CADOMEMO

TO: Distribution

FROM: Bob Thorne/Lou Thompson

SUBJECT: 14" Winchester System Operational Description

DATE: October 12, 1979

The following is a brief description of the operational characteristics of the 14" Winchester Drive Option. This option provides 13.2 megabytes of bulk storage on a single non-removable disk. It is designed to be used with either the 16K or 48K dynamic processor system and will support up to a total of 4 drives per system. A single drive system can be completely self-contained in the standard CADO stand alone cabinet.

The minimum single drive system consists of a Peripheral Bus Adapter, Host Adapter and Winchester Controller/Drive. Each major unit will now be described in greater detail.

I PERIPHERAL BUS ADAPTER

The Peripheral Bus Adapter is composed of a card cage and associated electronics. Its function is to translate the signals to and from connector J1 on the processor board to signals with which the Host Adapter board can operate. In some cases such as the data bus, the Peripheral Bus Adapter translates the bidirectional data bus on the processor to a unidirectional data bus on the Host Adapter. In other cases, such as the processor address bus, the only function performed on the Peripheral Bus Adapter is to permit data from the Host Adapter to be read on to the processor bus. Line driving capability is also provided on various control signals between the processor bus and the Host Adapter. The Peripheral Bus Adapter can accept a maximum of two Host Adapters.

II HOST ADAPTER

The Host Adapter provides the required interface between the Peripheral Bus Adapter and the Winchester Controller/Drive. Each Host Adapter can control up to a maximum of two Winchester drives

resulting in a total system capability of four Winchester drives with two Host Adapters connected to the Peripheral Bus Adapter and two Winchester Controller/Drives connected to each Host Adapter. The main function of the Host Adapter is to provide address decoding for alternate drive selection, as well as bi-directional encoding for the data bus signals.

Since the Peripheral Bus Adapter can support two Host Adapters, provision must be made on the Host Adapter for individual address selection. This is accomplished by jumper selection on the Host Adapter which provides the assignment of output address Hex 80 for one Host Adapter and Hex 84 for the other. As previously mentioned, each Host Adapter can accomodate two drives and therefore the data specified in each output command is used to select the desired drive. A data value of Hex 1 selects the first drive while the second drive is selected by a data value of Hex 2. As an example, if it is desired to select the second drive connected to the second Host Adapter, an output data value of Hex 2 must be given with an OUT #84 command.

III WINCHESTER CONTROLLER/DRIVE

The Winchester controller is a single printed circuit board located on the drive. It is 6800 microprocessor based, with 2K bytes of PROM and 1K bytes of RAM. The controller has been designed to appear to the host processor as a section of main memory, and therefore all data and control operations required by the rigid disk, with the exception of initial addressing, appear as memory read and write operations. The RAM memory in the controller is also used as the source and destination for data which is read from or written to the disk. Data is written to the disk in 256 byte sectors which, because of timing requirements, are physically staggered on each track. For example, logical sectors 0 and 1 are located in physical sector positions 0 and 8. There are 64 physical sectors in each track and these have been logically mapped as the equivalent of two floppy disk tracks. A complete Winchester drive has been mapped to contain an equivalent of 21 single sided floppy disk drives, resulting in a total storage capacity of 13.2 megabytes.

A. Controller Memory Map

The 1K byte RAM located on the controller has been mapped as shown below (all addresses in Hex).

3000-32EF	This section is reserved for 6800 microprocessor work area.
31F0-31FF	This is temporary storage for the 16 byte ID field which preceeds each data sector.
3200	This byte is referred to as the command register and every operation is initiated by writing a Hex 0 into this location.
3201-32EF	This section is reserved for bad track mapping of an individual drive.
32F0-32FF	The first four locations in this section specify the function to be performed, as well as the cylinder, head and sector number. Locations 32F4 through 32FF are not presently used.
3300-33FF	This section is reserved for the data to be written to or read from disk.

The controller RAM provides the central interface between the local 6800 microprocessor and the host system and as such the memory addressing is controlled from three separate sources: the 6800 microprocessor, the Host Adapter and from the local address counters on the controller board. These address counters are used when reading and writing information from and to the disk.

B. Power on Sequencing

When power is initially applied, logic in the drive itself conditions the drive to accept commands from the controller. This is signified by the drive raising the control ready signal (CRDYN) at the same time the program stored in the controller PROM initializes its own PIA and enters the program to sequence up the drive. The controller issues a sequence up command to the drive which causes the motor and start relays in the drive to energize. The drive then waits 6 seconds and the bytes per sector and sectors per track counters are initialized. The drive then checks the period between index pulses to verify that the drive is up to speed. If it is not, the drive automatically enters the sequence down mode and the drive motor is turned off. If the disk comes up to speed with no errors the drive program delays 3 minutes and then positions the heads over cylinder 0. The firmware in the controller then causes the 6800 processor to enter a wait state waiting for the first command from the host interface.

In order to understand the operation of the controller, a typical write data sequence will be analyzed. It should be recognized that the read sequence is very similar, with the major difference being only that of data flow direction. To begin a write operation the data to be written must be loaded into the 256 byte data buffer located in hex addresses 3300 to 33FF. During this time the 6800 microprocessor is in the WAIT phase where it remains until Hex 0 is loaded into the command register location 3200. In addition to the data, the cylinder, head and sector of the target location must be loaded into memory. The function code which specifies a write operation must also be loaded in 32F0. When this set up operation is complete, a Hex 0 is loaded into the command register at address location 3200. Address 3200 is decoded in the controller and used to generate an interrupt to the 6800 microprocessor. The 6800 firmware then causes the processor to read the information previously loaded into locations 32F0 through 32F3 to determine the type of operation to be performed and the target cylinder, head and sector information.

C. Disk Format

Before the disk can be used for normal read and write operations it must be formatted or initialized with the proper identification fields for each sector. This normally will be done only once since the initialization destroys all previous information written on the disk. The initialization process writes the appropriate headers on each track as shown below in Figure 1.

BYTE#	0	1	2	3	4		15
FUNCTION	-	CYL	HEAD	SECT	-		-

Figure 1 Typical Header Layout

When the disk has been completely initialized it is ready for normal data read and write operations.

D. Read Operation

As previously mentioned, before any operation can begin the appropriate information must be loaded into the RAM memory located in the controller. In the case of a read operation the host computer must first load the read opcode, cylinder, head and sector bytes into locations 32F0 to 32F3. During this time the 6800 processor in the controller is in the wait condition and therefore the host computer has access to the memory. When the appropriate information has been loaded, the Host Adapter then writes a Hex 0 into the command register location at 3200. Writing a Hex 0 into this location causes an interrupt to the 6800 microprocessor which in turn causes the 6800 to read the information previously loaded into locations 32F0 to 32F3 to determine the type of operation to be performed and the target cylinder, head and sector information. The controller microprocessor then issues a header read command which causes the controller logic to read the next header from the currently selected cylinder and track. The cylinder address read

from the header is compared with the target cylinder address previously loaded into memory by the host computer. If a match is not found, a seek command is issued to the drive. The seek command is followed by another header read command to verify that the cylinder and head are now correct. Successive header read commands are issued until (target sector-1) is found. The next physical sector is read and stored in the RAM buffer. If 192 header read commands are issued (three disk revolutions) without finding the target sector, an error status is stored in the RAM buffer and a host interrupt is issued. After the RAM buffer is filled following a successful sequence, the CRC bytes are checked. If the CRC passes, the status is stored in the RAM buffer and a host interrupt is issued. If the CRC failed, the entire READ sequence as described above is repeated. If CRC fails a second time, a failure status is stored and a host interrupt is issued.

E. Write Operation

The write operation proceeds identically to the read operation with the following exceptions:

(1) The host loads 256 bytes of data into the RAM buffer in addition to the opcode, cylinder, head and sector bytes.

(2) After the (target sector-1) is found, the next physical sector is written instead of read. The write starts at detection of the sector pulse and consists of the preamble, sync, ID, data, CRC and postamble.

(3) CRC checking is not performed since data is not being read.

F. Format Operation

The host sets up the format command by loading the appropriate parameters into the RAM buffer. Loading a Hex 0 into the command register starts execution. Three types of format commands are available; format disk, format cylinder and format track. Each command begins with a seek to the appropriate starting place; cylinder 0, track

0 for format disk, track 0 of target cylinder for format cylinder and target cylinder and head for format track. The disk index pulse is monitored to find the beginning of the track and sector 0 is written consisting of preamble, sync, header, data, CRC and postamble. The "data pattern" byte loaded by the host will be written for each of the 256 data bytes.

Successive sectors are written by first performing a header read command for the (target sector-1). This results in one revolution of the disk per sector written. If more than one track is to be written, a seek is performed to each successive track and cylinder. When all tracks have been written, the status word is updated in the RAM buffer and a host interrupt is issued. The formatting command will be terminated before completion if any of the header read operations fails to find the (target sector-1) within three revolutions of the disk.

G. Copy Operation

The host sets up the copy command by loading the appropriate parameters into the RAM buffer. Loading a Hex 0 into the command register starts execution. Up to 256 contiguous sectors may be copied on to any similar sized space of contiguous sectors, with the restriction that neither space may wrap around from the innermost cylinder to the outermost cylinder. Attempting to do so will result in error status. The copy operation proceeds one sector at a time seeking to the source sector, reading and storing it in the RAM buffer space (32F0-33FF), seeking to the destination sector, and writing it from the RAM buffer space. The 16 byte header field is copied from the source sector except for the cylinder, head and sector bytes which are supplied by the controller.

H. Maintenance Operation

Each of these operations is set up by loading the appropriate test number into the RAM buffer and a Hex 0 into the command register.

1. RAM TEST Each byte of the RAM space is tested by walking a single ONE bit thru the 8 bits of the accumlator, copying to the RAM byte and comparing after each step. Any failure halts the test with error status; otherwise, the test ends after one pass with operation successful status.

2. ROM TEST This test performs a summing of all firmware bytes except for interrupt vectors and compares it with the known value stored at the end of the program space. Error or success status is stored.

3. SEQUENCE UP/SEQUENCE DOWN These maintenance commands allow the drive to be sequenced without using RESET or cycling power.

4. RANDOM SEEK CONTINUOUS/RANDOM SEEK STOP ON ERROR These maintenance commands cause sectors to be read at random from the entire drive storage space. In the continuous mode, any CRC error indicatons are ignored, while in the stop-on-error mode, the controller will cease the operation immediately on detection of a CRC error and issue a host interrupt.

5. VERIFY PACK This maintenance command reads each sector starting from the beginning of the pack. For each sector read that has an error associated with it, a 4 byte record is stored consisting of the error code, cylinder, head and sector address. At the completion of the command, these records are moved to the RAM buffer area. Operation is terminated when the end of the pack is reached or when thirty-two errors have been recorded.

Distribution: L. Holsopple ✓
C. House
J. Leslie
J. Paster
W. Strike