line at the hardware level with write data being composed of 250-ns pulses for data or clock bits. Associated with it are a write enable line that brackets the write data and a low write current line that is turned on if tracks 43 to 77 are being written. Composite data and clock signals are transmitted over the single read data line. Two other read-lines—read data separated and clock separated—are available, but are not generally used for formats having missing clock patterns in address marks.

A head load signal line loads the read/write head against the medium; a head select line is used to select the appropriate head for double-sided operation. Head movement is usually controlled by a 10-μs pulse (3- to 10-ms step rate) that steps the head one track at a time, acting in conjunction with a direction signal that moves it toward innermost or outermost tracks, one track at a time. Some drive units use separated pulsed step-in and step-out signals instead. Generally, the desired track is specified to the controller, which should be able to calculate the direction and number of tracks to be crossed to reach the specified track from the current one, and to handle interim assigned replacement tracks (two maximum) while performing this seek.

Initial radial reference point for soft-sectored diskettes is established at power turn-on by stepping the head out one track at a time until the track 00 positioning signal is detected. Rotational timing is available in the index/sector pulse to give a rotational reference point on the diskette; normally, this signal is used only during formatting and for diskette speed checks. For hard-sectored diskettes, a sector pulse is generated for each sector that is photoelectrically sensed on the diskette; this pulse is generally encoded into and with the index hardware lead, but may be obtained via a separate lead.

Drive status includes a ready line indicating that medium is in place and that the drive is ready for operation; a write fault line monitoring the interactive control of the write data, write enable, and head load signals; and a write protect line indicating when the write protect slot in the diskette is uncovered. Unit select and write fault reset input leads provide signals to select a particular drive and to reset a drive after a detected write fault, respectively.

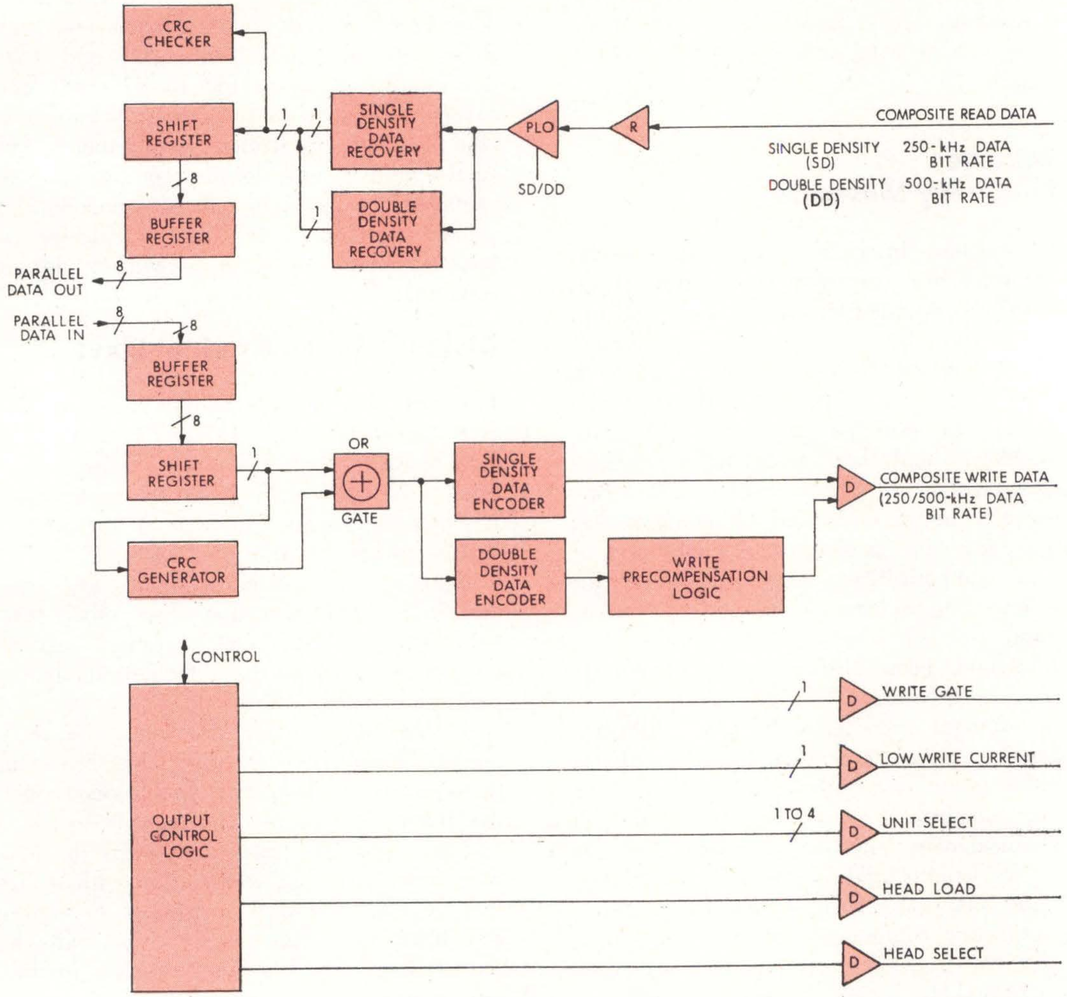


Fig 4 Diskette drive serial data interface. Data are available if unit is selected, head is selected and loaded, and write gate is not enabled. Write data frequencies, data encoding, and data recovery are different for single- and double-density encoded diskettes. Data are written on diskette by serial write data line if enveloping write-gate signal is true, unit is selected, and head is selected and loaded. Double-density diskettes may also require write precompensation. Low write current line is available to reduce write current for innermost tracks to minimize peak shifting. This functional system contains some logic that may be shared

Typical power requirements of diskette drives are 120 or 240 Vac, 24 Vdc, or ±5 Vdc. The designer should evaluate the feasibility of either the controller or an alternate source providing the power.

Subsystem Logical Interfaces

Subsystem interfaces for diskette controllers (Fig 5) usually involve either parallel data and control signal lines connected to an associated host processor or serial transfer of both data and control information between drive/controller and a remote processing system over a hardwired or communication type circuit. The obvious advantage of parallel over serial data transfer is speed—more than 35 times faster in most applications—but this is often offset by the simpler hardware (and software) requirements of the RS-232-C serial interface and baud rates transmitted over data set telephones. However, a standardized software level interface is not implied, and various communication protocols are not covered. Note that for either interface the controller must contain logic to handle a high level software interface, which specifies desired unit, track, sector, direction of data transfer, status exchange, and the ability to obtain desired single or double density format.

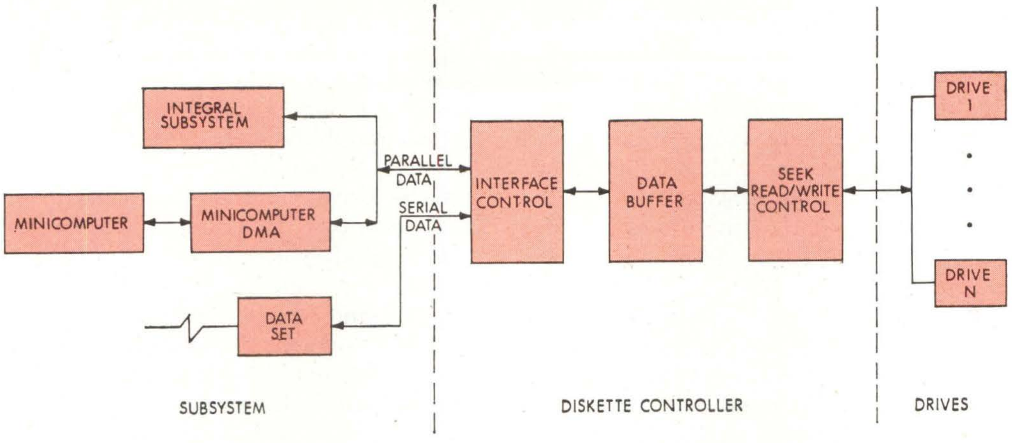
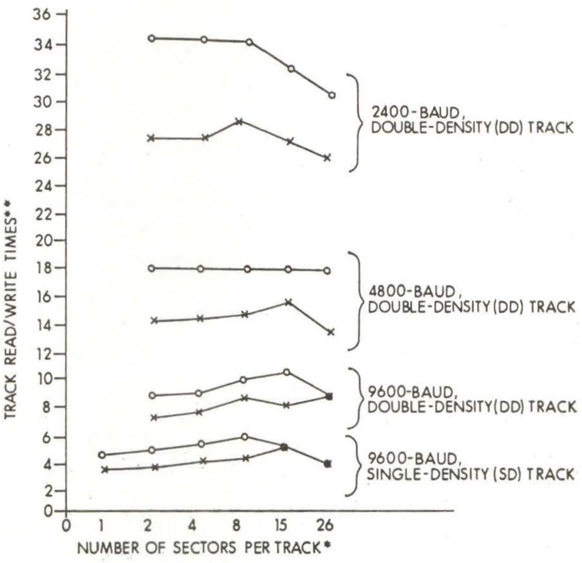


Fig 5 Diskette controller subsystem interface. Subsystem interfaces usually involve either parallel data and control line interface or serial transfer of both control and data over hardwired or communications interface. Advantage of speed via parallel interface must be traded off against low cost standard RS-232-C communications interface

For maximum flexibility, drive controllers use generalized parallel interfaces with character present and character accept handshake logic, since fast processor interfaces vary between manufacturers. Normally these controllers contain data buffers to account for the mismatch between data rates of the drive and parallel interface. A 2-byte buffer is adequate to handle data transfers on-the-fly if data transfer rate and byte response time of the host processor are less than 16 μs. However, if the interface has a fast data transfer capability but requires a relatively long time to get control of the data transfer channel, sufficient buffering must be provided to cover this extended access time.

Serial (RS-232-C) data transfer is generally effected at a maximum rate of 9.6k baud. Since the serial interface can not handle straight-through diskette data transfer rates in real time, data must be buffered at least on a sector basis.



NOTES: ● = ASYNCHRONOUS
× = SYNCHRONOUS

* SECTOR SIZE FOR DOUBLE-DENSITY IS TWICE DOUBLE FREQUENCY. THE NUMBER OF BYTES PER SECTOR IS PER TABLE 1

** INCLUDES SECTOR LATENCY AND ASSUMES A SINGLE SECTOR BUFFER. DOES NOT INCLUDE COMMUNICATION-LINE-TURNAROUND TIME

Fig 6 Read/write times for sectors per track. Track read/write times for RS-232-C interface are dependent on number of sectors per track, number of data bytes per sector, baud rate, and communications interface character structure (asynchronous vs synchronous). Double-density track takes longer than single-density track due to double number of bytes per sector. Impact of two extra bits (start and stop) for asynchronous character structure is most noticeable at lower baud rates. Impact of diskette rotational latency has visual impact on slope and parallelism of curves at all baud rates and number of sectors per track. Effectors of all communications protocols are not included

Contiguous-Sector Transfers and Interface Timing

Seek time for a data file is a function of file placement on the diskette, and is defined in terms of both track-to-track traverse and read/write head settling times. Typical times vary from 3- to 10-ms track-to-track and 10- to 20-ms settling time; thus a 77-track seek could require from 248 to 770 ms. Careful organization of data files on the diskette plus the final position of the read/write head after a file access operation can minimize seek times.

Contiguous sector reading or writing using a parallel interface results in the most efficient transfer of data between the drive/controller subsystem and its host processor. For example, an entire track of data may be transferred in one diskette revolution (0.167 s) plus the diskette latency (time to find the first sector), if contiguous sectors on the diskette can be read or written. The first sector accessed must always allow for rotational latency, ie, 0 to 167 ms (83 ms avg). However, where the next sector of data to be transferred is just missed, necessitating another complete rotation of the disc before writing or reading can occur, a delay of up to 26 revolutions (or 4.342 s) will result if 26 sectors/track are to be transferred. To read or write contiguous sectors with minimal data buffering, data transfer rate to the host processor must be equal to or greater than that of the diskette drive (62,500 bytes/s for a double-density drive, or 31,250 bytes/s for a single-density drive).

Another critical design parameter for consecutive sector transfers is the amount of information that must be transferred between host and controller. This information—transferred between the last data field byte and the preamble of the next data field—verifies a successful transfer of the previous sector and specifies the next sector transfer. Information interchange may vary from a complete drive status transfer; host specification of the next sector, head, and track; to a single good/bad status lead with an automatic controller sector, head, and track update; or to a simple read/write command. If system constraints disallow consecutive sector transfers, sector interleaving—separation of sequential logical sectors by one or more physical sectors—should be evaluated for enhancing system performance.

Contiguous sector read or write operations are not possible using a single sector buffer with the serial RS-232-C interface. Achievable track read/write times are shown in Fig 6. A timing analysis should be performed to assure that, after line turnaround times and line protocol times have been executed, the diskette did not just miss a multiple of its revolution time. If baud rate and timing just miss a multiple of revolution time, 4.3 s are added to the time for a 26-sector read and/or write. Parallel interface controller design techniques, as well as multiple sector buffers, can be used to eliminate wasted revolutions encountered with the serial interface. Nonparallelism of asynchronous and synchronous transmission modes (Fig 6) indicates this wasted revolution time. Effects of the two extra bits required for asynchronous transmission (start and stop), plus the effect of rotational latency due to the different time intervals employed, can be seen. Note the greater effect of the two extra bits at lower transmission rates. Slopes of the curves are affected primarily by the greater number of information bytes required for the lesser number of sectors per track and by the relationship of the diskette rotational position after data transfer to the next sector.

It is evident that software-interface timing considerations are concerned mainly with data transfer, while the controller handles head positioning and timing for diskette read and write operations with respect to the physical location of the sector on the diskette.

Summary

A dual density controller doubles data storage capacity of a diskette. Cost impact is in the controller design and not in the drive or medium. Industry standardization and media interchangeability are achieved if the IBM 3740 single-density soft sectored format is used. This format also permits utilization of either multiple-sourced tunnel erase head diskette drives or single-sourced straddle-erase head drives. System data storage capacities are increased either by utilizing a hard-sectored format or by redefining the format for straddle erase head performance parameters; however, media interchangeability and multiple source diskette drive problems are created.

Higher cost for a dual density controller results from use of a PLO to recover both single and double density encoded data, different or more complex data recovery and encoding circuitry (FM versus MFM, M^{2}FM, or GCR), different decoders for preambles and address marks, different gap sizes in terms of byte counts, doubled data transfer rates, and write precompensation for double density data encoding.

Diskette controller to processor interface design is determined mainly by performance tradeoffs. While a high speed parallel interface for both data and control yields greater performance, no standardized parallel interface exists and this interface differs for each processor. In addition, controller implementation, software/hardware interface, and diskette format impact realizable system performance due to the timing relationships required to achieve optimum performance levels with consecutive sector data transfers. Although yielding a lower system performance, the RS-232-C communications interface for both data and control implements an industry-standardized electrical interface. Since many communications protocols exist, final diskette controller design will depend upon the selected method. The determining performance factor is the communications baud rate, which requires at least a sector-length buffer in the controller, but timing considerations are more critical for the parallel interface.

References

1. D. J. Kalstrom, "Simple Encoding Schemes Double Capacity of a Flexible Disc," *Computer Design*, Sept 1976, pp 98-102
2. R. C. Franchini and D. L. Wartner, "A Method of High Density Recording on Flexible Magnetic Discs," *Computer Design*, Oct 1976, pp 106-109
3. P. S. Sidhu, "Group-Coded Recording Reliably Doubles Diskette Capacity," *Computer Design*, Dec 1976, pp 84-88
4. Magnetic Peripherals, Inc, "Product Specification (77834772) for Flexible Disc Drive Subsystem, Model 9474," Nov 1977
5. Magnetic Peripherals, Inc, "Product Specification (75888213) for Flexible Disc Drive, Model 9406," Feb 1978
6. IBM Corp, OEM Publication "IBM Diskette OEMI," Pub No. GA 21-9190-2

John Worden is a principal electrical engineer in the software engineering section of the systems and engineering services department at Magnetic Peripherals, where he provides systems engineering support for flexible disc and cartridge disc drive product lines. He holds a BSEE degree from the University of Oklahoma and an MSEE degree in large scale computing devices from the University of Pennsylvania, Moore School of Electrical Engineering.