PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 4 : G06F 3/06	A1	(11) International Publication Number: WO 87/ 02154
		(43) International Publication Date: 9 April 1987 (09.04.87)

(21) International Application Number: PCT/US86/01665

(22) International Filing Date: 8 August 1986 (08.08.86)

(31) Priority Application Numbers: 780,762
780,864

(32) Priority Dates: 27 September 1985 (27.09.85)
27 September 1985 (27.09.85)

(33) Priority Country: US

(71) Applicant: BURROUGHS CORPORATION [US/US]; Burroughs Place, Detroit, MI 48232 (US).

(72) Inventor: COOGAN, Ronald, S. ; 32071 Pleasant Glen Road, Trabuco Canyon, CA 92678 (US).

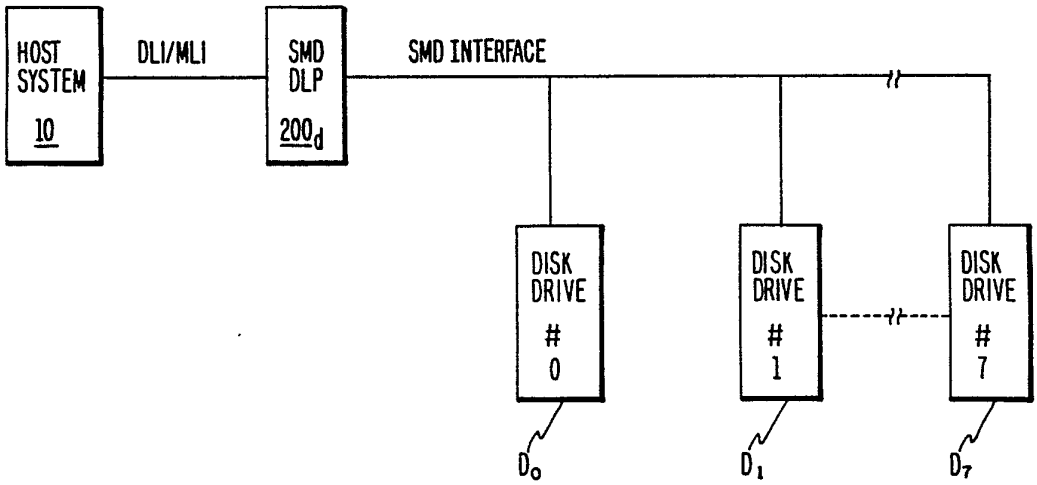
(74) Agent: PETERSON, Kevin, R.; Burroughs Corporation, Burroughs Place, Detroit, MI 48232 (US).

(81) Designated States: AT (European patent), BE (European patent), CH (European patent), DE (European patent), FR (European patent), GB (European patent), IT (European patent), JP, LU (European patent), NL (European patent), SE (European patent).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: SELF-TESTING PERIPHERAL CONTROLLER SYSTEM FOR MULTIPLE DISK DRIVE MODULES

Typical DLP Configuration.



(57) Abstract

A storage module device - data link processor (200_d, Fig. 1) manages data transfers between a main host computer (10) and up to eight disk drive units (D_0 - D_7, Fig. 1). The data link processor (200_d) provides a host access unit (Fig. 2) to the host computer (10) and connects to a Formatter Unit (Fig. 3) which establishes the protocol for accessing a particular sector of a selected disk drive unit and for transferring data to-from a selected disk drive (D_0 - D_7) via an Interface Circuit Unit (Fig. 4) which insures that only one particular disk drive is selected at a given time. The data link processor (200_d) initiates a self-test operation to verify integrity of the host access unit (Fig. 2) and the Formatter Unit (Fig. 3) before data transfer operations can take place.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	**FR**	France	**ML**	Mali
AU	Australia	**GA**	Gabon	**MR**	Mauritania
BB	Barbados	**GB**	United Kingdom	**MW**	Malawi
BE	Belgium	**HU**	Hungary	**NL**	Netherlands
BG	Bulgaria	**IT**	Italy	**NO**	Norway
BJ	Benin	**JP**	Japan	**RO**	Romania
BR	Brazil	**KP**	Democratic People's Republic of Korea	**SD**	Sudan
CF	Central African Republic			**SE**	Sweden
CG	Congo	**KR**	Republic of Korea	**SN**	Senegal
CH	Switzerland	**LI**	Liechtenstein	**SU**	Soviet Union
CM	Cameroon	**LK**	Sri Lanka	**TD**	Chad
DE	Germany, Federal Republic of	**LU**	Luxembourg	**TG**	Togo
DK	Denmark	**MC**	Monaco	**US**	United States of America
FI	Finland	**MG**	Madagascar		

SELF-TESTING PERIPHERAL CONTROLLER SYSTEM FOR MULTIPLE DISK DRIVE MODULES

FIELD OF THE INVENTION

This disclosure relates to peripheral controllers which are used to manage the transfer of data between a main computer system and a plurality of peripheral devices such as disk drive units.

BACKGROUND OF THE INVENTION

A continuing area of developing computer technology involves the transfer of data between a main host computer system and one or more peripheral terminal units. To this end, there have been developed

I/O subsystems which are used to relieve the monitoring and housekeeping problems of the main host computer and to assume the burden of controlling peripheral terminal units and to control the execution of data transfer operations which need to occur between the peripheral terminal units and the main host computer system.

A particular series of I/O subsystems has been developed which uses peripheral controllers known as "data link processors" whereby initiating commands from the main host computer are forwarded to the data link processor which manages the data transfer operations with one or more peripheral units. In these systems, the main host computer provides a "data link word" which identifies each task that has been initiated for the data link processor. After the completion of the given task, the data link processor will notify the main host computer system with a result-descriptor word to inform it as to the completion, incompletion, or problem involved in that particular task.

These types of data link processors or peripheral controllers have been described in a number of patents issued to the assignee of the present disclosure and these patents are included herein by reference as follows:

U. S. Patent 4,106,092, issued August 8, 1978 entitled "Interface System Providing Interfaces to Central Processing Unit and Modular Processor-Controllers for an Input Output Subsystem", inventor D. A. Millers, II.

U. S. Patent 4,074,352, issued February 4, 1978, entitled "Modular Block Unit for Input-Output Subsystem", inventors D. J. Cook and D. A. Millers, II.

U. S. Patent 4,162,520, issued July 24, 1979, entitled "Intelligent Input-Output Interface Control Unit for Input-Output Subsystem", inventors D. J. Cook and D. A. Millers, II.

U. S. Patent 4,189,769, issued February 19, 1980, entitled "Input-Output Subsystem for Digital Data Processing System", inventors D. J. Cook and D. A. Millers, II.

U. S. Patent 4,280,193, issued July 21, 1981, entitled "Data Link Processor for Magnetic Tape Data Transfer System", inventors K. W. Baun and J. G. Saunders.

U. S. Patent 4,313,162, issued January 26, 1982, entitled "I/O Subsystem Using Data Link Processors", inventors K. W. Baun and D. A. Millers, II.

U. S. Patent 4,390,964, issued June 28, 1983, entitled "Input-Output Subsystem Using Card Reader-Peripheral Controller", inventors J. F. Horky and R. J. Dockal. This patent discloses the use of a distribution card unit used to connect and disconnect the data link processor (peripheral controller) to/from a host computer as required to accommodate data transfer operations.

The above referenced patents, which are included herein by reference, provide a background and understanding of the use of the type of specialized peripheral-controllers known as "data link processors" (DLP) which are used in data transfer networks between a main host computer and various types of peripheral terminal units.

The present peripheral-controller called the storage module device-data link processor (SMD-DLP) provides features and solutions which were not available in earlier versions of similar data link processors.

For example, some peripheral-controllers (DLP's) required an "Emergency Request" cycle from DLP to the host computer to establish a communication channel for data transfer service when conditions of overload or underload occurred. This was especially so in systems using magnetic tape peripherals.

In the present situation, using disk drive modules, this need for an "Emergency Request" cycle is eliminated since the disk data can easily be accessed or not on the next turn of the disk and the amount of data in a sector is of a limited quantity.

Another feature in the present system is the use of "one logical address" to select an area of two physical sectors of data wherein the least significant bit (LSB) of the address field will select either the first or second physical sector.

To handle the situation of different characteristics of various disk drive units, an attribute table in a PROM is used to provide information to the SMD-DLP as to characteristics of the selected disk drive module. This PROM table informs the DLP as to sector numbers, beginning addresses, ending addresses, number of track-heads available in a particular disk unit, and the number of bytes per track in each unit. Thus, rapid and accurate sector location and data transfer operations can occur.

Since two sets of interface unit cards are used to communicate with two groups of disk drives (each group handling four (4) disk drive modules), the selection of one interface card is accomplished through a unit select logic circuit after verification that only one disk unit has properly responded for use in data transfer operations.

The handling of up to eight disk drives by one data link processor (DLP) is facilitated by a queue file section in buffer memory which can store up to eight I/O commands for later execution. However, the system also provides for immediate execution of an I/O command even while commands remain stored in the queue file.

These and other features are provided by the described data link processor for multiple disk drive modules.

SUMMARY OF THE INVENTION

The present disclosure involves a data transfer network wherein a data link processor is used to manage and control data transfers between a main host computer and a multiplicity of disk drive storage units.

The storage module device-data link processor described herein provides for data communication between a main host computer and up to eight separate disk drive units. The data link processor includes three functional sections known as the host adapter unit, the formatter unit and the peripheral interface unit which, in combination, function to monitor and control data transfer operations between any one selected disk drive unit and the main host computer system.

The host adapter unit receives the I/O command descriptors from the host system, verifies their correctness and checks their parity, and provides a queue list in a buffer memory of the host adapter, which indicates the jobs or tasks to be accomplished with each of the eight disk drive units.

The command task information is processed and conveyed to the formatter unit which formats information from an attribute table in the formatter in order to provide address data as to the particular cylinder, track-head, and sector to be accessed in a selected disk drive unit. This information is passed on through the peripheral interface card which communicates directly to the selected disk drive unit.

In order to prevent useless and time-wasting operations, it is advantageous to check the operational integrity of the peripheral-controller hardware before actual execution of command operators.

The present system, upon "power-on", or upon pressing a manual self-test pushbutton, or upon a self-test commmand from the host computer, will initiate self testing

of the host adapter card and the formatter card. A light emitting diode (LED) on each card will remain lit up should a misfunction be indicated. The LED will turn off if normal operation is satisfied. On self-test initiation by the host computer when a misfunction is indicated, then the peripheral-controller will not reconnect to the host computer, thus preventing further I/O operations.

Unit select logic in the peripheral interface card insures that only one selected disk drive unit will be communicated with at any given time.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a network drawing showing how the disclosed data link processor interfaces the host system to a plurality of disk drive units;

FIG. 2 is a block diagram of the section of the data link processor called the host adapter unit;

FIG. 3 is a block diagram of the section of the data link processor called the formatter card unit;

FIG. 4 is a block diagram of the interface card unit of the data link processor;

FIG. 5 is a schematic diagram of the three sub-units of the data link processor and their cable connections to a plurality of disk drive units;

FIG. 6 is a schematic diagram of the data buffer RAM of the host adapter unit of the data link processor;

FIG. 7 is a schematic layout drawing showing the format that is used for track/sector control in the disk drive unit of the network;

FIG. 8 is a drawing showing a layout of the bit locations for the cylinder address, track address, and sector address.

Referring to FIG. 1, there is seen a data transfer network system, whereby the main host computer designated as host system 10 communicates over a set of buses designated as DLI/MLI (data link interface-message

interface) to a peripheral controller herein called a data link processor and specifically a storage module device-data link processor (SMD-DLP) 200_d. The output the data link processor includes an interface which has bus connections to a series of disk drives D_0 through D_7. Thus, the network provides for data transfer operations between eight disk drive units and the host system 10 by means of the peripheral-controller 200_d.

As will be seen in FIG. 5, the storage module device-data link processor (SMD-DLP) is made up of three separate units which are seen as the host adapter unit 2, the formatter unit 3 and the interface unit 4 which may be formed of first and second interface boards 4_a and 4_b.

A series of disk drives designated as D_0 through D_7 are shown connected to the data link processor through different sets of cables.

Referring to FIG. 5, the A cable may connect the formatter 3 to each and everyone of the disk drive units D_0 through D_7 on a daisy chain connection basis.

Likewise, the first interface 4_a may use the B_x cable to connect to disk drives D_0 through D_3. Then the cable designated as the B_y cable can be used to connect the second interface 4_b over to the disk drives D_4 through D_7.

An "M bus" (FIGS. 2, 3) is used to connect the host adapter 2 to the formatter card unit 3. Then a cable B' (FIG. 5) is used to connect the formatter 3 to the interface card unit 4_a and to the second interface unit 4_b.

Host Adapter Card Unit 2

Shown in FIG. 2 is a block diagram of the host adapter unit which connects the host computer system 10 for the data link processor 200_d.

As seen in FIG. 2, a transceiver 20 connects to the host system 10 by means of a data link interface bus designated DATAAx. The transreceiver 20 connects to an I/O bus 20_s which has a terminal transceiver 70. The

transceiver 70 connects to an M bus designated DATxx which connects to the formatter card unit 3.

A RAM buffer 22 is connected to the I/O bus 20_s in order to temporarily hold data words which are being transferred between the host system and the disk systems. A more detailed exposition of the buffer RAM 22 will be later described in connection with FIG. 6.

The general purpose of the data link processor 200_d, as effectuated by its three sub-unit cards, is to select one of the eight disk drive units, then to address certain areas of the selected disk and execute a data transfer routine either in the read direction or in the write direct-on and at the same time to concurrently assure that data integrity is maintained so that error free data transfers will occur in all instances.

The basic purpose of the host adapter card unit 2 (FIG. 2) is to receive I/O descriptor commands from the host system 10 and verify the command to make sure that it is correct, including the making of parity checks. The host adapter card unit 2 will place the I/O descriptor command from the host system into its RAM buffer 22 into the queue area (FIG. 6) starting at the hex address F00. Thus, I/O descriptor commands are placed into eight separate areas, each of which involves a command function to be executed with respect to each of the eight disk drives connected to the system. Thus, the queue area of the buffer memory 22 shown in FIG. 6 can store up to eight basic data transfer commands which will subsequently be executed by the data link processor 200_d.

The commands for operators which are stored in the queue file blocks are designated as "queueable OPs". However, there is also provision for an "immediate OP", whereby the execution of a command may be effectuated without going through the queue of the buffer memory 22.

In operation, the data link processor will do a "seek" function, whereby an operator OP will be taken from the queue area and placed into the area known as the device control block or DCB. As seen in FIG. 6, there are eight DCB areas or device control block areas designated as DCB0 up to DCB7. These occupy the hex addresses from E00 up to the beginning of the queue area at hex F00. Each of the device control blocks consists of 32 words which are sent to the formatter 3 which will then develop the track/sector format which is illustrated in FIG. 7.

The OP information in the DCB of buffer 22 that is selected will be used to "wake up" the formatter into action.

Referring to FIG. 2 and the host adapter card unit 2, a series of input signals are fed to the sequencer 26 to select microcode from the host control store 30. The host control store 30 is the main engine for controlling and operating the data link processor 200_d. The sequencer 26 receives various data signals which are multiplexed into the host control store 30 in order to select routines which permit decisions for which branch operations are to be performed by the microcode of the control store 30.

The sequencer 26 receives signals from the OP register 46 and OP decoder 48, burst carry signals from the burst counter 50, "command received" signals from the line 33 to the multiplexor test-input 24 (T-Way) and also from the PROM scan 28, in addition to signals from the control store 30 which are fed back on lines 24 and 25.

The status input signals to the test-input multiplexor 24 involves a multiple number of test points connected throughout the host adapter unit 2. These test points include connections for buffer memory management control, ALU status, counter status (of elements 50, 52, 54), DLI interface control, M-bus monitoring and handshaking, and status of operator (OP) types.

The stack counter 60 counts the locations in the RAM buffer 22 that point to branch addresses in the control store 30.

The host access counter 62 is a counter for the RAM buffer 22 which is used to count data words transferred to/or from the host system and the buffer 22.

The peripheral counter 64 is a counter used to count the number of data words transferred between the RAM buffer 22 and a selected one of the disk file units of FIG. 1.

The word counter 52 is used as a "block counter" which will count up a specific number of words such as 256 words in order to signify the counting of a block of data. This counter provides an output "CNTZ" which is fed to the control store 30 via MUX 24.

The address multiplexor 66 receives input data from the host counter and the peripheral counter, in addition to address data from the formatter card unit 3. This count data is fed to the RAM buffer 22 in order to address various locations in the RAM buffer 22.

The peripheral counter 64 is operated so as to count only up to number 4,000 which is equivalent to the address DFF (hex) which can be seen in FIG. 6 in the data buffer section which holds 16 x 256 words (with parity). The section designated DB1 through DB14 represents 14 blocks of words, each having 256 data words in it. At the address DFF, the address pointer will return to the first block at 000.

The burst counter 50 is used to count blocks of data words (each block is 256 data words) which provides a carry signal BCCRY which is sent to the host control store 30 to indicate that a block of data has been sent to the host 10.

The time-out counter 54 is used to put a time-out data signal to the control store 30 if certain conditions do not occur by a preset time.

The OP register 46 and the OP decoder 48 will receive an operator signal (OP) from the control store 30. The OP register 46 provides an address to the OP decoder 48 which is a PROM which will provide signal data to sequencer 26 for the control store 30.

The arithmetic logic units 42 receive microcode signals from the control store 30 which can be used to process data such as performing various logical operators and/or shifting and manipulating data which can then be returned back to the I/O bus 20_S. This processed data can be sent onto the M bus of FIG. 2.

The D latch 44 is a pipeline register which can receive data from a disk drive unit and formulate a longitudinal parity word for error checking purposes. Thus, data from a selected disk file will be transferred to the buffer RAM 22 and then placed in the D latch 44, after which it can be transferred to the transceiver 20 for transmission to the host 10 which can then verify the longitudinal parity word.

The PROM scan unit 28 is used to tri-state the sequencer 26 and acts as a downcounter in order to deliver address data for the control store 30, so that PROM parity check unit 35 can verify the correctness of data from control store 30.

The circuit block designated PROM parity check 35 is used as a parity checker to verify odd parity.

The constant register 32 is used to provide a data field for the control store 30 so as to provide a basic data pattern for initialization operations.

The secondary pipeline register 34 is used for interface purposes to the host system 10 to provide handshake operations. This provides extra control in conjunction with the control store 30.

The peripheral control store 38 is used to control the M bus and the interface to the formatter card unit 3. This is done through a gate 40. A sequencer 36 receives various input data in order to select microcode routines in the control store 38. The sequencer 36 receives data from the slave case 37 on the line OPx and also receives data from the control store 30 on line S_x in addition to feedback data on line 41 and select signals from receiver 39.

A series of driver gates 3, FIG. 2, will be seen to connect to the formatter card unit 3 in order to provide bi-directional transfers. The gates 67 and 68 connect the address lines to and from the formatter unit 3. The gates 39 and 40 connect the control line data between the formatter unit 3 and the host adapter.

The address lines ADRXX in FIG. 2 re the "wake up" lines which are used to activate the system to read the selected data control block (DCB_x) in the buffer memory 22 of FIG. 6.

In FIG. 2, the transceiver 70 connects the M bus between the formatter unit 3 and the host adapter 2. This provides a bi-directional connection for data words to be transferred between the two units.

The multiplexor test signal unit 24, designated "T-Way" can be considered as a "test input" unit, whereby certain status data inputs are multiplexed into the sequencer 26, as were previously enumerated.

The pipeline register 44, designated as D latch, is a data register used to store input data to ALU 42 (FIG. 2). This unit type is described in a publication entitled "Bipolar Microprocessor Logic Interface" at page 5-166, and published by the Advanced Microdevices, Inc. of 901 Thompson Place, Sunnyvale, Ca. 94088.

It should be noted that the data link processor 200_d is provided with self-test routines which allow

internal operation of the various card units to be checked before actual data word transfer operations are initiated.

The slave case 37 is a PROM unit which is used for restricting any invalid operators which come from the formatter card unit 3. Thus, if any invalid command and address data from the formatter 3 are received, the slave unit 37 will operate to prevent any execution and to inform the host of such an invalid operator, via the result/description.

The Formatter Card Unit 3

The formatter card unit has the purpose of selecting a disk drive to be communicated to and providing data whereby selected areas of the particularly selected disk drive may be accessed for either Read operations or for Write operations.

As seen in FIG. 3 data words are transferred between the formatter 3 and the host adapter 2 by means of a transreceiver 70.

Address data between the host adapter and the formatter is transferred through the transreceiver 70, which is controlled by the M bus controller 76 which receives control signals through the control line monitored by gates 74 and 75. The M bus controller 76 can also transmit information through gate 74 to the control line of FIG. 3.

Similarly to the host adapter card unit, the formatter card 3 has a control store 90 which receives input addresses from a microprogram controller sequencer 88. The sequencer 88 receives various data from test circuitry (82, 84, 86), from the control store 90, and from masking circuitry 77, 78 and 79. The vector RAM 89 is a counter similar to the stack counter 60 of FIG. 2 which provides address data to the control store 90.

In FIG. 3, a set of arithmetic logic units 80 are provided to the data bus in order that processing and data

manipulation can occur for any data words on the data bus line.

The mask circuitry is composed of a mask register 77, a mask unit 78, and an encoder 79, which are used to eliminate unnecessary data and to pass required data to the vector RAM 89 to develop addresses for control store 90.

The test circuitry (82, 84, 86) receives 16 bits of data from the formatter RAM buffer 22_2 in order to perform status tests and provide signal data to the sequencer 88 for the control store 90.

The constant multiplexors 92 and 94 are used as a repository of reset and intialization data patterns which are used to reset and initialize various circuitry after clearing operations.

An address register 96 is used to provide addresses both to the formatter buffer RAM 22_2 and also to an attribute table 97. The attribute table 97 is a specific table of data which provides information as to sector numbers, beginning addresses, end addresses, the number of heads per drive and the number of bytes per track for each of the eight separate disk drive units.

The formatter buffer RAM 22_2 is used as a data buffer to temporarily hold data words being transferred from the selected disk drive over to the host access adapter 2 card unit.

The unit, designated as sector format 9, has 84 sectors and 42 records to provide a pattern of zeros as required for the sector format protocol. It provides the pattern timing necessary for the formatting of a record containing two (2) sectors.

The micro-decoder 99 is used to decode the microcode received from the control store 90, and to select the sector format from unit 98, and to provide control data to M bus control circuit 76.

The register 100 is a first-in first-out register (FIFO) which is used for serial to parallel translation or from parallel to serial translation so that serial data received from the disk unit is translated into parallel form for transmission to the host and vice versa, whereby parallel data from the host is transferred into serial data for transmission to the selected disk drive.

The shift register 101 is used to transmit read data (which comes from the interface 4 through gate 112, gate 113 and register 101) into the FIFO 100 which data is then placed on the data bus. The unit 102 is a gate which feeds data to the cyclic redundancy checker and error correction code unit 103. Here, every data field is checked for accuracy and corrected by an error correction code if necessary.

Gate 104, flip-flop 110 and gate 111 are for retransmission of Write data from the FIFO 100 over to the interface 4 for transmission to the selected disk drive unit.

Gate 105 is for enabling the "unit select" (identification) signal from interface 4 into the formatter 3.

Register 106 and gate 107 permit data words to be transferred from the formatter data bus on the A cable (FIG. 5) to a selected disk drive as commands to be performed.

Register 108 and driver gate 109 permit data from the microcode decoder 99 to be transferred to interface unit 4 (FIG. 3).

Gate 104 and flip-flop 110 control the synchronization of data transfers by using the write clock signal from the disk drive.

Gate 109 is designated as driver (DRV) tags and will provide tag information data to the disk drive.

Format for Disk Drive Units: The disk packs which are used

with the storage module device data link processor can be of the type which contain four and six platters respectively. The bottom side of the lowest disk is pre-written at the factory with reference servo patterns. Each pack is indexed from a signal derived from the servo track which denotes the beginning of a track. Data is stored in a 180 bytes per sector format. Sector addresses are stored on the pack in binary notation through the controller by means of the initialization function. The drive is preset in a soft-sectored mode whch means that sector pulses are derived from variable byte counters synchronized to an index.

Each record is divided into three areas: (i) the control address area and (ii) the two data areas.

The control area is used to:

(1) Establish bit sync and character frame for the sector address;

(2) Contain the sector address;

(3) Contain delays (gaps) for head switching;

(4) Establish bit sync and character frame for the data;

(5) Contain error codes for address error detection.

The two data areas (ii) each contain the user data storage area on the track. All data areas are separated by control areas.

There are N spare records per cylinder located at the end of each cylinder, i.e., the last five records on the head with the highest number. The value of N, which can be different for each drive type, is specified in the device attribute table 97 which was described in connection with FIG. 3 on the formatter card unit. For example, in one type of disk drive the value of N is equal to "five" spare records.

The last cylinder of each unit is reserved for drive maintenance. The beginning and the ending sector addresses of the maintenance area are provided in the device attribute table 97.

Format Description: The disk pack will use the track/sector format which is shown in FIG. 7. A track is divided into 42 records (physical sectors), each record containing two parts of 180 bytes (logical) sectors. Each record consists of a beginning-of-record (BOR) gap, a header, two sectors (each with a beginning of sector gap) and an end-of-record (EOR) gap.

The Header (FIG. 7): The header consists of: an address sync byte (19 HEX), and address field. The address field is further sub-divided into: (i) a flag field, (ii) a cylinder address field, (iii) a track address field (iv) a sector address field, and (v) an address-cyclic redundancy check (CRC) field. A four bit flag field contains a flag bit designated "bad sector", and also contains a flag bit designated "relocated sector", in addition to two reserved bits (which are always 0).

Sector (FIG. 7): Each sector consists of:

A beginning-of-sector (BOS) gap;

A data sync byte (19 HEX); this is a "binary" 0001 1001;

A 180 byte data area;

A data error correction code (ECC) field.

The "gaps" contain all zeros. The size of the end-of-record (EOR) gap is different for the last sector of each track.

As seen in FIG. 7, each field can be defined as follows:

(1) GAP 1: 16 bytes -- this is required to allow for head selection (5 microseconds minimum).

(2) VFO Synch: 11 bytes -- this contains all zeros. This allows the variable frequency oscillator (VFO)

in the disk drive to synchronize to new data transitions and the speed of the disk. It is contained in the BOR and also in the GAP 2.

(3) ADDRESS SYNC: 1 byte -- this identifies the beginning of the address field and synchronizes the byte counters in the controller. It is a Hex 19 pattern.

(4) ADDRESS FIELD: 6 bytes -- this consists of a four bit flag field, a 12 bit cylinder address field, an eight bit track or head address, then eight bits (1 byte) of sector address and two bytes of CRC (cyclic redundancy check) error codes for address error detection.

(5) GAP 2: 19 bytes -- this contains all zeros. This includes one byte for WRITE splice which allows time for the WRITE driver turn-on and 11 bytes minimum for the VFO sync. The rest is required by controller delays.

(6) DATA SYNC: 1 byte -- this identifies the beginning of the data field and synchronizes the byte counters in the controller. This is a Hex 19 pattern.

(7) Data Field: 180 bytes -- this is the user data field.

(8) Data ECC Field: 4 bytes -- the data error detection and correction code. ECC is encoded by the correction controller 103 for each 180 bytes of data field and is subsequently decoded.

(9) GAP 3: -- this constitutes 46 bytes for records 0-40, and constitutes 30 bytes for record 41. This EOR (end-of-record) gap contains all zeros at the beginning and then transits to the WRITE gate turn-off. There must be a minimum of four bytes of zeros to prevent WRITE turn-off transients from causing errors, and there must be a byte (8 bits) of zeros for delay times due to WRITE encoding/READ decoding. This may be juggled with GAP 1 depending on the pulse timing of the WRITE gate to sector pulse timing.

As seen in FIG. 8, the cylinder address uses 1.5 bytes (12 bits) of data bits. The track address uses one byte of eight bits of data while the sector address uses eight bits (1 byte) of data. The least significant bit (LSB) in the sector address (which is not contained in the field written on the disk platter) can be a 0 or 1 in order to select either the first or the second physical sector of data located by the single sector address SSSSSSS0.

Interface Card Unit

Referring to FIG. 4, there is seen the interface card unit 4. Each card unit 4_a and 4_b follows the pattern of FIG. 4.

Connections to four disk drive units (D_0, D_1, D_2, D_3) are shown with data transfer line connections for the Write operation (designated as WT) by passage through drivers D. Likewise, data transfer operations on a Read operation are effectuated through the receiver gates, designated R. Each one of the disk drive units also has unit select lines which pass through the receivers marked R_S.

The major purpose of the interface card 4 is to receive the unit select signal (USS) of a given disk drive unit for connection to the formatter card 3 so that appropriate data transfer operations, either Read or Write, can be effectuated.

The unit select signal can come from any one in a group of eight signal lines.

The unit select signal (USS) is originated from two sources:

(i) a disk drive unit;

(ii) a single one of a group of jumpers.

The unit select signal is transmitted via driver 105 (FIG. 3) to unit select logic in 115. The select logic in 115 then decides which "function" is requested, that is, if it is a Read unit type (jumpers) or Read unit type (disk drive).

The unit type function uses all of the eight USS lines (0-7) while the unit select function uses only a selected one line of the eight lines.

The jumper information tells the formatter unit 3 what "type" of disk drive is connected, that is, to say, what manufacturer type it is and thus its cylinder, track and sector capacity as seen, for example, in FIG. 7 which shows two types of Memorex disks, the 214 and the 226.

From the jumper identification the formatter 3 can then address the attribute table 97, FIG. 3, to access specific information regarding the characteristics usable for that specific type of disk drive.

Thus, any data words received from a selected drive unit will be transferred through the receiver gate on the data line marked RD and passed onto the formatter 3, and into buffer 22_2.

Similarly, Write data on data line WR can be received from buffer 22_2 in the formatter 3 and passed onto the gate drivers D for transit to the selected drive unit.

Driver gates 117, 119, 121 and 123 are made to work in conjunction with jumpers to provide signals to the unit select line which goes to formatter 3. The jumpers operate to identify a disk drive unit as to its type.

Driver gates 116, 118, 120 and 122 are used to convey the unit select signals to the formatter 3. The unit select logic 115 also receives signals from the PROM 97 and the unit select gate R_S in order to send a signal to the formatter 3 to indicate that only one unit has been selected.

The driver gates 125, 127, 129 and 131 are used to enable any Read operations on the RD data line. These gates are controlled by 115.

The power detect logic unit 150 provides a channel ready signal to sense if there is a power supply problem in any of the selected disk units. If a power supply problem is detected in a disk module, then that particular disk interface will be shut off.

There is also a clock line from the disk drive to formatter 3 as indicated in FIG. 4. This line is called the WR drive clock line and is used to provide clock operations during the writing and reading of data of the disk that is selected.

The unit select logic 115 (FIG. 4) receives inputs from each disk drive module (D_0, D_1, D_2 .... D_7) and operates to scan these inputs to see whether only one disk module has responded (error). If two or more disk modules have responded, then logic 115 will output an error signal to the formatter unit 3. The formatter will report the error (multiple response) to the host adapter unit 2 which will signal host computer 10. This will initiate a retry of the operator (OP) which caused the error signal.

The logic circuit 115 is a programmed array logic (PAL) unit of which a typical version is manufactured by Monolithic Memories Inc. of 2175 Mission College Blvd., Santa Clara, California 95050. This unit is described as a 20X4 PAL in the 1983 publication entitled PAL-Programmable Logic Handbook - 3rd Edition and published by Monolithic Memories, Inc.

The power detect logic 150 of FIG. 4 is used to monitor the voltage levels of the host 10 (host adapter 2, formatter 3 and interfaces 4 (4_a, 4_b). Should the power supply to any of these units fall below a specific voltage, then the power detect unit 150 will command a "turn-off" of the disk drive interface connections to the disk modules.

As is indicated in the patents incorporated by reference, the host system 10 sends the DLP an I/O descriptor (command) and a descriptor link which identifies that particular command cycle. The DLP performs the required operation and returns a result/descriptor (R/D) to

the host to indicate the status of that command (complete, incomplete, error, etc.).

The result/descriptor fields provide all information needed by the host 10 for a "retry" in the event of unsuccessful completion of the I/O descriptor command.

A result/descriptor is used to immediately report (to host 10) any "exception" conditions such as:

(a) a vertical or a longitudinal parity error in the I/O descriptor or the descriptor link;

(b) an invalid unit number (for a peripheral) is detected;

(c) the command queue (in RAM 22) for the designated peripheral unit is already full;

(d) Read or Write errors.

The host does not disconnected (from DLP), process or send other command operators until the DLP accepts present OPs. A command operator having an error condition will not be executed or placed in the memory queue.

If there are deficiencies in the magnetic layer areas of a disk which will cause "media errors", this will be reflected in "exception bits" in the result descriptor words. These exception bits identify the location of the defective sector.

Error Recovery:

The data link processor will do an automatic error recovery operation for:

(i) Seek errors -- where difficulty occurs in locating the desired cylinder and track;

(ii) Read errors - due to a bad address or bad (incorrect) data.

On seek errors the DLP does a retry by repositioning to cylinder "0" and then retrying the seek to the desired cylinder and track.

On Read errors which may show up as "Bad Address" data (BA) or Bad Data (BD), the DLP will retry the Read for nine retries using track offsets which position the head at nominal center, then an inward offset and outward offset until a correct address and data is effected. If data is still incorrect, the formatter will attempt to correct the data by using the ECC as a pointer.

The DLP uses an operator called the Verify Data/ EPC OP. This command causes the DLP to check for Read errors. The DLP reads the sector specified by the logical address and continues reading until (i) a Read error or (ii) a media error or (iii) end of cylinder is encountered.

For each sector the DLP verifies its position relative to the index (FIG. 7), address and data EPC (error protection code) fields and data field.

The data field contents are compared to a 16-bit test pattern which was transferred to the DLP as Write-Data. Thus, "correct" readout or "error" readout is spotted.

The storage module device-data link processor, as previously indicated, operates under host generated commands called I/O descriptors. A typical I/O descriptor format is shown in Table I as having five words. Each word has four area positions shown as 4-bit group A, group B, group C and D. The name and functions of each field are described in Table I, shown hereinbelow.

Table I. Common I/O Descriptor Fields

Bit Designations	A8	A4	A2	A1	B8	B4	B2	B1	C8	C4	C2	C1	D8	D4	D2	D1
I/O Descriptor Format:																
Word 1:	<---OP---->				<-Sub-OP-->				<---Unit-->				<-Variant->			
Word 2:	0	0	0	0	0	0	0	0	<---Storage Address----							
Word 3:	----------Storage Address---------->															
Word 4:	0	0	0	0	0	0	0	0	<-Data Transfer Length-							
Word 5:	----------Data Transfer Length---------->															

OP Field — Specifies the nature of the data transfer that occurs as part of the successful execution of the operation. The legal codes are as follows:

A8	A4	A2	A1	Definition
0	0	1	0	TEST OP - no data transfer.
0	1	0	0	WRITE OP - data transfer from host to DLP.
1	0	0	0	READ OP - data transfer from DLP to host.

Sub-OP Field — Specifies the type of operation to be performed by the DLP. The actual values for each OP are shown in the section on individual OP types.

Unit Field — Designates the Unit within the subsystem to which the operation is directed. The range of values is 0 to 7.

Variant Field — Specifies options for a given Sub-OP. The DLP ignores any part of the variant which is not defined for a given OP.

Address Field — Specifies either the logical (sector) address or the physical (sector) address within the specified unit, depending on the OP.

Length Field — Specifies the length of the data to be transferred, in bytes, for a given OP.

In order to select a particular disk drive module (designated 0 through 7), each command operator (OP) will designate the particular disk module to be used. The format OP (I/O descriptor) is used as seen typically in Table II. Basically three words will specify the drive unit number, the cylinder address, the head (track) address and the desired sector address. These are indicated in the locations shown in Table II hereinbelow.

Table II Format OP (I/O Descriptor)

Bit Designations	A8	A4	A2	A1	B8	B4	B2	B1	C8	C4	C2	C1	D8	D4	D2	D1
I/O Descriptor Format:																
Word 1:	0	0	1	0	1	1	0	1	0	UN	UN	UN	0	0	0	0
Word 2:	0	0	0	0	0	0	CA	CA	CA	CA	CA	CA	CA	CA	CA	CA
Word 3:	0	0	HA	HA	HA	HA	HA	HA	0	SA	SA	SA	SA	SA	SA	SA

Word 1 I/O Descriptor Abbreviations:

UN - Drive Unit Number (0-7 hex)

Word 2 I/O Descriptor Abbreviations:

CA - Cylinder Address

Word 3 I/O Descriptor Abbreviations:

HA - Head Address

SA - Sector Address

There are several maintenance techniques for testing the data link processor. One method is by the host computer to run maintenance and confidence tests by its master control program (MCP) which can detect and isolate one particular unit of the system.

The other method is the DLP self-test which is an inbuilt diagnostic routine which can diagnose any DLP malfunctions and isolate them to the failing circuit board or module on it.

The DLP self-test does not test the disk drive peripherals nor the DLL interface to the host computer 10. The self-test operates to test the hardware of the logic modules of the DLP. Table III indicates a self-test initiate OP using one word (16) bit and the result/ descriptor format of four words which are returned to the host 10 for information.

Table III Self Test Initiate OP

Bit Designations	A8	A4	A2	A1	B3	B4	B2	B1	C8	C4	C2	C1	D8	D4	D2	D1
I/O Descriptor Format:																
Word 1:	0	0	1	0	1	0	1	1	0	0	0	0	0	0	0	0
Result Descriptor Format:																
Word 1:	0	UI	VP	LP	0	0	0	ID	0	0	0	0	0	0	0	EX
Word 2:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Word 3:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Word 4:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Word 1 Result Descriptor Abbreviations:

A8 - 0 - Not Used	C8 - 0 - Not Used
A4 - UI - Unable to Initiate	C4 - 0 - Not Used
A2 - VP - MLI Vert. Parity Error	C2 - 0 - Not Used
A1 - LP - MLI Long. Parity Error	C1 - 0 - Not Used
B8 - 0 - Not Used	D8 - 0 - Not Used
B4 - 0 - Not Used	D4 - 0 - Not Used
B2 - 0 - Not Used	D2 - 0 - Not Used
B1 - ID - Invalid I/O Descriptor	D1 - EX - Exception

Self-Test Initiation: The following four methods are provided by the SMD DLP to initiate a self-test. These methods are compatible with DLPs in any base.

1. An automatic self-test is performed when the base is powered on. This is a complete hardware test.
2. A self-test is initiated manually by a pushbutton. This is a complete hardware test.
3. A self-test is initiated by test operation on the DLI test bus. This operation can be initiated by the maintenance card or by system hardware that has access to the DLI. This can initiate a complete self-test or a subset of the self-test.
4. A self-test of the DLP may be performed by sending a host-initiated I/O descriptor to the DLP. This descriptor is sent by a miantenance program such as PTD (peripheral test driver routine). In addition, tests of subsystems or modules may also be performed by I/O descriptor.

The DLP begins its self-test upon receipt of any three clear signals: a foreplane clear, which is generated by a pushbutton switch located on both the host adapter card and the formatter card, a powerup clear, and an initiation signal from the test bus.

After initiation of the self-test, the DLP disables its peripheral and DLI interfaces. If the DLP is addressed when executing its self-test, it appears at status zero (non-present DLP). The DLP remains at status zero until successful completion of its self-test.

Manual Self-Test Initiation: The host adapter card 2 and formatter card 3 each contain a pushbutton switch and an LED used for the manual self-test initiation and self-test result reporting.

Self-Test Initiation While MCP is Executing: (MCP = master control program in host 10)

If the self-test is initiated while the MCP is executing, all paths from all hosts to DLP(s) on which self-tests are to be initiated must be reserved prior to the self-test. This is required to ensure the integrity of the base and MCP during self-test.

While the self-test is executing, DLI state is not disturbed. If the DLP is addressed while self-test is executing, a status of zero is seen by the host due to the status lines being disabled.

Self-Test Initiation by I/O Descriptor: Initiation of self-test by I/O descriptor is done using PTD (peripheral test driver routine). The initiation of self-test by I/O descriptor does not provide the single module failure resolution that self-test initiated on the test bus lines does. This lower level of resolution when initiating tests by I/O descriptor is due to the requirement that the HDP, MLI, connection modules, and DLI must be functional to access the DLP.

The HDP is the host dependent port of the host system 10 in FIG. 1 while MLI refers to the message level interface and DLI refers to the data link interface (FIG. 1).

The self-test status I/O informs the host if the formatter card fails, but the host adapter card passes the self-test.

In order to prevent useless and time-wasting operations in a peripheral-controller, it is advantageous to check the operational integrity of the hardware before actual execution of operators.

The presently described data link processor (peripheral-controller) includes a host adapter card and formatter card which undergoes a self-test upon power-on, or by a manual pushbutton on each card, or by a host initiated self-test command (I/O descriptor).

Upon power-up, or manual self-test, a host control store (30) will be sequenced to probe test the hardware in host adapter card 2, FIG. 2, after which sequencer (26) will receive signals which indicate proper function or misfunction. Sequencer (26), upon misfunction signals, will cause a control store 30 to light up (via circuit 34, FIG. 2) a light emitting diode, 20_d, which will continue to be lit if the host adapter card 2 does not operate properly.

Likewise, formatter card unit 3, FIG. 3, will have its control store (90) sequenced by sequencer (88) to probe test the hardware components afater which sequencer 88 will receive signals to indicate proper function or misfunction of card unit 3. If misfunction occurs, the sequencer 88 will cause control store 90, FIG. 3, to light up the light emitting diode 20_f via bus control 76. If no misfunction occurs during self-test, then control store 90 will disable (turn-off) the LED 20_f via bus control 76.

The host and peripheral connections to the DLP (data link processor) are disabled during the self-test operation.

Another self-test operation can be initiated by the host 10 via an I/O command descriptor to initiate the self-test routines.

Here the host 10 is connected to the DLP and transmits a self-test initiate command which is verified by a result/descriptor word to the host 10. Then the host disconnects from the DLP while the self-test operations continue in the host adapter card 2 and the formatter card 3. If no misfunction is sensed, the DLP will reconnect

to the host for further operation. If a misfunction is sensed, the DLP will not reconnect to the host and the misfunctioning card unit will maintain the "lighted" condition of its light-emitting diode.

Self-Test Result Reporting: The SMD DLP provides the following methods to report the result of the self-test. All methods are compatiable with any I/O base module which can house a multiple number of data link processors.

1. The result is displayed on LEDs which are visible outside of card air shields.
2. The result is returned on the test bus lines.
3. The self-test status I/O descriptor returns the result of the self-test.

The result bits have the same format and information for all methods in obtaining the result of self-test. The LEDs, display bits (DPLY01-02) and self-test status result/descriptor word two bits have identical meanings. When the DLP is either in the process of self-test, has failed the self-test, or has detected an on-line failure, the LEDs are on, the test bus display lines are active LOW and the self-test status result/ descriptor word 2 bits are TRUE. The primary bit, top LED, and DPLY01 display the status of the self-test in relation to the main logic board. If the LED is on after the specified time for execution of the self-test, it indicates that the board failed. If both the primary bit and other bits are TRUE after the specified time for the test, it indicates that no other boards were tested and that the main logic board has failed its self-test or that the connection to the main board is bad.

When the test is initiated, the primary bit is set to TRUE, the LEDs are turned on, DPLY01 is HIGH, and D1 is TRUE. If the test completes successfully, the result bits are all FALSE. If the bit for the primary board is FALSE, and one or more bits remain TRUE, the bit(s) which is TRUE

indicates which module is failing. The maintenance flow for the individual system can be followed to determine the order of module replacement.

The self-test status opration states the correspondence between result bits and boards, or modules, contained in the DLP.

Host-initiated self-test results may be obtained for maintenance purposes using the self-test status operation. This is accomplished by using a program such as PTD (peripheral test driver). If the DLP is not sufficiently operational to return a result, a status of zero is presented when connection is attempted. A status of zero is interpreted as a non-present DLP by the host.

When the DLP is addressed by host 10, then a line designated LCPON/O is active; therefore it is possible to differentiate between a "non-present" DLP, and one that is in the process of self-test or has failed self-test. (When non-present DLPs of either type are addressed, LCPON/O is not active). The LCPON/ signal is used to indicate whether the host is connected or not connected to the DLP.

Systems Operation:

Referring to FIG. 1 and FIG. 2, the host adapter card 2 receives requests for disk operations from the host system 10 by means of the MLI/DLI (message level interface-data link interface). Thus, the request data (I/O command) comes through transceiver 20 and is put into RAM buffer 22. Here the OP register 46 and OP decoder 48 are used to check out the validity of the requested operator OP. The parity check generator 56 will check parity and set the parity error flag 58 should the parity be improper. The parity error flag 58 will then be checked by control store 30.

The host adapter 2 makes a further check of the command request operator by using the arithmetic logic 42 and the D latch 44 to generate a longitudinal parity word as a further check upon the command request operator from the host.

At this stage of the sequence, the host adapter 2 notifies the formatter 3 that an operator (OP) is pending. Here the operator is placed from the queue (FIG. 6) into one of the device control blocks (DCB_0 through DCB_7). Further, there is a bit set in the DCB activity map (address FAO in FIG. 6).

A "wake-up" address is developed in the peripheral address counter 64 (FIG. 2) and passed on through gate 68 into the formatter 3 by control store 30, the microcode passing along line S_X on to sequencer 36 and thence to the peripheral control store 38 which will cause a Write command to be issued at the wake-up address in counter 64.

The formatter 3 will read the activity map (FIG. 6) of RAM buffer 22 at the address FAO and will use this information in order to read out the proper block (DCB0-DCB7) of the device control block of the memory 22, FIG. 6.

The formatter 3 reads the appropriate operator (OP) from the RAM buffer 22 and sets up an active flag in the command descriptor section (CMND DESC) of the flags shown as an extension of block 3 (DCB3) of the memory map of FIG. 6.

The host adapter card 2 controls the transfer of data between the host system 10 and the buffer memory 22 (FIG. 6). This is done through the control store 30 (FIG. 2), whereby a result descriptor (R/D) is formed in the "RSLT DESC" section of FIG. 6 (shown as an extension of a typical data control block). A flag is set in the interrupt message. Thus, at the end of the input/output operation, the host adapter 2 assembles the result/descriptor words in the buffer RAM 22 (FIG. 6).

The control store 30 of FIG. 2 is basically connected to all of the other elements of the host adapter system 2 and it sends out control signals and receives back information by which it can decide on the next appropriate routine to be executed.

The formatter unit 3 shown in FIG. 3 provides the logical interface between the storage module device disk drive and the host adapter card by means of the M bus.

Thus, the formatter 3 provides a bus for address data (ADRxx), a bus for data words being transferred (DATAxx) and a bus for control signals. Subsequently the host adapter card 2 then interfaces the host system 10 via the interface bus designated MLI/DLI (FIG. 1).

The formatter 3 generates (a) drive control signals via the control store 90 and the micro-decoder 99 which are passed on through registers 106, 108 and driver gates 107 and 109 on to the interface cable to the disk drive and the formatter further generates (b) data signals as follows: (i) on Write operations, the data from the host 10 and host adapter card 2 will go into the formatter RAM buffer 22_2 and thence on to the FIFO 100 for conversion from parallel to serial format; and thence through gate 104 and flip-flop 110; then through driver gate 111 on to the interface card 4; and (ii) on Read operations, the data from the selected disk drive comes through the interface 4 on to the Read data line through receiver gate 112, thence through gate 113, the shift register 101 to the FIFO 100 for conversion into parallel data and thence is transmitted to the formatter RAM 22_2. Then the data words are transferred to the M data bus through transceiver 70 and into the buffer RAM 22 of the host adapter card 2 for subsequent transmission to the host system 10.

As seen in FIG. 3, the formatter 3 receives all of the mentioned drive "status" signals from the disk drive and the "data" signals (from Read and Write operations) via the interface board 4.

Read: During the disk Read operations, the formatter 3 will accept:

(a) serial data from the selected disk via gates 125, 127, 129, 131 (FIG. 4) according to which particular drive unit has been selected;

(b) the formatter interprets the synchronization bit patterns by means of shift register unit 101;

(c) the formatter verifies data validity through the use of the cyclic redundancy checker (CRC) and the error correction code (ECC) unit 103 of FIG. 3;

(d) the formatter does serial-parallel or parallel-serial conversion through use of the FIFO register 100;

(e) the formatter passes the parallel organized data into the host adapter buffer, RAM 22, in addition to passing any data on error conditions which were detected.

Write: During the Write disk operations (OP), the formatter 3 then:

(a) performs parallel-serial conversion using the FIFO 100. Thus, data is transferred from the RAM 22_2 of the formatter 3 over to the FIFO 100, thence to the gate 104 and flip-flop 110 through driver 111 over to the interface 4 for transmission to the disk drive;

(b) transmits serial Write data over to the data driver using driver 111 over to the interface 4 and thence to the selected disk;

(c) the formatter 3 receives a "Write drive clock" signal from the disk drive to synchronize the serial data with the disk rotation.

The formatter 3 (FIG. 3) is provided with logic for the error recovery procedure. This is handled by the ECC unit 103. This unit will match the error correction code which comes from the disk drive unit and pass the information to the control store 90 of the formatter 3 so that corrective action may be instituted. A result

descriptor is then passed on to the host adapter informing it as to the action that has taken place.

Interface Card:

As seen in FIG. 4, the interface card unit 4 provides connection for data transfers between the formatter card unit 3 and the disk drive. This is basically done with the B cable shown in FIG. 5.

The interface card 4 has drivers and receivers shown in FIG. 4 and marked as either a D for driver or a R for receiver. Each disk drive unit is connected to two receivers and one driver. The two receivers are for drive unit selection (Rs) and for Read operations (R) while the one driver D is for Write operations.

As seen in FIG. 4, there are jumper switches which are attached to the drivers 117, 119, 121 and 123. These can be used as unit identification switches for each one of the four drive units connected to these drivers.

In FIG. 4 of the interface card unit 4 the unit selection logic 115 is used to detect an error condition when signals indicate that multiple units have been selected. This constitutes an error condition since only one unit can properly be selected at any given time. When a multiple unit selection signal is received, then corrective action must be initiated in order to identify and select the intended disk drive unit.

When the multiple unit selected-error signal occurs, it will inhibit or shut down the unit select mechanism through the progammed logic of 115.

As previously mentioned, the power detect logic 150 in FIG. 4 is used to turn off the disk drive interface whenever there is detected a power fault in any one of the data link processor card units 2, 3 or 4 or in system power.

As seen in FIG. 5 the A cable goes directly from the formatter unit 3 over to each of the disk drive units from D_0 over to D_7 on a daisy chain basis.

The interface unit 4 can be made by use of two interface cards 4_a and 4_b . Card 4_a provides cables B_x to four drive modules D_0 - D_3 while card 4_b provides cables to four modules D_4 - D_7.

The unit select logic 115 (FIG. 4), upon receiving a "unit select" signal (as, for example, from unit 2 via its receiver R_s to logic 115), will select the interface card 4_a of FIG. 5. However, should logic 115 receive two or more "unit select" signals, it will shut down the communication lines to the disk drive units by "disabling" drivers 125, 127, 129, 131.

In regard to the formatting of addresses for the selection of a specific disk drive and a specific cylinder and head area, the host system 10 will provide a binary (FIG. 8) address into the data buffer RAM 22. This data will be placed in the data control block DCB (FIG. 6) and passed on to the formatter data buffer RAM 22_2.

This information is then translated into cylinder data, head data and sector data which is transmitted on the A cable directly to the selected disk drive unit.

The cylinder data, head data and sector data resides in the formatter data buffer RAM 22_2 before being passed on to the selected disk drive.

Referring to the Read data line "RD data" shown in FIG. 3, the data words come from the disk through the interface 4 and over to the receiver 112. Thence it is passed on through gate 113 and shift register 101 over to the FIFO register 100, as well as gate 102 to 103 to generate an ECC for the incoming data.

A comparison occurs at this stage of the sequence by use of the arithmetic logic unit 80 which compares the generated ECC word and the ECC Read from the disk so that it can give an "OK" (enable signal) to the control store 90 to continue with transferring data words from the disk drive into the formatter data RAM 22_2 and thence over the

data bus to the host access data RAM 22 and thence on the data bus to the host system 10.

While the described data link processor and disk peripheral system has been described in its preferred embodiment, other variations and embodiments may be effectuated but which still are encompassed by the concepts as defined in the attached claims.

What is claimed is:

1. In a data transfer network wherein a single peripheral-controller manages data transfers between a main host computer and selected ones of a plurality of disk drive storage modules, a peripheral-controller maintenance test system comprising:

(a) first control means for receiving and storing I/O commands from said host computer and transmitting a selected I/O command to a second control means;

(b) second control means for receiving said selected I/O command and for formatting address data and Read/Write control data to a disk module interface means, and including:

(b1) means to execute data word transfers between a selected disk module and said host computer;

(c) disk module interface means for transmitting said address data and Read/Write control data to a selected disk module and for enabling data transfers between said selected disk module and said second control means;

(d) host computer means connected to said first control means and including:

(d1) means to generate I/O commands for accessing a selected disk drive module to cause the execution of data transfers, said I/O commands including:

(d1a) disk drive module identification data;

(d1b) said address data for selecting cylinder address, track/head address, and sector address;

(d1c) said Read/Write control data specifying the action dictated by said I/O command;

(d2) means for receiving result-data from said first control means to provide information regarding said I/O command and operating status conditions in said first and second control means;

(e) means for initiating and executing a self-test operation of said first control means, second control means and said disk interface means.

2. The system of claim 1, wherein said means for initiating and executing a self-test operation includes:

(a) a self-test initiate command generated by said host computer for transfer to said first control means for execution.

3. The system of claim 1, wherein said first control means includes:

(a) push button means for initiating a self-test operation in a first control sequencer means;

(b) first control sequencer means for initiating test routines in a first control store RAM and for assembling result/data in a first control buffer memory means;

(c) first control buffer memory means for temporarily storing result/data assembled by said first control sequencer means.

4. The system of claim 1, wherein said second control means includes:

(a) push button means for initiating a self-test operation in a second control sequencer means;

(b) said second control sequencer means for initiating test routines in a second control store RAM and for assembling result/data in said first control buffer memory means.

5. The system of claim 2, wherein said first control means includes:

(a) means to assemble result/data of an executed self-test operation;

(b) means to transmit said result/data to said host computer.

6. The system of claim 3, wherein said first control means includes:

(a) first light indicating means to indicate a fault condition within said first control means.

7. The system of claim 3, wherein said first control means transmits said result/data to said main host computer.

8. The system of claim 4, wherein said second control means includes:

(a) second light indicating means to indicate a fault condition within said second control means.

9. The system of claim 4, wherein said first control means transmits said result data to said main host computer.

10. The system of claim 4, wherein said means for initiating and executing a self-test operation includes:

(a) means for sensing a power turn-on (power up) condition;

(b) means responsive to said sensing means for initiating said first control sequencer means and said second control sequencer means to execute self-test operations.

11. In a data transfer network wherein a single peripheral-controller manages data transfers between a main host computer which generates I/O commands and a plurality of individual disk drive storage modules, said peripheral-controller having means to connect and disconnect from said main host computer, the said peripheral-controller comprising:

(a) first control means for receiving and storing a group of "X" I/O commands, where "X" represents the number of disk drive modules serviced by said peripheral-controller, and including:

(a1) means for connecting to and disconnecting from said host computer;

(a2) a first buffer memory means for storage of said I/O commands and for storage of data words being transferred;

(a3) host sequencer means for executing data word transfers between said host computer and said first buffer memory means;

(a4) peripheral sequencer means for executing data word transfers between said first buffer memory means and a second control means;

(b) second control means for receiving control data and address data for selection of a specific cylinder, track and sector on a selected one of said plurality of disk drive modules, and including:

(b1) second buffer memory means for storing data words received from said host computer or from a selected disk drive module;

(b2) third sequencer means for executing data word transfers from said second buffer memory means to said first buffer memory means, and for executing data word transfers between said second buffer memory means and a selected disk drive module;

(b3) means for communicating with a disk module interface means, said communicating means including:

(b3a) Read data means for receiving data words from said selected disk module into a conversion means for storage into said second buffer memory means;

(b3b) Write data means for receiving data words from said conversion means for sychronized transmittal to said disk module interface means;

(b3c) conversion means for executing serial-to-parallel and parallel-to-serial conversion of data words;

(b3d) register-driver means for transmitting control signals to said disk module interface means;

(b3e) disk drive module select signal means for enabling connection to said selected disk drive module;

(c) disk module interface means for enabling connection of a selected disk drive module to said second control means, said interface means including:

(c1) select logic means for disabling connection to said selected disk drive module when said logic means receives more than one disk drive select signal, said logic means including:

(c1a) means for generating a disk drive select signal in response to instructions in said I/O command.

12. The peripheral-controller of claim 11 wherein said first buffer memory means includes:

(a) a queue storage area for storing up to "X" sets of I/O commands, one for each of "X" disk drive modules;

(b) a data word storage area for temporarily storing data words being transferred between said host computer and a selected disk drive module;

(c) a device control storage area for temporarily storing data for addressing a desired track/sector in each disk drive module.

13. The peripheral-controller of claim 12 which includes:

(a) means for immediately executing an I/O command from said host computer even while said queue storage area is fully loaded with I/O commands.

14. The peripheral-controller of claim 13, wherein said means for immediately executing includes:

(a) a scratch queue area in said first buffer memory means for storing an immediate I/O command which is granted first priority in execution.

15. The peripheral-controller of claim 13 wherein said second control means includes:

(a) attribute table memory means for storage of data on operating characteristics of each drive module, said data being addressable by said third sequencer means.

16. The peripheral-controller of claim 11, wherein said second control means includes:

(a) means for verifying the accuracy of data read and received from a selected disk drive module during a Read operation;

(b) means for transmitting an error signal to said first buffer memory means when the accuracy of received data is not verified;

(c) said conversion means for converting data words from serial-to-parallel on said Read operation.

17. The peripheral-controller of claim 11, wherein said second control means includes:

(a) said conversion means for converting data words from a parallel-to-serial configuration on a Write operation;

(b) means for synchronizing a write-clock signal from a selected disk module with data words being serially transmitted to said selected disk module;

(c) means for generating an error correction code (ECC) for writing at the end of a selected data field on a selected disk track/sector.

18. The peripheral-controller of claim 17, wherein said disk module interface means includes:

(a) means for disabling said interface means upon the occurrence of inadequate power being supplied to said host computer or said first and/or second control means.

19. The peripheral-controller of claim 17 which includes:

(a) means for automatically correcting errors in seeking a desired track/sector in a disk drive module by commanding the drive head to the "zero" cylinder position and retrying the seek for the desired track/sector.

20. The peripheral-controller of claim 17 which includes:

(a) means for detecting address errors in the track/sector address data and re-reading the desired sector for up to "k" times until a correct address is verified.

21. The peripheral-controller of claim 17 which includes:

(1) means for detecting errors in data read-out from an addressed disk sector and for executing "k" retries of sector read-out until accurate data read-out is verified.

22. The peripheral-controller of claim 17, wherein said second control means includes:

(a) error detection means for writing data on a selected disk and reading back said data for comparison with the originally written data;

(b) error correction means for taking inaccurate data read from a disk and processing the data to conform to accurate data.

FIG.1. Typical DLP Configuration.

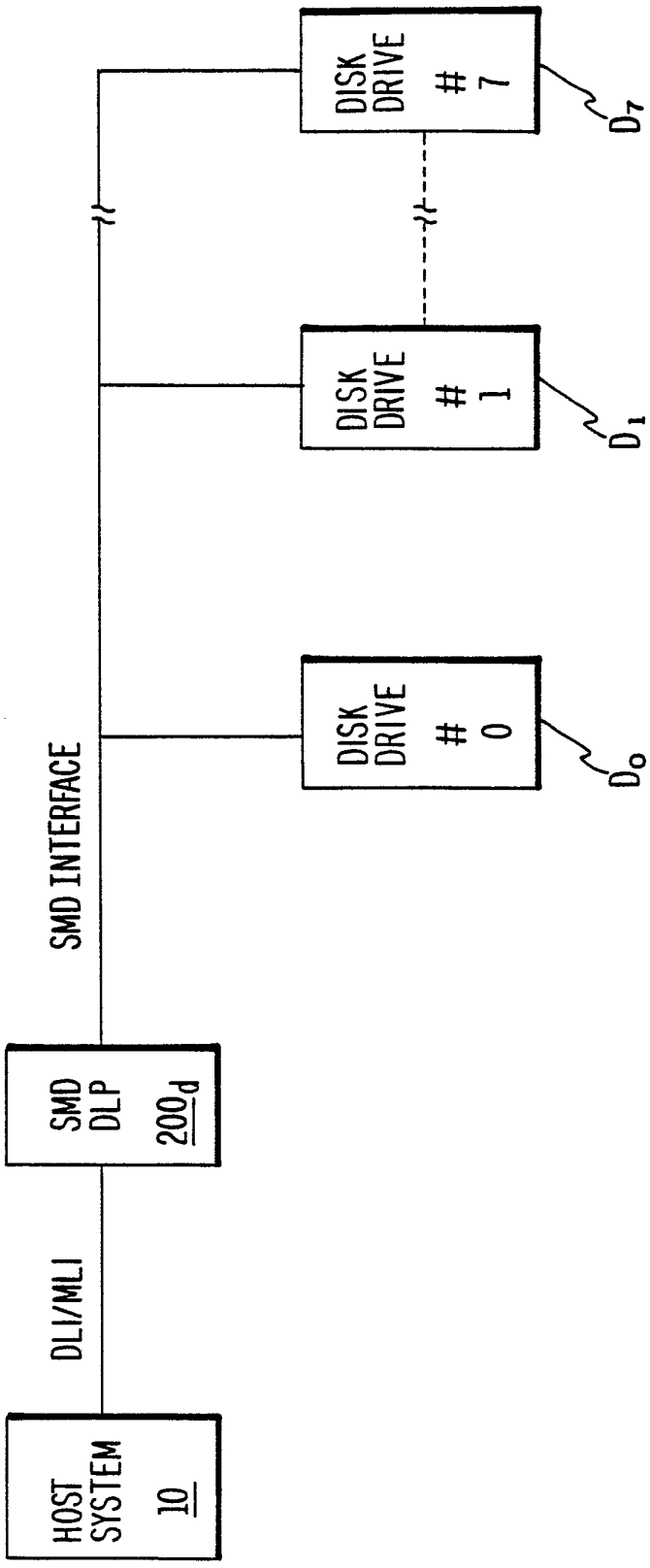


FIG.2A. Host Adapter Basic Block Diagram. 2/11

FIG.2.

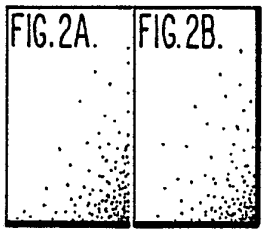


FIG.2B.

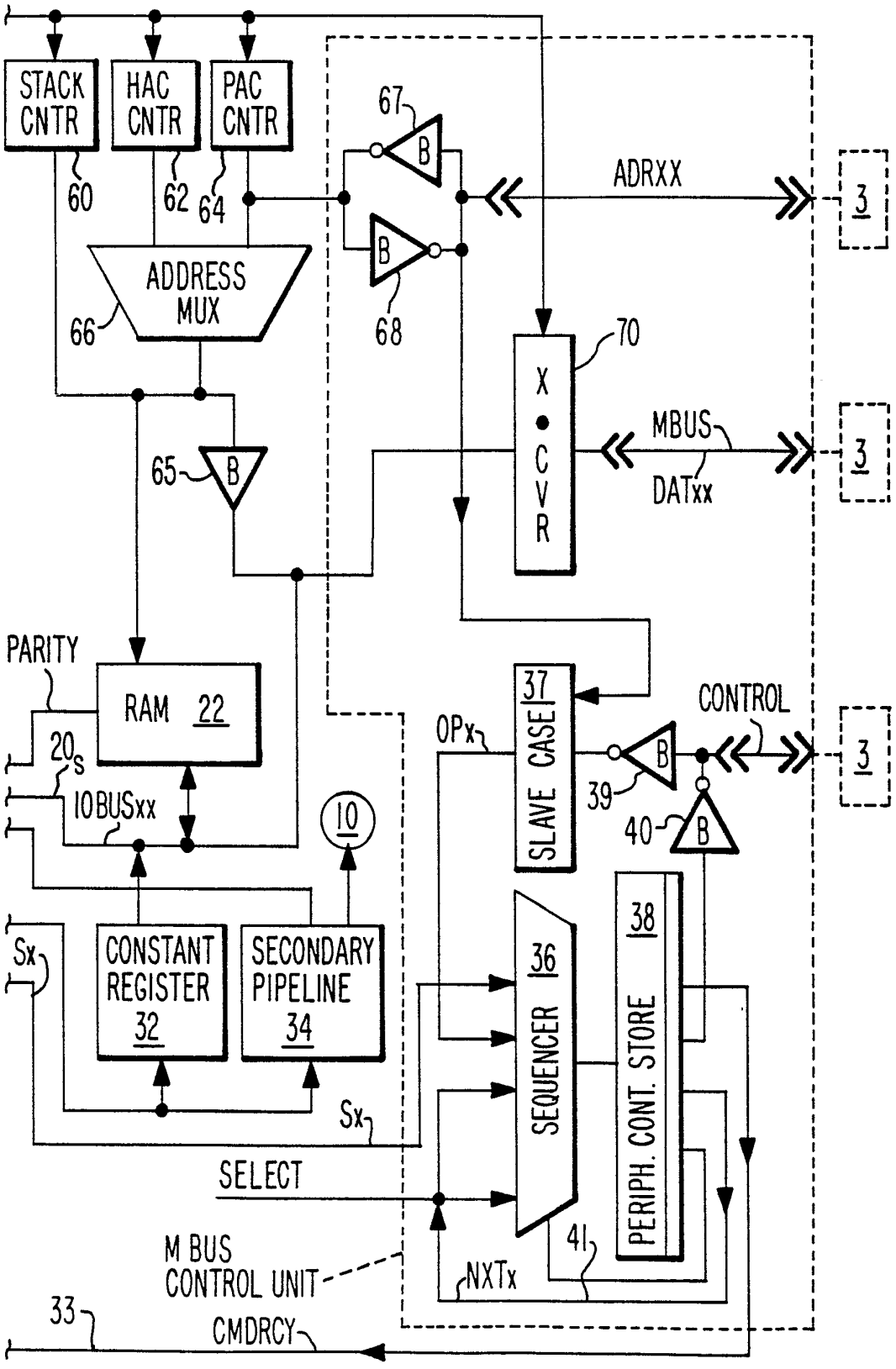

STACK CNTR
HAC CNTR
PAC CNTR
60
62
64
67
B
68
ADDRESS MUX
66
ADRXX
3
70
X • C V R
MBUS
DATxx
65
PARITY
RAM 22
20s
IOBUSxx
10
37
SLAVE CASE
OPx
CONTROL
39
40
CONSTANT REGISTER 32
SECONDARY PIPELINE 34
Sx
36
SEQUENCER
38
PERIPH. CONT. STORE
SELECT
M BUS CONTROL UNIT
41
NXTx
33
CMDRCY

FIG.3A. Formatter Card Basic Block Diagram. 4/11

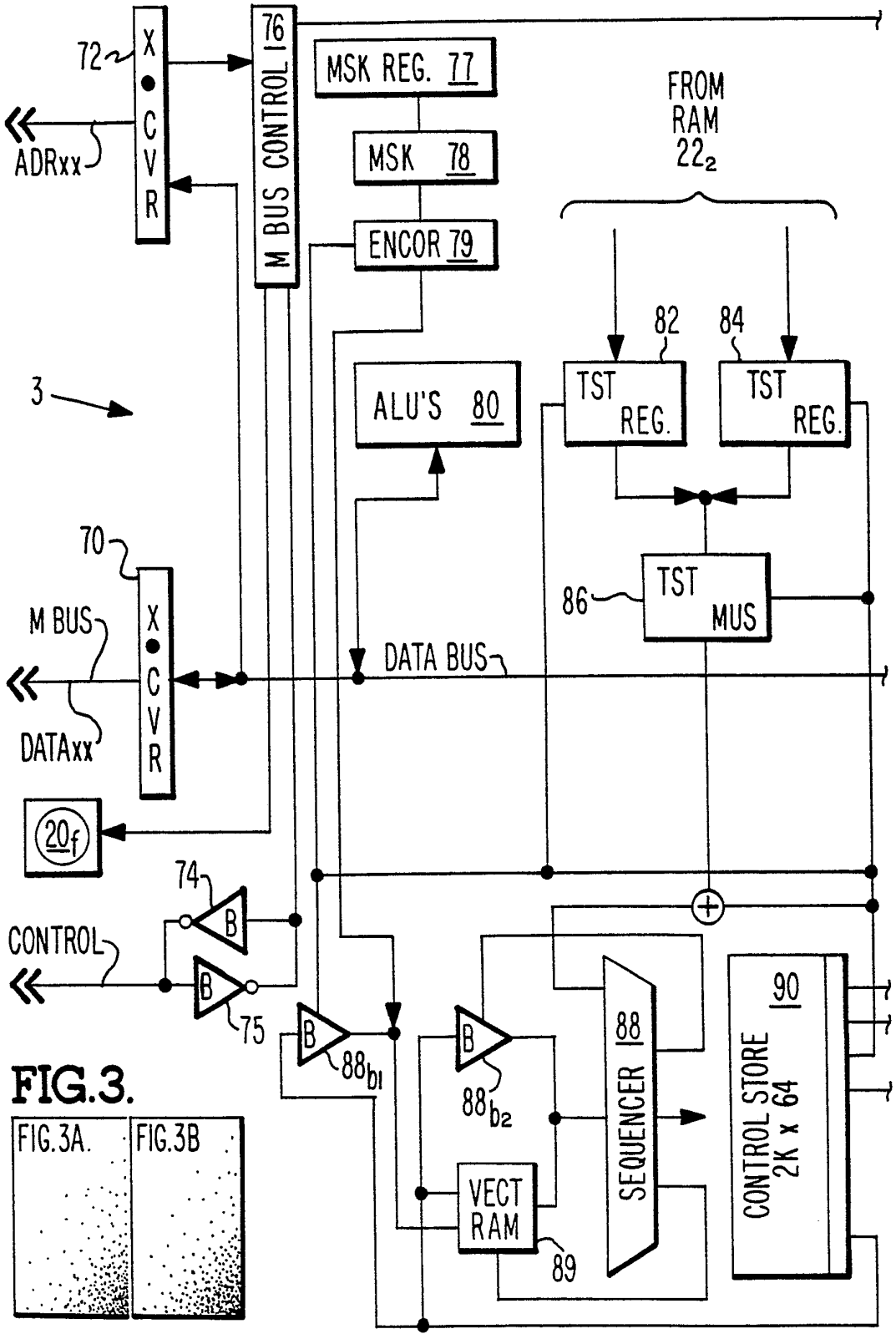


FIG.3.

FIG.3B.

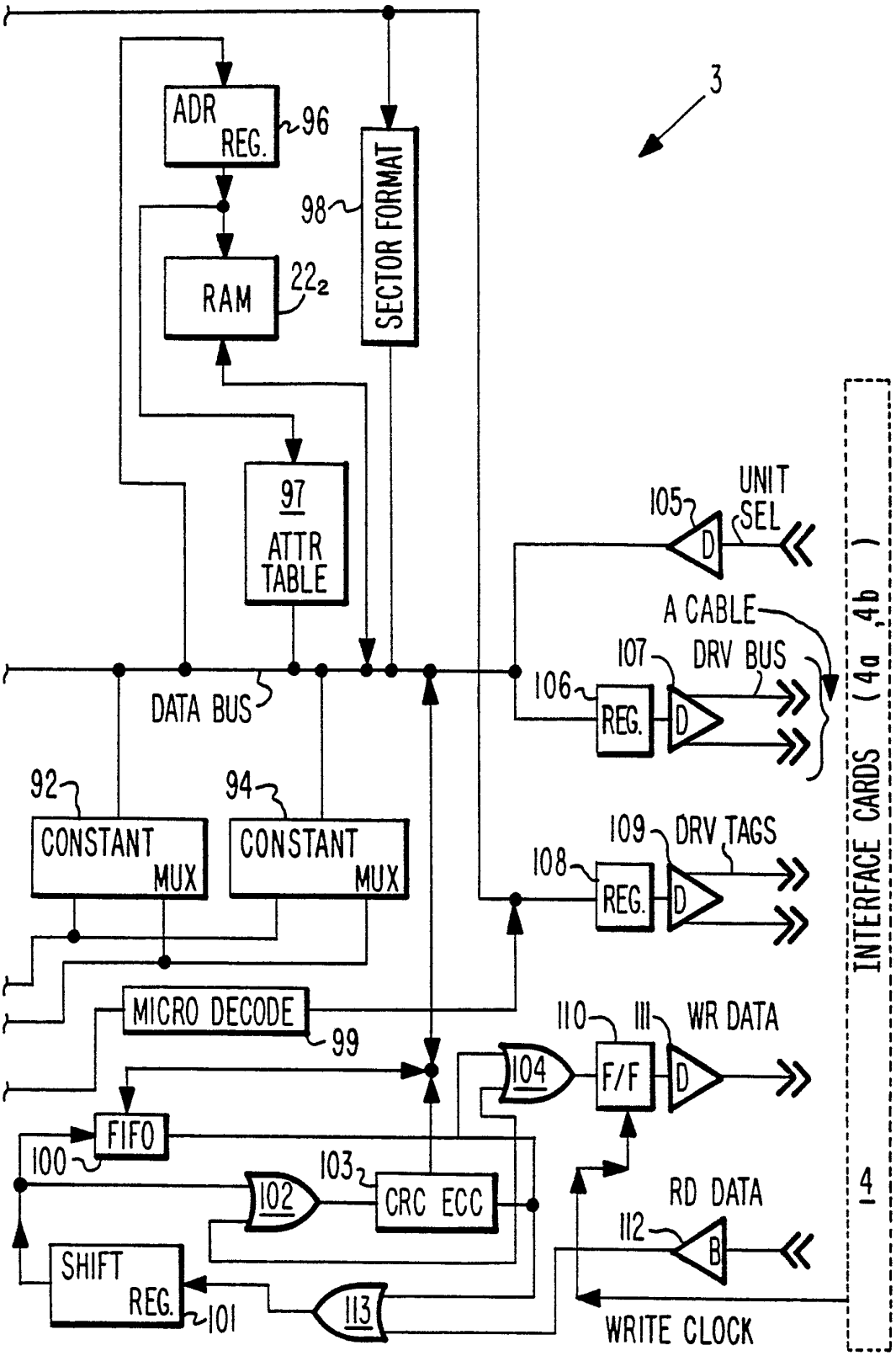

3
ADR REG.
96
98
SECTOR FORMAT
RAM
22₂
97
ATTR TABLE
DATA BUS
92
CONSTANT MUX
94
CONSTANT MUX
MICRO DECODE
99
FIFO
100
102
103
CRC ECC
SHIFT REG.
101
113
104
105
UNIT SEL
A CABLE
107
DRV BUS
106
REG.
109
DRV TAGS
108
REG.
110
111
WR DATA
F/F
RD DATA
112
WRITE CLOCK
INTERFACE CARDS (4a ,4b)
4

FIG. 4A. Interface Card Basic Block Diagram.

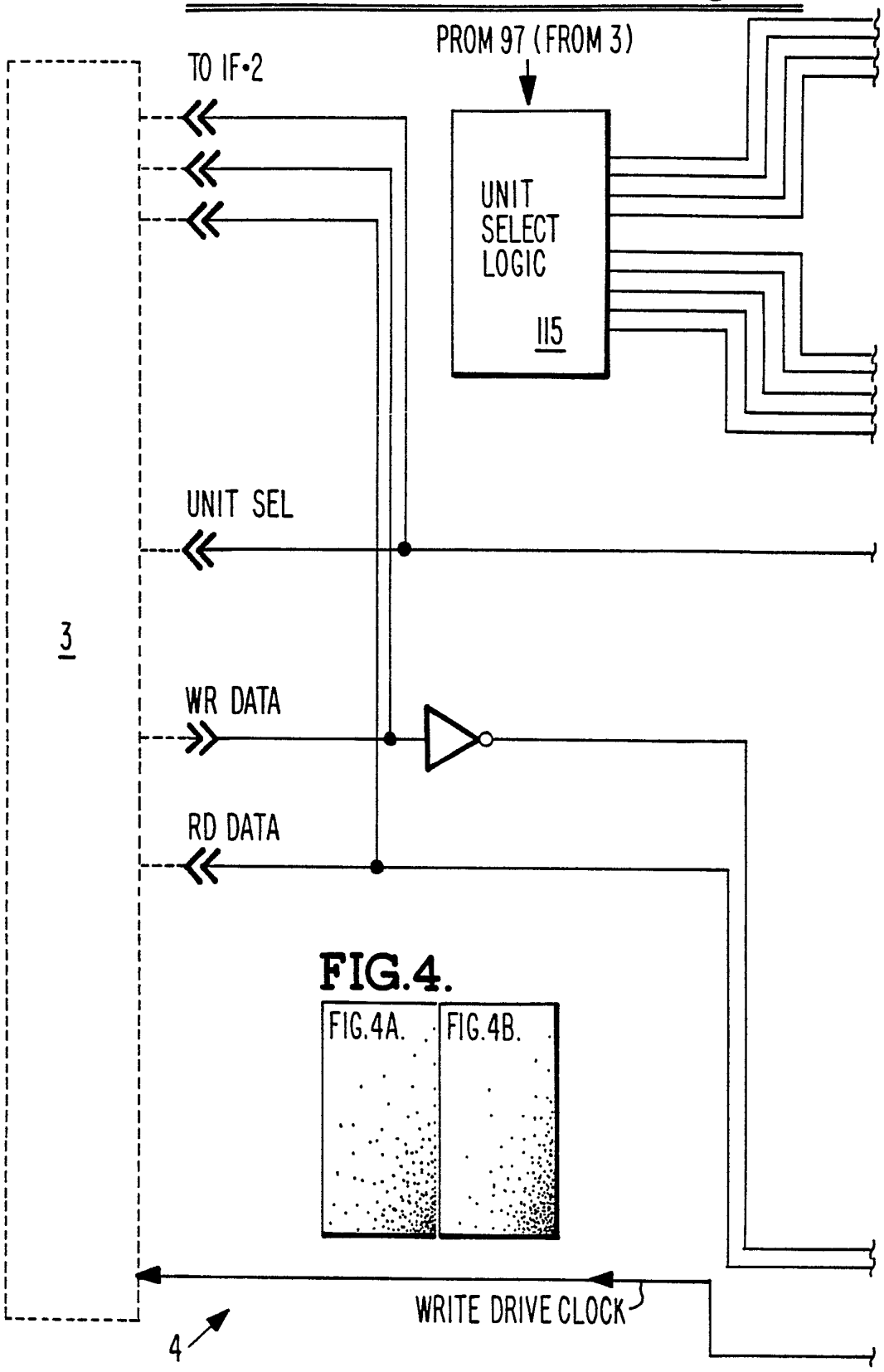
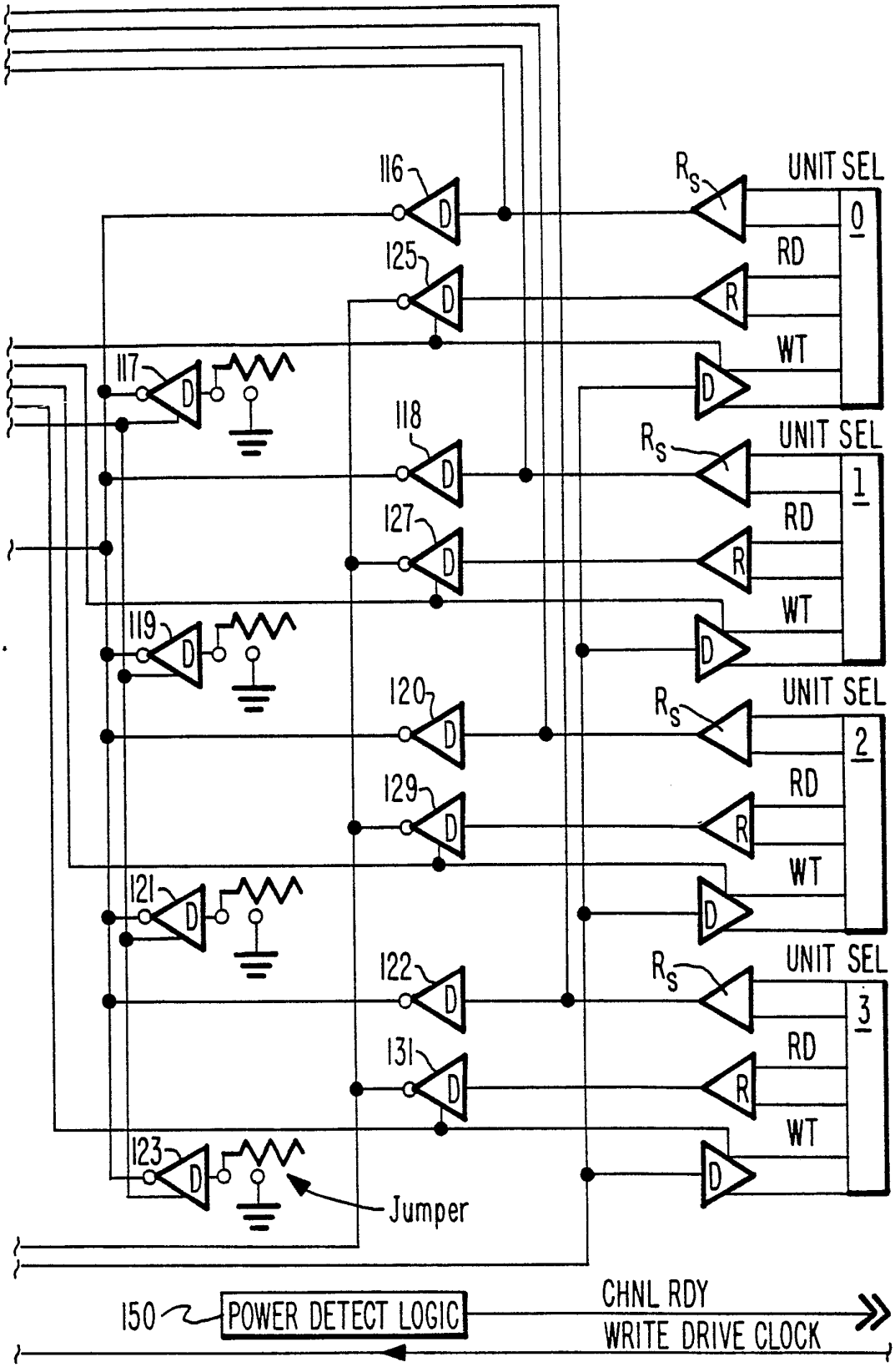


FIG. 4.

FIG.4B.


116
125
117
118
127
119
120
129
121
122
131
123
Jumper
R_S
UNIT SEL
0
RD
R
WT
D
1
2
3
150
POWER DETECT LOGIC
CHNL RDY
WRITE DRIVE CLOCK

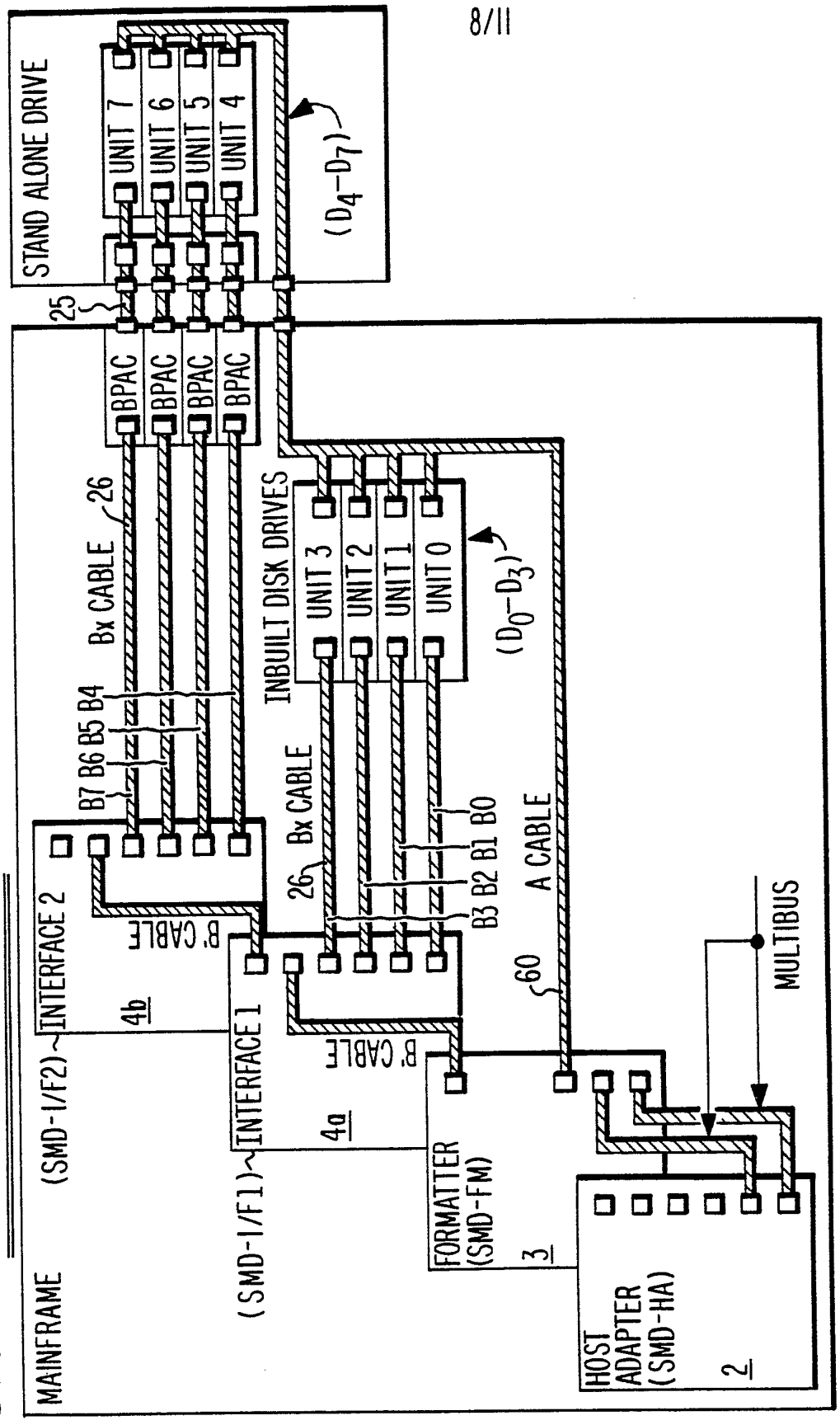


FIG.5. SMD DLP Cabling.

FIG. 6. SMD HA Memory Map (22, of Fig. 2.1

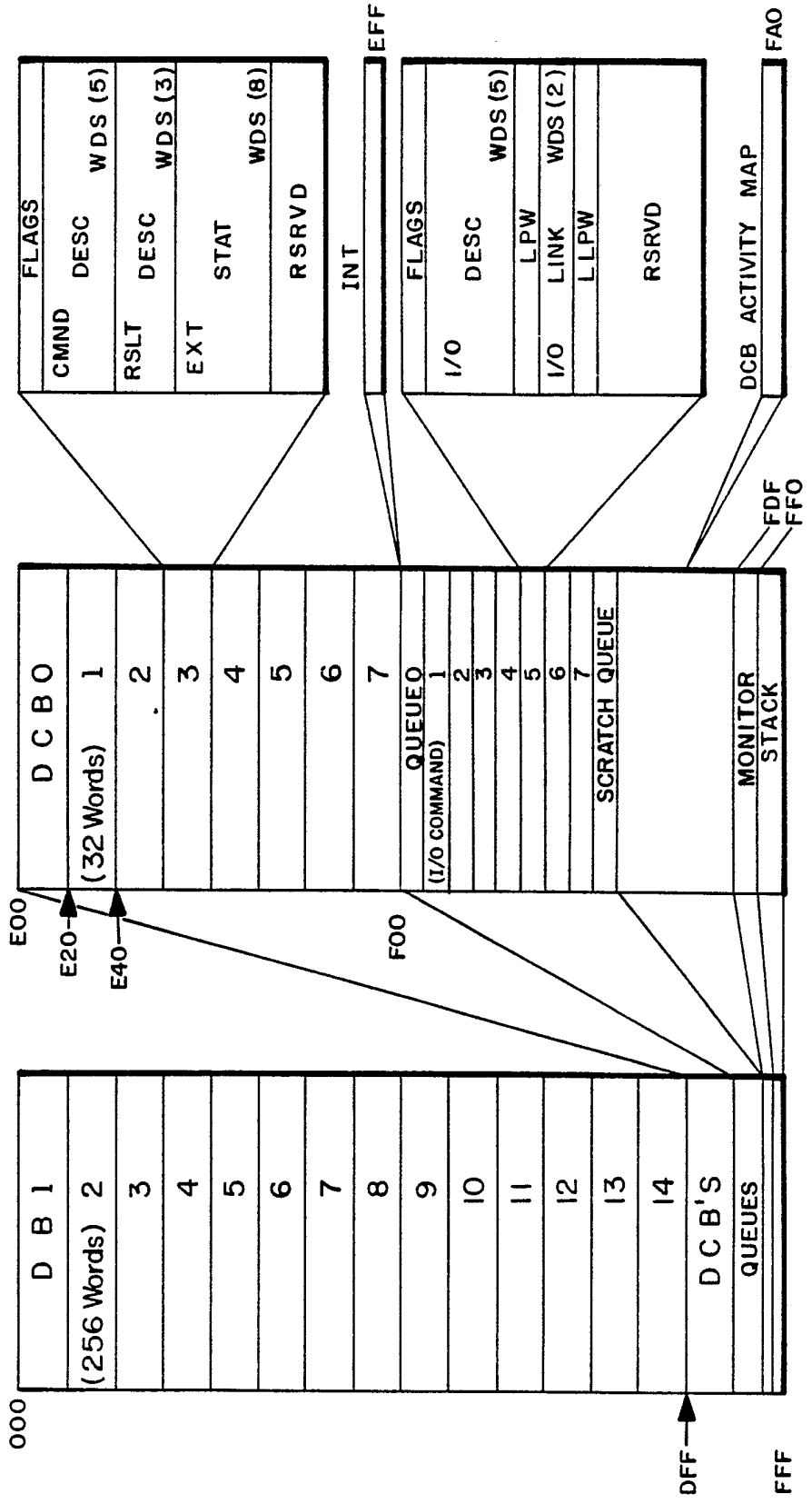


FIG. 7. Track / Sector Format.

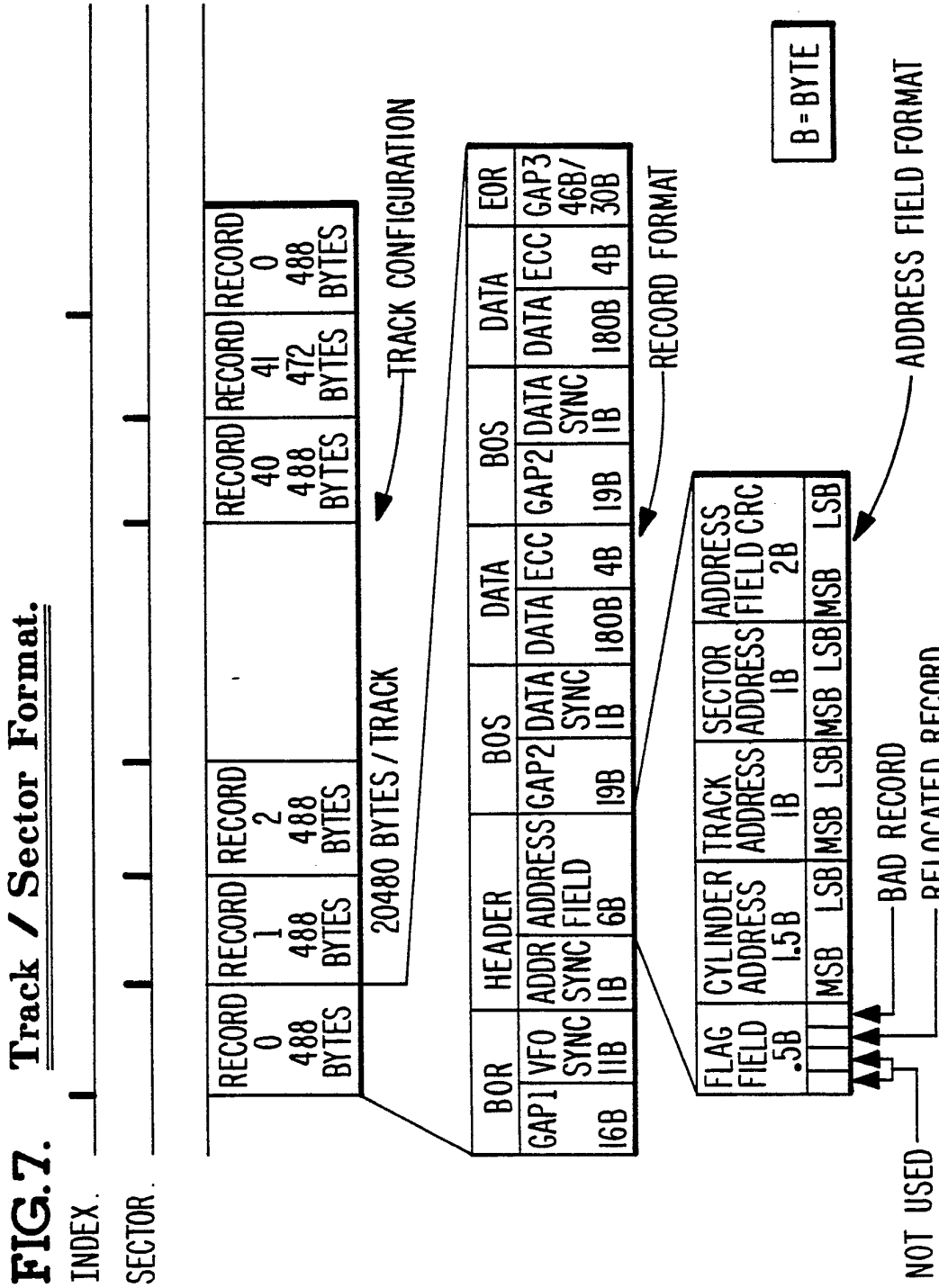
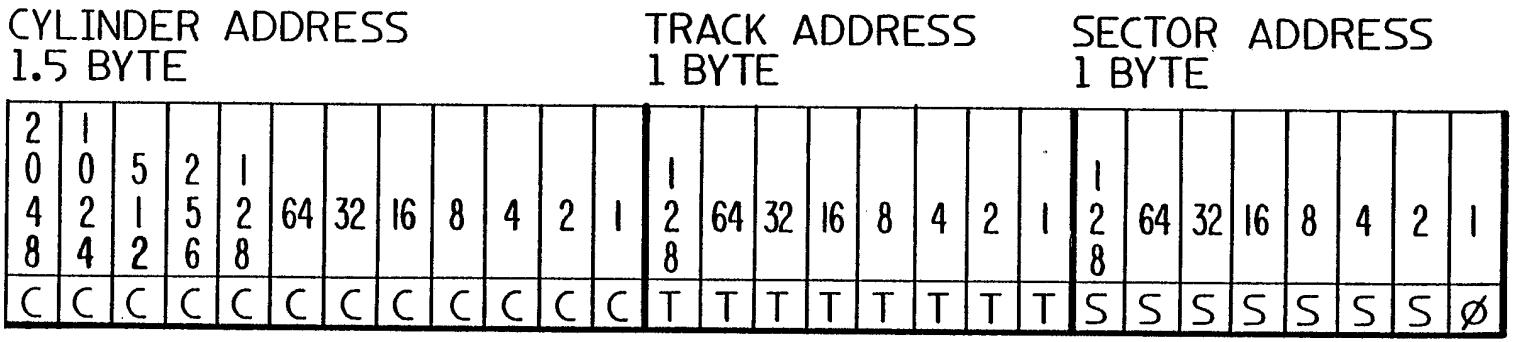


FIG. 8.



C = CYLINDER ADDRESS

MRX 214 : Ø → 24C HEX (Ø → 588 DECIMAL)

MRX 226 : Ø → 337 HEX (Ø → 822 DECIMAL)

T = TRACK ADDRESS

MRX 214 : Ø → 6

MRX 226 : Ø → 9

S = SECTOR ADDRESS

MRX 214 : Ø → 53 HEX (Ø → 83 DECIMAL)

MRX 226 : Ø → 53 HEX (Ø → 83 DECIMAL)

[ONLY EVEN SECTOR NUMBERS ARE RECORDED IN THIS FIELD, Ø, 2, 4, ••• 82]

INTERNATIONAL SEARCH REPORT

International Application No PCT/US 86/01665

I. CLASSIFICATION OF SUBJECT MATTER (if several classification symbols apply, indicate all) [5]

According to International Patent Classification (IPC) or to both National Classification and IPC

IPC[4]: G 06 F 3/06

II. FIELDS SEARCHED

Minimum Documentation Searched [7]

Classification System	Classification Symbols
IPC[4]	G 06 F 3/06; G 06 F 11/22; G 06 F 11/26 G 06 F 13/42

Documentation Searched other than Minimum Documentation to the Extent that such Documents are Included in the Fields Searched [8]

III. DOCUMENTS CONSIDERED TO BE RELEVANT [9]

Category [10]	Citation of Document, [11] with indication, where appropriate, of the relevant passages [12]	Relevant to Claim No. [13]
A	US, A, 4390964 (HORKY et al.) 28 June 1983 see column 4, lines 51-68; column 5, lines 19-40; column 10, lines 13-20 cited in the application	1,11
A	US, A, 4313162 (BAUN et al.) 26 January 1982 see column 5, lines 41-50; column 26, line 65 - column 27, line 3 cited in the application	1,11
A	US, A, 4183084 (LAWSON) 8 January 1980 see column 2, lines 61-64; column 2, line 67 - column 3, line 3; column 4, lines 7-11	1,11
A	EP, A2, 0126610 (DATA GENERAL CO.) 28 November 1984 see page 1, lines 1-21; page 2, lines 27-36; page 3, lines 21-24	1,11
A	US, A, 3999163 (LEVY et al.) 21 December 1976 see column 3, lines 54-60	1,11

./.

* Special categories of cited documents: [10]

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

IV. CERTIFICATION

Date of the Actual Completion of the International Search	Date of Mailing of this International Search Report
27th February 1987	- 2 APR 1987
International Searching Authority	**Signature of Authorized Officer**
EUROPEAN PATENT OFFICE	M. VAN MOL

Form PCT/ISA/210 (second sheet) (January 1985)

III. DOCUMENTS CONSIDERED TO BE RELEVANT (CONTINUED FROM THE SECOND SHEET)

Category *	Citation of Document, with indication, where appropriate, of the relevant passages	Relevant to Claim No
A	Review of Electrical Communication Laboratories, volume 30, no. 1, 1982, (Tokyo, JP), Takanami et al.: "Storage control for 3.2Gbyte multi-device disk storage", pages 8-13 see pages 11-12, paragraph 5: "Test and microdiagnostic program"	1,11
A	US, A, 3525081 (FLEMMING et al.) 18 August 1970 see column 1, lines 37-52; column 2, lines 13-27; column 3, lines 63-68	1,11
A	EP, A, 0104091 (XEROX CORP.) 28 March 1984 see abstract	1,11
A	IBM Technical Disclosure Bulletin, volume 16, no. 6, November 1973, (Armonk, US), Houdek et al.: "Internal wrap test for microprogrammed binary synchronous communications adapter", page 1862 see page 1862, lines 1-9	1,11
A	Patents Abstracts of Japan, volume 8, no. 107 (E-245)(1544), 19 May 1984, see abstract, & JP, A, 5923944 (Fujitsu K.K.) 7 February 1984	1,11