Controller for a flexible disk*

by RONALD G. HARRIS, JAMES E. SUSTMAN and JOHN F. MCDONALD

Yale University
New Haven, Connecticut

INTRODUCTION

In late 1971 several manufacturers of mass recording media\(^1\) began to announce initial specifications for a class of simple low-cost magnetic disks based on the IBM flexible disk initial microprogram system loader.\(^2\) These disks were marketed for data logging and other applications where repeated read-write activity was minimal. Their extreme low cost made it possible to integrate them into inexpensive hardware configurations as a substitute for conventional disk drives. The Memorex 651 has characteristics that are typical of the available Floppy Disk drives and is selected here as the device to be interfaced.

DRIVE CHARACTERISTICS

The removable medium of the Memorex 651 is a flexible 4 mil Mylar platter about the size of a 45 rpm record, costing about $2 in large quantities. It is similar to the IBM 23FD-11. Only one surface is used for recording, the density is 3100 bits per inch on the innermost track, data encoding is Frequency Modulated (FM), and the maximum capacity of the platter is 2.5 million bits, assuming the most compact organization. The platter rotates at a speed of 375 rpm, the data rate is 250 kilobaud, and the read/write head is positioned by a simple stepping motor. There are 64 tracks on the Memorex 651 (77 track IBM compatibility is available with the Memorex 652).

A single electronics card handles all level conversion, signal amplification, and power driving appropriate to the internal operation. Communication with external TTL control logic is through line receiver/driver pairs specified by the manufacturer.\(^3\)

The signals that concern the interface designer are illustrated in Figure 2. These signals are functionally divided into four groups which (1) control head motion, (2) sense circumferential position on the disk, (3) read the disk, and (4) write the disk. All signals to and from the disk are asserted low, that is, a (TTL) low level is interpreted as a logic 1.

The head motion logic receives three control signals, STEP IN, STEP OUT, and HD LD, and returns one status signal, TRACK0. Asserting HD LD (Head Load) causes a pad to press the recording surface of the platter against the head. The head is not settled until 20 milliseconds after the assertion of this signal; the interface must wait before indicating that the load operation is complete.

A 10 microsecond pulse on STEP IN or STEP OUT will move the head toward or away from the center of the disk. The head takes 10ms to move between adjacent tracks and an additional 10ms to settle on the destination track. When the head is positioned over the outermost track (track 0 by convention), it closes a switch, asserting TRACK0, which is the only data available to the interface about the radial position of the head.

The drive senses circumferential position on the disk optically, returning two signals, SECTOR H, and INDEX H. The assertion of SECTOR H indicates the beginning of a new sector on the disk. The assertion of INDEX H indicates that the next sector is sector 0. Since the drive returns only these two pulses to communicate the circumferential position of the disk, the disk must make one full revolution before the sector count in the interface can be assumed to be valid.

To write on the disk, the interface must assert WT ENB H (write enable) and then transmit the data string encoded as in Figure 3 on WT CLKS.\(^4\) A clock signal is recorded with the data to make the read “self-clocking,” as opposed to systems which have a separate timing track, so WT CLKS is simply the sum of the data and the clock. To ensure proper synchronization during a read operation, a string of 128 zeros is inserted at the beginning of every record followed by a synchronizing pattern (it need only be one bit long) in order to identify when the first word of data has been “framed.” Guard zeros are appended to the end of the record to reduce noise on the last data bit. The computer or DMA channel should insert track and sector identifying information at the beginning of the data record and a checksum at the end to facilitate error detection.

The read logic in the drive separates the clock pulses from the data and gives two signals to the interface, SEP CLOCK and SEP DATA (Figure 2). The 128 bit zero header is necessary to insure correct separation; the controller ignores the header until the framing pattern is in its buffer because drive’s synch circuit may miss the first bit and fail to lock onto the data immediately.

---

* This work was partially supported by the Alfred P. Sloan Foundation.
The drive requires three power supplies, a 5 volt, 1 amp supply for the logic, a 15 volt, 600 ma supply for the read and write amplifiers, and a 24 volt, 2 amp supply for the head position stepping motor. The power supplies are a major fraction of the cost of the interface.

DATA INTERFACE DESIGN

The design presented here is highly modular, so that it may be easily adapted to different computers or I/O architectures. (Figure 4) The modules are:

(A) Read
(B) Write
(C) Sector Match
(D) Track Position
(E) I/O

The I/O module contains the data buffers and all the logic.
to decode the control signals from the computer. We will not describe this module since its details are specific to each computer.

The Read module (Figure 5) performs the conversion of incoming data from the disk drive's serial format to 16 bit parallel format for transfer into the computer. The module also contains the synchronization logic to frame the data at the beginning of a record.

To initiate a read operation the Sector Match module issues ST RD (Start Read). The trailing (rising) edge of ST RD sets the READ flip flop. Once RD ENB H is asserted,
SEP CLOCK is gated from the drive's synch separator to the DATA WINDOW ONE-SHOT. The leading edge of the gated SEP CLOCK initializes the DATA flip flop to zero and triggers a 3 microsecond aperture during which SEP DATA can set the DATA flip flop. If SEP DATA is asserted during this interval, the DATA flip flop will capture it. The falling edge of the DATA WINDOW ONE-SHOT clocks the DATA bit into the shift register and increments the INCOMING BIT COUNTER. The INCOMING BIT COUNTER asserts its carry (low) when the count is 15. The trailing (rising) edge, which occurs after the 16th bit is shifted, clocks the READ BUFFER STATUS flip flop, asserting TAKE H. When the buffer has been read, DATA TO BUS must be asserted to reset the STATUS flip flop.

If the computer or DMA channel does not read the data buffer before the next carry, the new data will overwrite previously buffered data. Thus, data transfer must occur within 64 microseconds for a 250 kilobaud rate and 16 bit buffer.

The Read module synchronizes with the data at the beginning of a record by asserting RD ENB H, enabling the SYNC flip flop. Once enabled, the arrival of the synch pattern in the incoming shift register asserts DATA FRAMED L, enabling the INCOMING BIT COUNTER. The pattern used in this implementation is '1001' (octal) which is not likely to occur during the synchronizing header.

To make the data self-clocking the Write module (Figure 6) must have an internal time-base to generate the clock to which the data is added for recording. To initiate a write operation the Sector Match module issues ST WT (start write). The trailing (rising) edge of ST WT sets the WRITE flip flop, asserting WT ENB H and starting the clock, which generates three timing signals, \( \Phi A \), \( \Phi B \), and \( \Phi C \). These signals provide edges at two timing points equally spaced over a 1 microsecond period. Phases \( \Phi A \) and \( \Phi B \) are nearly identical, except that \( \Phi A \) starts and finishes at a low level, while \( \Phi B \) starts and finishes at a high level.

Rising transitions of \( \Phi A \) increment the OUTGOING BIT COUNTER. The WRITE DATA SHIFT REGISTER shifts or loads on falling edges of \( \Phi B \), insuring that the shift/load level input of the register does not change simultaneously with its clock input. The least significant two bits of the OUTGOING BIT COUNTER are used in the following sequence to generate clock and data pulses with the

---

From the collection of the Computer History Museum (www.computerhistory.org)
proper timing:

<table>
<thead>
<tr>
<th>States</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>QB QA</td>
<td></td>
</tr>
<tr>
<td>0 0</td>
<td>Advance or load shift register.</td>
</tr>
<tr>
<td>0 1</td>
<td>Generate synch pulse on drive's WT CLK line.</td>
</tr>
<tr>
<td>1 0</td>
<td>Enable data to WT CLK one-shot.</td>
</tr>
<tr>
<td>1 1</td>
<td>Generate a pulse on WT CLK line if shift register data bit is one.</td>
</tr>
</tbody>
</table>

Finally, COUNT16 H asserts the clear input of the OUTGOING BIT COUNTER and the load input of the shift register. The next falling edge of φB then loads the shift register and the subsequent rising edge of φA resets the counter.

The load operation transfers the contents of the write buffer to the shift register and asserts GIVE H. In order to insure an uninterrupted flow of data to the disk, the write buffer must be reloaded before the next assertion of COUNT16 H and φC H. If DATA AVL L does not reset the WRITE BUFFER STATUS flip flop before COUNT16 H is asserted again, STOP L is asserted, WT ENB H falls, and the CLOCK stops after one final rising edge on φA (which clears the counter).

SECTOR AND TRACK CONTROL

The Memorex 651 disk platters can be formatted for 32 sectors per track, but the information capacity is reduced to 2.2 megabits, or 88 percent of the maximum capacity, when this is done. With the disk divided into only 8 sectors, 97 percent of maximum capacity is achieved. Since every sector must be written to completion, division into 8 sectors
The Sector Match Module offers a reasonable compromise. The Sector Match Module divides the disk into 8 logical sectors.

The SECTOR COUNTER of Figure 7 is a five bit counter made from a four-bit synchronous counter and a single flip flop for the least significant bit. The three most significant bits define the logical sectors. The index pulse asserts CTR-CLR so that the next sector pulse will reset the SECTOR COUNTER to zero. Whenever a sector match occurs with the sector (SEC0, SEC1, SEC2) specified by the current operation, a pulse is generated for ST RD or ST WT if either is enabled.

Track addressing is handled by the Track Address Module shown in Figure 8. Once the ACTUAL TRACK POSITION counter is set to the correct value, the computer need only specify the target track address (T0, T1, T2, T3, T4, T5) and pulse NEW TRK H. The Track Address Module compares the target track and the current track, and, after waiting 200 ns to complete the comparison gates, a 10 microsecond pulse to either STEP IN L or STEP OUT L. At the same time, a 10 ns delay is generated to allow the head settle. While the result of the previous comparison was not “equal,” the falling edge of the settling pulse is gated to the up or down count input of the ACTUAL TRACK POSITION counter by MVD L, continuing the motion.

The program should initialize the ACTUAL TRACK POSITION counter before the first disk access by asserting TINIT H and NEW TRK H. TINIT H sets the target track (T0, ..., T5) to zero and starts the step out sequence cycling the head toward the home position. A microswitch in the drive senses when the head is home, asserting TRACK0 H which clears the ACTUAL TRACK POSITION counter.

The HD LD H signal must be asserted for the duration of any read or write activity. A pulse is generated on the rising edge of HD LD H which provides sufficient delay for the pressure pad to engage the platter.

When any head motion is taking place, the Track Address Module asserts MOVING L to prevent the controller from attempting to read or write the disk until the head has settled. INDEX H triggers a 160 ms pulse from a retriggerable monostable multivibrator. As long as the drive is up to speed, this pulse is continuously retriggered, enabling READY H, which may be read as a status bit by the computer. Finally, when the unit has READY H asserted and neither RD ENB H or WT ENB H is asserted, DONE H is asserted.

STATUS SIGNALS

If this interface is used in a program transfer controller, GIVE H and TAKE H signal the computer to read or write the device’s buffer. In a DMA controller, these signals instruct the DMA channel to transfer data to or from memory. Read and Write are not symmetric with respect to the action taken when a buffer has not been transferred in time. The Write module has its own clock, enabling it to detect the end of a write cycle and shut itself off. Read, however,
is clocked by the data, and so may receive no further pulses after the last valid data has been read. Therefore the computer or DMA channel, which knows when the last datum has been read, must reset the READ flip flop by deasserting the enabling input, RD H. DONE H signals the completion of a disk operation, i.e. Had Load, Move Head, Read, or Write, and will normally cause an interrupt to the host computer.

THE PROGRAMMER’S VIEW

The four modules described here are the basis of any design for an interface to a specific computer. The details of how specific control lines are driven, or how transfers of data are accomplished, may be found in the computer manufacturer’s interface manuals.\textsuperscript{6,7}

We have built a program transfer controller for the Memorex 651 and the Digital Equipment Corporation PDP-11 using these modules. (Figure 9) The interface plugs directly into a PDP-11 Small Peripheral Slot and requires only the DEC M782 interrupt and M105 bus address modules.

The PDP-11 interface has three program-addressable registers:

1. CSR Control and Status Register
2. TAR Track Address Register
3. DBR Data Buffer Register

Through these registers, the computer program can initiate activities on the disk, transfer data, and determine the status of the disk. In addition, the interface can generate two different interrupts to the computer: a “status” interrupt

Figure 8—The Track Position Module (Head Load timing included)
which indicates the completion of a command, and a "data" interrupt which is a request to transfer the next word of data to or from the DBR.

The commands to the interface are to read or write sector number (SECO, SECI, SEC2), to move the head to track number (T0, ..., T5), and to load or unload the head. The status information available to the computer is head loaded or unloaded, present disk sector, actual track address, ready for next command, require next data transfer, and head home.

SUMMARY

We have presented a design for a simple disk controller for a Floppy Disk. The low cost of both platter and drive make the Floppy Disk an excellent substitute for a conventional disk on a small minicomputer system. The device can be used for both data and the resident operating system since in our experience media wear has proven low and data reliability relatively high (approaching the quality claimed by the manufacturer).

ACKNOWLEDGMENTS

The authors wish to thank C. Minter, R. Tuttle, J. Zornig, M. C. van Leunen and C. Carr of Yale University for their assistance. We would also like to thank P. Weiner of Rand Corporation for his contributions to the success of the project.

REFERENCES

3. Interface Manual for the Memorex 651 Floppy Disk Drive, Memorex Corporation, Santa Clara, California.
5. The TTL Data Book, Texas Instruments Incorporated, P.O. Box 5012, M.S. 84, Dallas, Texas.