THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 62, No. 1, January 1983
Printed in U.S.A.

The 3B20D Processor & DMERT Operating System:

3B20D File Memory Systems

By R. E. Haglund and L. D. Peterson

(Manuscript received March 10, 1982)

The 3B20D file memory system employs microprocessor-based intelligent Disk File Controllers (DFC), each supporting a complement of one to eight 300-megabyte disk drives. The DFC communicates with the system via a high-capacity dual-serial channel link to the Direct Memory Access Controller (DMAC) that accesses the processor's main memory. The DFC communicates with its disk files via an industry standard interface. The disk power system is also microprocessor based. The system operates disk drives on commercial ac power until a power failure occurs. The disks are then switched under microprocessor control to inverter power until the commercial ac fault is cleared, at which time the disk is returned to commercial ac power.

I. INTRODUCTION

The 3B20D file store consists of a microprocessor based Disk File Controller (DFC) plus one to eight 300-megabyte disk files. The files are powered by a microprocessor-based power system consisting of an inverter, a cycloconverter and a static switch. The DFC communicates via the 3B20D Dual-Serial Channel (DSCH) and through the Direct Memory Access Controller (DMAC) to the processor's main memory (see Ref 1). The DFC communicates with its disk files via a disk interface that utilizes the industry standard Storage Module Drive (SMD) interface.

The DFC provides for stand-alone processing of disk accesses via approximately 30,000 bytes of code stored in its PROM memory. Also stored in the PROM memory are 30,000 bytes of diagnostic code.[2]

The DFC uses a bit-sliced bipolar microprocessor to manage the 10-

MHz serial data streams to/from the disk files, the disk control functions, and the system interface. In addition, the processor runs a disk exerciser program to test disk drives during periods when no system accesses are in progress.

II. SUBSYSTEM COMPONENT DESCRIPTION

The disk file memory system for the 3B20D Processor consists of three major elements: a DFC, 300M-byte moving head disk memories, and a 208 Vac power-control system for the disk memory units. The power system consists of a Power Control Unit (PCU) and an emergency -48 Vdc to 208 Vac Disk File Inverter (DFI).

2.1 The 300-megabyte disk drive

The disk drives used in 3B20D Processor are high-speed, random-access, data-storage devices (Fig. 1) supplied by Century Data Systems and the Control Data Corporation. The disk drive unit consists of two physical parts. The basic drive assembly contains the chassis, power supply, air-filtration system, read/write head assembly, control circuit cards, and the input/output interface. The second part is the removable disk pack upon which data is recorded.

The disk pack contains 12 stacked platters. The inner ten are coated on both sides with the recording medium, ferrous oxide. The top and bottom platters are only for disk pack protection. The remaining ten platters have a total of 20 recording surfaces, of which 19 are used for recording data while the other contains prerecorded servo information that is used for head positioning and timing. Each disk pack is capable of storing 300M bytes (unformatted) of information. Each read/write head may be positioned over 815 different cylinders. The 19 read/write heads plus the servo head are mechanically attached together on a single carriage arm. Therefore, all heads are moved in parallel such that all heads are always positioned over the same cylinder. By selecting one of the 19 data surfaces, an individual track is selected. Each track is divided into 32 sectors or blocks of 512 bytes each. A sector is the smallest quantity of data written or read by a disk file controller.

Communication to and from the drive is via the industry-standard SMD type of interface. Two cables interconnect the drive to the disk file controller. One contains the control and status signal lines while the other contains the read and write data signal lines. The disk drive is powered from 208-volt single-phase power. Starting current is a maximum of 50 amperes, with running current of approximately 9 amperes. The disk pack is rotated at approximately 3600 RPM; thus the average rotational latency is 8.5 ms. The average head positioning

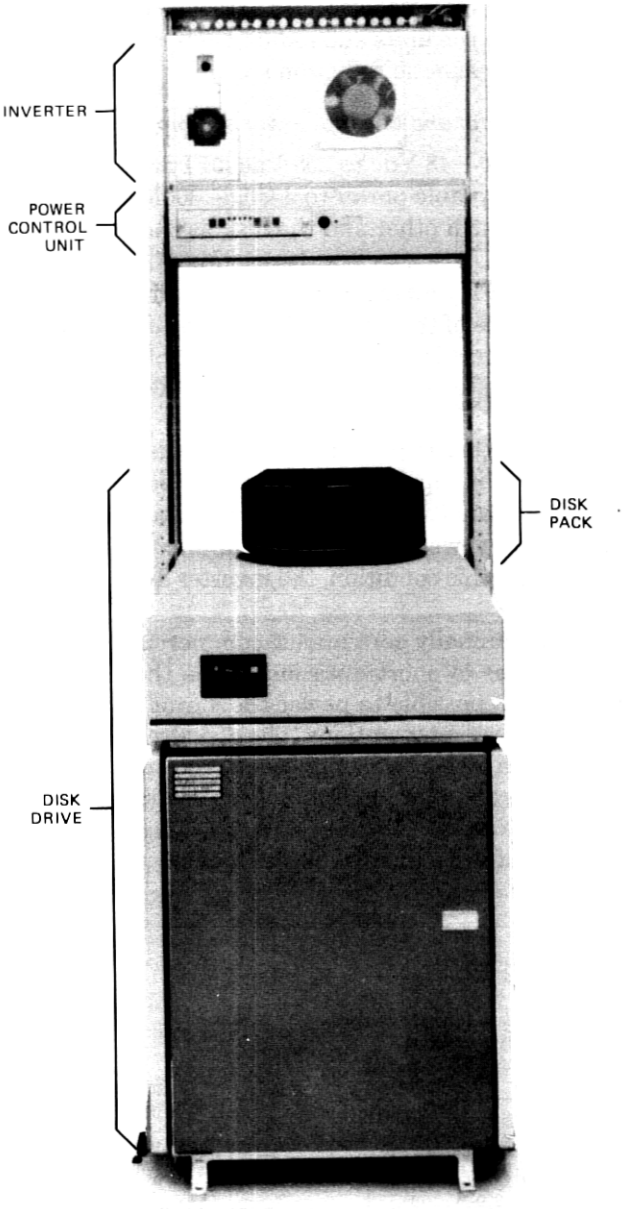

Fig. 1—Moving head disk frame.

(seek) time is 30 ms. Data is transferred to and from the disk drive at a 9.67-MHz serial data rate.

The disk units are designed to meet Bell System environmental requirements. Acoustical noise generated by a disk unit is less than 65

dBA. RFI gasketing material was used extensively to limit electromagnetic radiation from the units and reduce the susceptibility of the units to external electromagnetic radiation sources.

2.2 Disk power control and backup power system

The PCU and the −48 Vdc to 208 Vac DFI are used in combination to supply uninterruptable power to a single 300M-byte disk. The units interconnect with each other and the disk as shown in Fig. 2.

Inputs to the PCU are −48 Vdc, two sources of 208 Vac, and control signals from the DFI. Power for the logic circuitry is derived from the −48 Vdc. One source of the 208 Vac is from the commercial power grid or the essential ac grid in a central office. The other source of 208 Vac is from the DFI. Status and control functions also are received by the PCU from the DFI.

Cooling for the DFI is provided by two fans. The fans are active only when the unit is supplying 208 Vac. The power dissipated by the DFI under a load condition is 500 watts when supplying 2000 watts to the disk. Normally power to the disk is supplied by the commercial or essential ac. Under this condition, the inverter dissipates no more than 150 watts.

Even though nominally continuous, commercial power is subject to minor interruptions of short duration. Since the disk pack has substantial inertia it is possible to power the motor from the commercial ac mains despite such interruptions. The disk drives used in the 3B20D Processor have ferroresonant supplies for their own internal dc power, and these supplies are tolerant of minor interruptions in input voltage.

The 3B20D disk power system takes advantage of these characteristics to deliver power to the disk drive from the most efficient source.

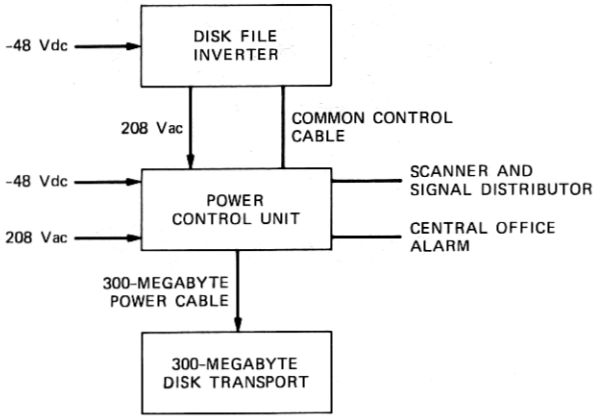

Fig. 2—Interconnections between the control panel, inverter, and disk transport cabling units.

The DFI powered from the battery plant supplies the power for the disk drive when the ac mains are not available or out of tolerance in voltage or frequency. A silicon controlled rectifier static switch provides the changeover within two cycles of the time the ac mains are found to be out of tolerance. The inverter is phase locked to the ac mains voltage when it is present so that the transient due to the switchover to battery power is minimized. The operation of the disk drive is unaffected by the switchover between ac and inverter power.

The standby inverter (Fig. 3) operates at 600 Hz; its output is converted to the required 60 Hz ac by means of a tap-changing cycloconverter. This provides a piecewise approximation to the desired 60-Hz sine wave (Fig. 4). The inverter frequency was chosen high enough to allow the use of a small, efficient inverter transformer and also to allow a sufficient number of intervals in which the output voltage could be selected to make a good approximation to a sine wave. The cycloconverter output has the same peak and rms values and same average value over a half cycle as the sine wave it replaces. The third and fifth harmonics are also reduced to small values by appropriate choice of cycloconverter parameters.

The disk power system is controlled by a microprocessor housed in the inverter. This processor administers the operation of the inverter, cycloconverter, and static switch so that the proper power source is connected to the disk drive. The choice of power source is made on

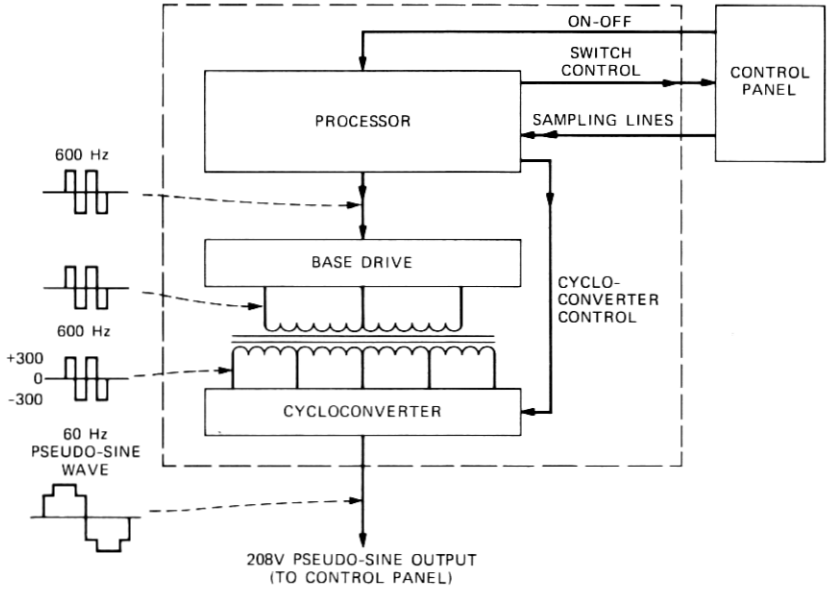

Fig. 3—Disk inverter operation.

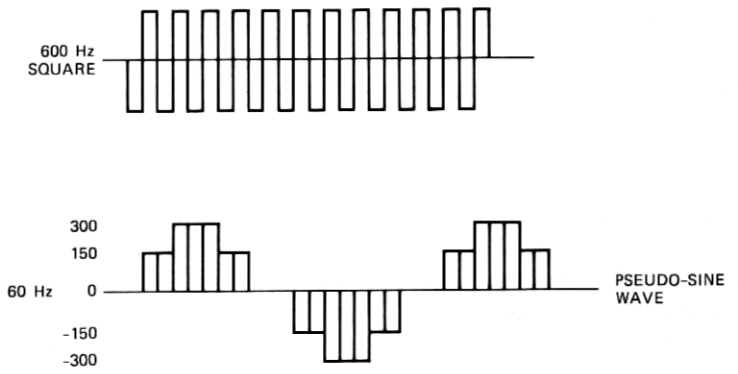

Fig. 4—Pseudo-sine wave generation.

the basis of sampling the ac mains for voltage above the programmed threshold value ten times per half cycle. If the mains voltage supply is found to be deficient for two samples in any half cycle, the static switch disconnects the ac mains and the cycloconverter supplies power to the disk drive. When the mains voltage has been determined to be within acceptable limits in voltage and frequency for a sufficient time, the cycloconverter output is inhibited and the static switch again allows the mains to power the disk drive.

While operating from the battery plant, the inverter regulates its output voltage to compensate for battery variations. A reflex regulator circuit uses some of the 600-Hz inverter output voltage in a phase-controlled rectifier to control the apparent input voltage to the inverter to maintain it constant at −52.2 volts in the face of battery voltages in the range of −42.75 to −52.5 volts. The microprocessor controls the operation of the regulator so that the peak current demands on the inverter transistors are not increased by the operation of the regulator. The regulator operation is inhibited during the times the peak output current is delivered to the load, and regulator operation is only allowed when the 60-Hz output is in the part of the cycle where the output voltage, and therefore the inverter load current, is near zero.

Since it may be necessary to change the disk pack while ac power is not available, the inverter must be able to start the drive motor. The current required to start the disk drive motor is much larger than the current required to run the motor. Since it would be uneconomical to size the inverter to carry the starting load, the microprocessor controls the cycloconverter output voltage during the time the motor is starting so that the starting current is greatly reduced. This power reduction is done at the expense of starting time; the time required to start the drive motor is 45 seconds on battery and 15 seconds on the commercial mains.

With the inverter always powered up even while the disk is being

powered by commercial ac, the possibility exists that the power-handling components could suffer a failure that would be undetected until the inverter was required to supply power. A diagnostic triggered by a disk being out of service is available to detect this failure. The microprocessor continually monitors the state of the power switch to detect when the disk has been taken out of service. When this condition is detected, the microprocessor starts the diagnostic routine, which causes the inverter to supply power to the disk drive for a few minutes while monitoring the cycloconverter output voltage for the correct levels. Thus, any failure of power-handling components will be detected during the regular daily diagnostic exercise of the disk frame.

2.3 Moving head disk frame

The Moving Head Disk (MHD) frame (Fig. 1) is the frame configuration used to mount a disk, PCU and DFI.

The PCU and DFI share a common interface cable for transmitting status and control information. The DFI 208-Vac output is connected to the PCU via a flexible conduit. The disk unit is in turn connected to the PCU via a flexible power cord. The −48 Vdc input for the PCU is connectorized at the top of the frame. However, the DFI −48 Vdc input is routed directly to a power-distribution frame due to the wire gauge and currents involved.

The disk drive is rolled onto and secured to a metal base plate fastened in the base of the MHD frame. The drive protrudes out the front of the frame 18 inches. The base plate is hinged and is raised or lowered by jackscrews on both sides of the base plate.

The I/O cables between disks and the DFC are routed in a rectangular duct at the rear of the MHD frame. The duct serves to provide additional electromagnetic shielding and physical protection for the cables. In multiple, adjacent MHD frame applications, the cable ducts form a continuous trough for the I/O cables.

2.4 Disk growth and floor plan

The MHD frame was designed to provide for growth in a modular fashion. Depicted in Fig. 5 is a typical floor plan for the 3B20D Processor. The Peripheral Control (PC) frame is situated between two sets of MHD frames. Each PC frame contains a DFC (Fig. 6).

The total cumulative control cable bus length between a DFC and all disks controlled by the DFC is 100 cable feet. Thus, all MHDs controlled by a particular DFC are placed adjacent to each other so as not to exceed the 100-foot limit.

2.5 3B20D disk file controller

The 3B20D Disk File Controller (DFC) is a high-performance, microprogrammed processor whose design provides efficient control of

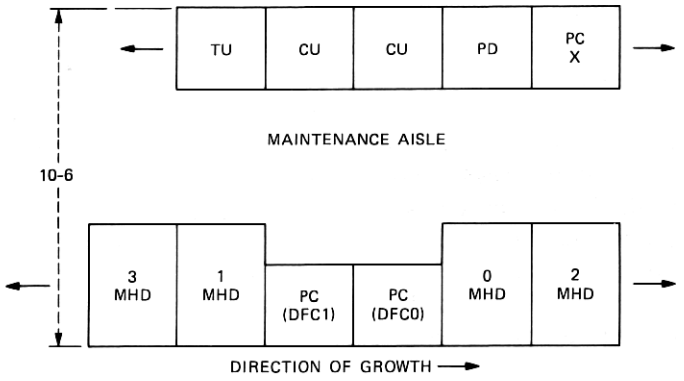

Fig. 5—Typical floor plan of the 3B20D Processor.

MHD drives and data transfers between the 3B20D Control Unit (CU) and MHD units. Each DFC supports a maximum of eight MHD units and utilizes a standard 3B20D input/output peripheral interface. The following sections discuss the hardware architecture and functionality of the DFC.

2.5.1 3B20D DFC architecture

2.5.1.1 DFC communications bus. A DFC with its associated MHD units—as shown in Fig. 7—consists of a processor, MHD interface, and the two circuit boards that provide the interface to the 3B20D Processor. These are linked together via a common communications bus.

The bus, which is dc coupled, provides 16 data bits plus 2 parity bits, 5 bits for source address, 5 bits for destination address, and 1 synchronization clock line. The source/destination addresses are binary encoded allowing a maximum of 32 unique hardware registers to be addressed on the bus. The DFC's processor and the 3B20D Processor interface each utilize 8 address codes, and the remaining 16 codes are assigned to the MHD interface. Since the bus provides unique source and destination address buses, only one bus cycle, which is defined by the synchronizing clock, is required to transfer a 16-bit data word from one hardware register to another. All data transfers on this bus are initiated and controlled by the DFC's processor. The period of the synchronizing clock may be 150, 200, 250, or 300 ns and is dependent on the instruction being executed by the processor, the capabilities of the hardware, and the timing requirements of specific firmware routines.

2.5.1.2 DFC/MHD interface. Each DFC will support a maximum of eight MHD units equipped with an SMD interface. As depicted in Fig. 7, each MHD is connected to the DFC via two cables, one of which is common to all MHDs and is daisy chained from one unit to the next.

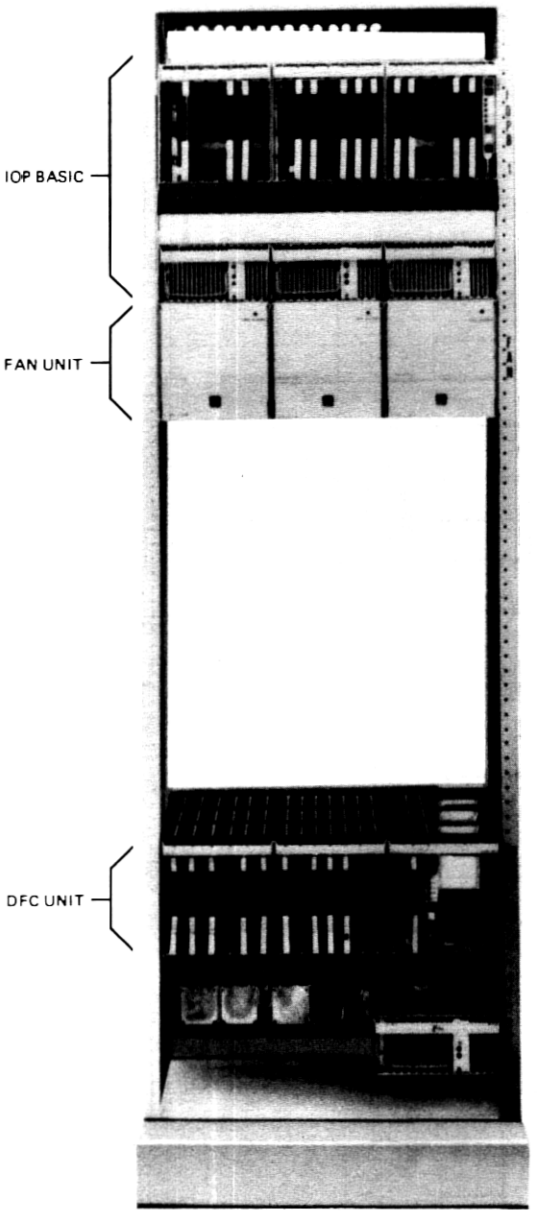

Fig. 6—Peripheral control frame.

This cable provides a communication path between the DFC and MHDs for command and status information. The transfer of data between the MHDs and the DFC occurs serially over a private path provided by the other cable. The data transferred is in a Non-Return

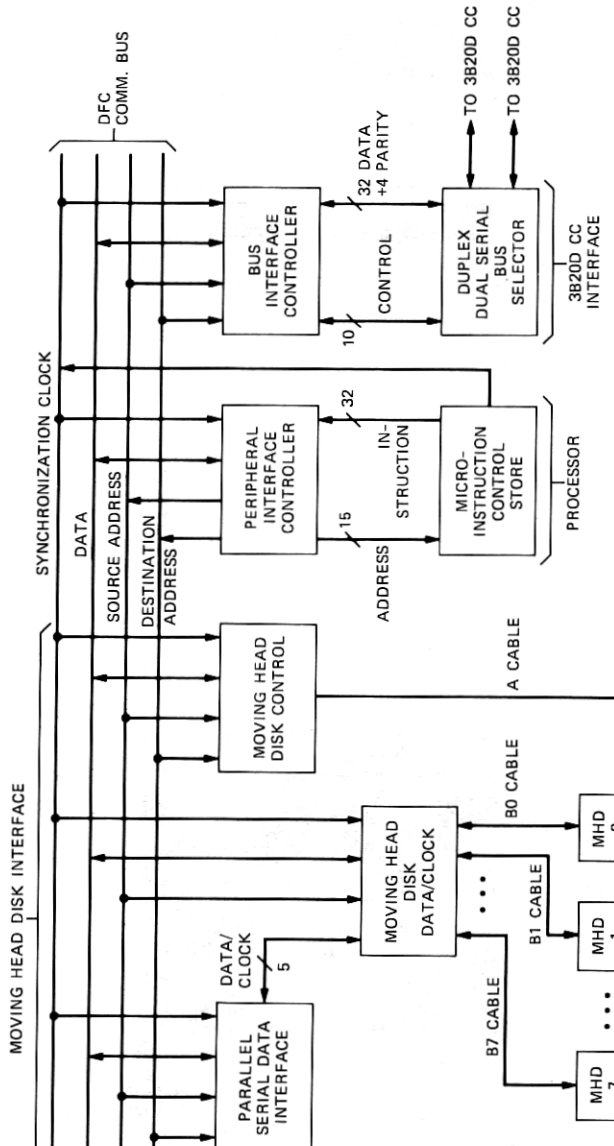

Fig. 7—Disk system block diagram.

to Zero (NRZ) format and a synchronous bit clock accompanies the data to simplify the circuit requirements within the DFC and MHD.

Because the MHD units may be located up to 100 cable feet from the DFC, signaling paths in both cables are implemented as twisted pairs, and both the DFC and MHDs interface with the cable using differential line drivers and receivers.

2.5.1.3 DFC/3B20D control unit interface. The DFC communicates with the 3B20D CU via a DSCH interface consisting of the Duplex Dual Serial Bus Selector(DDSBS) in the DFC and DSCH in the 3B20D CU. The communications path between the DDSBS and the DSCH consists of five twisted-pair dc-coupled signaling paths. The pairs are divided into two bidirectional data pairs, two unidirectional clock pairs, and a single pair used to transmit service requests to the DMAC and interrupt requests to the 3B20D CU.

Using the clock and data pairs, the minimum amount of information that is transferred is 32 data plus 4 parity bits. The data is transmitted serially on the two data pairs (16 data plus 2 parity on each pair) in an NRZ format. A synchronous bit clock is transmitted on the appropriate clock pair by either the DDSBS (DFC to CU transfer) or the DSCH (CU to DFC transfer). The interface does support a block transfer mode where 16 words of 32 bits may be transferred as a single block between the DFC and CU. This mode of operation is useful in minimizing the overhead associated with transferring large blocks of data between the DFC and CU.

The DFC's processor does not access the DDSBS directly, but rather has read/write access to several registers in the Bus Interface Controller (BIC). The BIC functions as a buffer between the DDSBS (32 bits) and the DFC's processor (16 bits). The BIC buffers data and commands for the processor and performs the necessary handshaking to communicate with the DDSBS.

2.5.2 DFC processor

The DFC's processor consists of the Peripheral Interface Controller (PIC) and its associated Microinstruction Control Store (MCS). The PIC is a 16-bit, bit-sliced microprocessor that is capable of performing all the common arithmetic, logic, and sequencer flow-control operations found in many 16-bit minicomputers. The PIC contains a microprogram data register, Arithmetic Logic Unit (ALU), 8K bytes of RAM that serve as a temporary data store, vectored interrupt control, and microprogram address generator.

The MCS stores instructions that the PIC executes in Programmable Read-Only Memory (PROM) devices. Each MCS circuit pack has the capability of storing a maximum of 4K microinstructions. A total of three MCS circuit packs are utilized. In addition to functioning as a

store, the MCS also generates the synchronizing clock for the PIC and DFC communication bus based on bits stored with each microinstruction.

On the rising edge of each synchronizing clock signal, the PIC generates a new microprogram address that is sent to the MCS. The MCS provides the PIC with the instruction stored at the requested address before the next synchronizing clock, at which time the PIC loads the instruction in its microprogram data register and sends a new address to the MCS. The PIC then executes the registered instruction while the next succeeding instruction is being fetched from the MCS. This pipelined operation that is provided by the microprogram data register allows a higher instruction execution rate than would otherwise be possible and contributes to a higher level of DFC performance.

Each PIC microinstruction is 40 bits wide and is divided into several fields as shown in Fig. 8. The PIC registers bits 0 through 31 in its microprogram data register while bits 32 through 35 are used by the MCS to verify parity on each instruction that is fetched, and bits 36 through 39 determine the period of the synchronizing clock generated by the MCS. Bits 0–4 and 5–9 are the source and destination address bus fields, and since these bit fields are dedicated, each PIC instruction specifies the movement of data from one register to another. Bit 12, when set, disables all hardware interrupts. The capability to disable interrupts on a per-instruction basis is required for the proper functioning of instructions that implement "JUMPS" and also allows the programmer to construct code segments that are critical from a real-time viewpoint and cannot be interrupted. Bits 10 and 11 determine how the overlay field, bits 13 through 31, are to be interpreted by the PIC.

2.5.3 MHD interface

The MHD interface consists of three circuit packs that allow the PIC to communicate with its MHD units. These circuit packs are the MHD Control (MHDC), MHD Data/Clock (MHDDC), and Parallel Serial Data Interface (PSDI).

2.5.3.1 MHD control. The MHDC consists of four registers, varying in width, whose outputs drive the control bus of an MHD with an SMD interface. These registers may be loaded and read by the PIC via the internal DFC communication bus. In addition to these registers, the

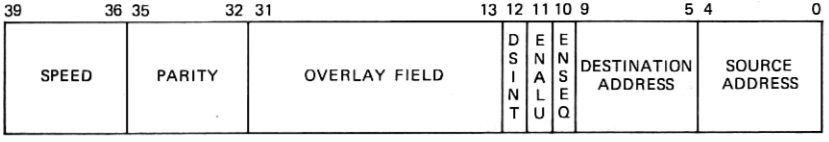

Fig. 8—Microinstruction of the peripheral interface controller.

circuit pack also provides a port that allows the PIC to sample 8 bits of status information from the MHD that is currently selected by the DFC. The output of each register drives the input of the differential line driver whose outputs drive the control cable to the MHD units. Likewise, the 8 bits of MHD status are driven onto the cable by the selected MHD unit and differential line receivers in the MHDC pack translate the differential voltage to a TTL level.

The device-enable register is a single bit register that is reset by the DFC during a power-up sequence, thus disabling the interface between the DFC and MHDs. The PIC writes this register to a one, enabling the interface, after executing its power-up microcode.

The unit-select register is a 4-bit register that contains the address of the MHD unit communicating with the PIC.

The data register is 11-bits wide and its outputs form the data portion of the control cable. This register is used with the disk tag register to transfer command information to the MHD unit consisting of cylinder number, read/write head number, and read/write/initialization commands.

The tag register's outputs form strobe signals that cause the MHD units to use information present on the data portion of the control cable. For example, bit 3 going active causes all MHDs on the control cable to compare the state of the unit-select bits with the MHD's address code. If a match occurs, then that MHD becomes selected and honors commands sent using bits 0 through 2 of the tag register. Since each MHD connected to a DFC has a unique address code, only one MHD may be actively communicating with the DFC at a given time. time.

2.5.3.2 MHD data/clock. The MHDDC interfaces the serial data and clock between the MHDs and the PSDI, and provides the PIC access to a select acknowledge signal generated by each MHD. The select acknowledge signal is made active by an MHD whenever it is selected, and by providing one acknowledge signal per MHD allows the PIC to verify that the correct disk volume has been selected for use. With respect to the serial data and clock signals, the MHDDC functions as a voltage level translator (differential signaling to TTL and vice versa) and signal multiplexer, routing the data/clock signals between the selected MHD and the PSDI.

2.5.3.3 Parallel serial data interface. The PSDI allows the PIC to transmit data to and receive data from the MHD. A block diagram of the PSDI is shown in Fig. 9.

Data exchange between a DFC and an MHD drive takes place over a 10-MHz serial data path. During a disk write, the PSDI performs parallel to serial data conversion, checks data parity as the data is being shifted out serially, and computes a 32-bit cyclic ECC that is

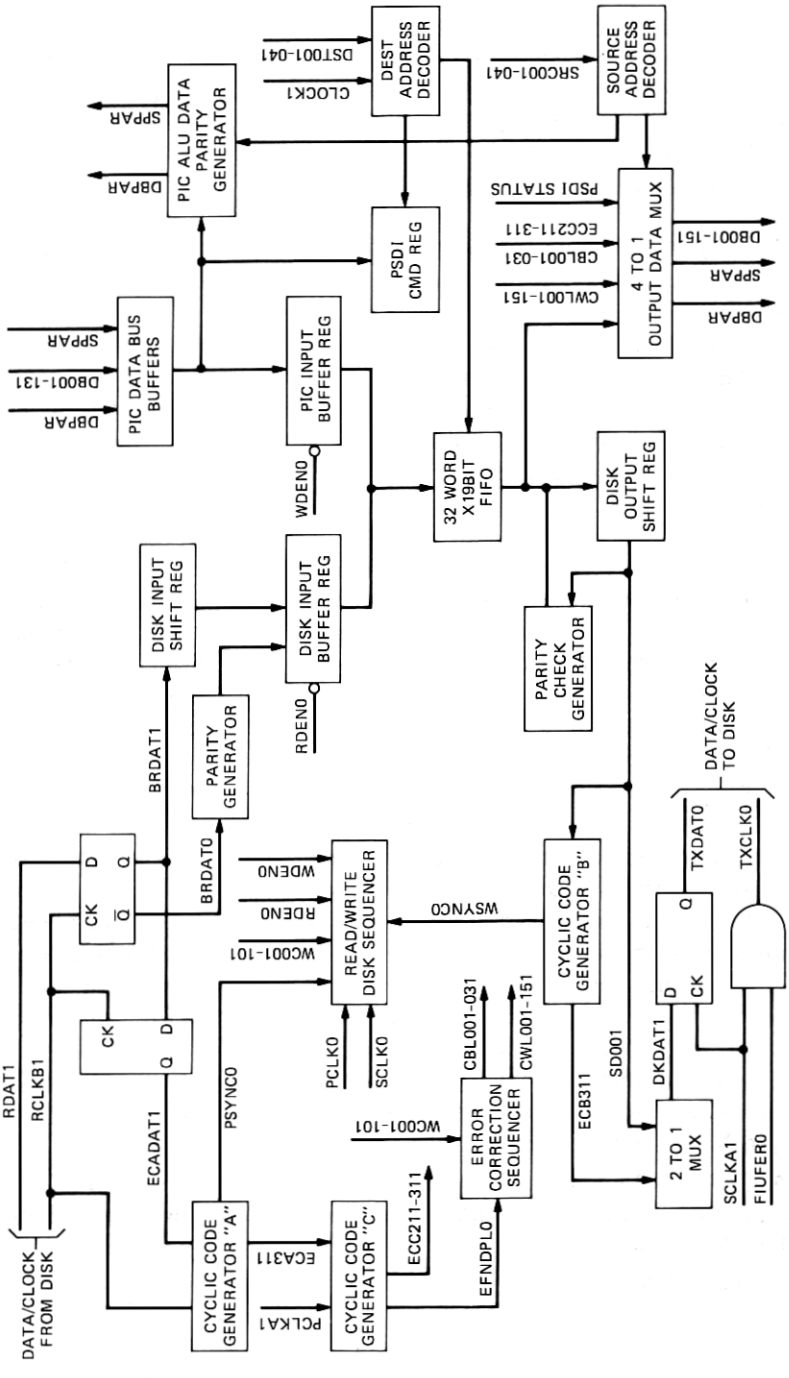

Fig. 9—Parallel/serial data interface.

also transmitted to the disk after a predefined block of data has been transmitted. During a disk read, the PSDI performs serial-to-parallel data conversion, computes data byte parity, and evaluates the ECC read from disk to determine if any errors are present in the data. While the PSDI itself does not perform the error-correction function, it does provide the PIC with the following information about each data block read from the disk: (*i*) The presence of errors, (*ii*) whether the error is correctable or uncorrectable, (*iii*) the location of the error relative to the start of the block, and (*iv*) an 11-bit correction mask that is to be exclusive ORed with the received data to affect the correction.

2.6 Disk read and write operations

2.6.1 Disk read operation

The PSDI functions as a serial data interface between the PIC and an MHD drive. During disk reads serial data and clock transmitted by the disk drive are assembled into 16-bit words in the disk input shift register, and stored in the data FIFO. Simultaneously, the serial data is also presented to the input of a parity generator where data byte parity is assigned and stored in the FIFO together with the data, and to cyclic code generator A where a 32-bit code word is computed to be utilized during the error correction search process. In addition, this cyclic code generator searches the incoming data stream for the presence of data synchronization bits. The generator searches the synchronization characters following the preamble to locate the beginning of the address/data fields in the serial data stream. Once sync is found, the read/write disk sequencer begins assembling 16-bit words and storing them in the data FIFO. The only information stored in the FIFO is the address and data bytes. Once the ECC field from disk has been input to the code generator, the contents are serially transmitted to a second code generator C. Then, to start error correction generation, the first code generator begins its search for synchronization characters again.

Cyclic code generator C and the error correction sequencer function together to determine the presence and location of an error in the received serial data stream. Code generator C is a 32-bit shift register which has several feedback paths around it. The error correction sequencer supplies shift pulses to generator C and if after each shift the contents of the low-order 21 bits of generator C are zero, the correction sequencer stops and indicates that a correctable error has been found. If they are nonzero and both the bit and word location counters have not reached their maximum counts, the sequencer will generate another shift pulse. If the 21 bits remain nonzero and both the bit and word location counters have reached their maximum

counts, the sequencer stops and indicates that an uncorrectable error is present in the data stream.

An uncorrectable data error is one where the first and last data bits in error are separated by more than nine consecutive bits. A correctable data error is one where the first and last data bits in error are separated by nine or less consecutive bits. The first case of no error is distinguished from the error case by the value of the upper-order 11 bits for code generator C. With no bits in error, these 11 bits will all be zero.

As mentioned earlier, the PSDI does not perform the error correction function, but does provide the PIC access to the information necessary to perform this task. The correction mask code generator C high-order 11 bits, the bit location counter, word location counter, and correction sequencer status (correctable or uncorrectable data error) are made available to the PIC through the output data multiplexer whose outputs are the PIC data bus. The error correction code contained in each section of the disk is utilized together with code generators A and C, and the error correction sequencer are designed to detect and locate errors in the address, data, and the error correction code fields recorded on disk.

2.6.2 Disk write operation

When writing the disk, information contained in the FIFO is loaded into the disk output shift register where it is serialized and transmitted to the disk drive under the control of the read/write disk sequencer. Simultaneously, the serial data stream is applied to the input of the parity check generator and cyclic code generator B. Although each data word in the FIFO has two parity bits associated with it, they are not transmitted to the disk, but are loaded into the parity check generator each time the output shift register is loaded from FIFO. In addition to the parity bits, each data word also has associated with it a parity check control bit, which is loaded into the parity check generator during the FIFO to output shift register transfer. The parity check control bit either enables or disables the parity generator on a per-word basis allowing words with known bad parity to be transmitted without causing parity errors.

Cyclic code generator B computes the 32-bit ECC field to be written on disk based on information transmitted in the address and data fields. Since writing a sector on disk requires all fields shown in Fig. 6 to be written, code generator B also searches for the synchronization characters. When sync is found, the read/write sequencer configures code generator B to operate as a feedback shift register to compute the ECC field for the address and data fields that follow the sync characters. In addition, at sync time, the read/write sequencer begins counting the words it loads into the output shift register from the

FIFO. After the last data word has been serially transmitted from the output shift register, the read/write sequencer disables all the code generator B feedback paths, enables the contents of the generator to be shifted out to the disk, and then redirects the serial stream back to the FIFO/output shift register to transmit the post-amble characters. Once the ECC field is transmitted, both the read/write sequencer and code generator B revert back to their initial states searching for synchronization characters.

For diagnostic purposes it is useful to be able to write an invalid ECC field when writing a sector on disk. Setting a bit in the PSDI command/status register prevents the read/write sequencer from transmitting the contents of code generator B to the disk. Instead, the read/write sequencer continues to transmit information to the disk contained in the FIFO. This scheme allows any two 16-bit words to be written in the ECC field in place of the normally computed value provided by code generator B.

2.6.3 Data FIFO

A 32-word FIFO memory is loaded by the read/write sequencer and read by the PIC during a disk read operation, with the reverse being true during a disk write operation. Each word in the FIFO is 19 bits wide: 16 data, 2 parity, and 1 parity check enable bit. As explained earlier the parity check enable bit is significant only during a disk write operation. FIFO write access is controlled by the mode bit in the PSDI status register. Depending on the state of this mode bit, data is written into the FIFO by the read/write sequencer or by the PIC. Because the FIFO requires that valid data be present for 70 ns after the write pulse makes its low-to-high transition, two input data buffer registers are provided for use by the read/write sequencer and PIC.

Associated with the FIFO are four status flags. The FIFO ready and FIFO half full/empty flags provide the PIC information to load or unload the FIFO properly and efficiently. When the PSDI is in the read mode, FIFO ready low/high controls whether the PIC may or may not read a word from the FIFO. Likewise, when FIFO half full/empty occurs, it allows or prevents the PIC from reading 16 words from the FIFO.

In write mode, these flags function similarly with FIFO ready low, indicating that the PIC may load a single word in the FIFO, and FIFO half empty low, indicating that the PIC may load 16 words in the FIFO. In addition to these FIFO status indicators, FIFO underflow and overflow error flags have been provided. A FIFO underflow error occurs whenever the read/write sequencer or PIC reads data from the FIFO when the FIFO is empty or valid data is not present at the bottom of the FIFO. FIFO overflow occurs when either the read/write

sequencer or PIC writes the FIFO when it is already full or data in the top memory location has not rippled down to a lower memory cell. Both of these errors are latched into individual flip-flops that can only be cleared when the PSDI is master cleared.

2.7 DFC firmware structure

2.7.1 Introduction

The disk file controller firmware provides the interface between the 3B20D Processor and the moving head disk for storage and retrieval of information. The firmware can make certain decisions of its own regarding the integrity of the MHDs or of itself, but will not take any actions on data content. It has a routine diagnostic exercise to check itself and a disk exerciser to check disk data format integrity. It also has sanity timers on data transfers to/from the processor and timers on disk accesses. Data is transferred between the DFC and the processor at an average rate of 1 megabyte per second. The DFC can handle up to 8 MHDs on a single controller. There are two job-processing options that may be invoked by the 3B20D. The DFC will process jobs on either a first-in/first-out basis or by a modified elevator algorithm. The elevator algorithm is used to keep the number of long head seeks to a minimum. There is also an autoread retry to help in recovering marginal data from the media.

The firmware will do as much as possible to guarantee the data integrity on a disk pack and will not allow data with bad parity to be written either to the disk or the 3B20D. The DFC will try, as much as possible, to perform 'overlapped' seeks. That is, if there are two or more requests, each to a different disk, the DFC will issue a seek to each disk before trying to do the job on any one of the disks. Thus, if a seek requires a long time to complete, and another disk has completed its seek, then the job for the disk which has completed its seek will be done.

To assure the operational capabilities of a disk, a disk exerciser is run periodically. The disk exerciser assures that the data within a given area of the disk is usable. By sequentially addressing various areas of the disk with the exerciser, eventually all areas of the disk will be verified for data integrity.

2.7.2 Timing considerations

The time it takes to get or put data on a disk file is determined primarily by how far away the data block is from the current position of the disk heads. The majority of disk accesses are less than 20 blocks in length, thus the rotational latency and seek times of the disk file become the determining factors in data transfer. In the optimal case, there is about 2 milliseconds of delay from the time the disk file

controller gets the command until it is through transferring the data. This is the sum of verifying the disk head location, doing error correction, and reading the requested block from a disk back to the processor. The worst case, assuming no DMA contention problems, is 74 milliseconds: the sum of the maximum seek time plus one rotation of the disk. The average time is 41 milliseconds.

2.7.3 Firmware tables

The DFC maintains an orderly record of data within the controller by using six tables: SYSGEN data table, circular queue table, Active Job Table (AJT), read data buffers, write data buffers, and the disk verify data table.

SYSGEN is a programmed input/output command received by the DFC. This command contains the address in 3B20D main memory of the SYSGEN table. The SYSGEN data table contains information for the DFC. The information includes the:

unload pointer
load pointer
disk driver queue address (virtual)
maximum cylinder allowed
optimization data
maximum number of jobs in disk driver's queue.

The job queue is an image of the data from the disk drivers queue. There are 16 bytes in each disk job description. There can be a maximum of 64 such jobs in the DFC at any one time.

The AJT has a series of entries to describe to the firmware the status of a particular drive. There is one such entry for each disk allowed on the controller. There are 32 words associated with each disk entry in the AJT. Some of the more important items in the list are: command, current disk maintenance state, current cylinder number, and where on the disk to start the job.

There are eight buffers set aside for the storage of data to be transferred to or from the disk. Each buffer is 263 words (16 bits each). There are seven status words and 256 data words in each buffer.

The disk verify routine checks the validity of data on the disk. This is done by reading a sector and determining if the header and data are correct. If they are not, then that disk block number is recorded in the disk verify buffer. Up to 64 such bad blocks are allowed before the verify routine will terminate.

III. SUMMARY

The 3B20D file memory system provides a maximum formatted storage capacity of 2,030 megabytes of storage on eight 300-megabyte

disk drives. The disks are managed by a disk file controller using a high-speed bit-sliced microprocessor, which utilizes 30,000 bytes of microcode to asynchronously process data transfers to and from the 3B20D main store via DMA. The disk controller interface to the 3B20D processor is via the high-speed dual-serial channel.

IV. ACKNOWLEDGMENTS

Acknowledgment is made to R. D. Adams, C. F. Ault, T. J. Clarey, R. A. Ehle, R. M. Gagliardi, T. S. Greenwood, D. H. Leonard, P. D. Olson, and E. M. Schaefer for their design contributions; and to R. C. Hatfield and N. C. Zielke for their supporting activities.

REFERENCES

1. M. W. Rolund, J. T. Beckett, and D. A. Harms, "3B20D Central Processing Unit," BSTJ, this issue.
2. J. L. Quinn and F. M. Goetz, "Deferrable Maintenance and Diagnostics," BSTJ, this issue.
3. M. E. Grzelakowski, J. H. Campbell, and M. R. Dubman "DMERT Operating System," BSTJ, this issue.