THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 53, No. 1, January 1974
Printed in U.S.A.

Automatic Intercept System:

File Subsystem

By J. W. HOPKINS, P. D. HUNTER, R. E. MACHOL, J. J. DiSALVO, and R. J. PIERETH

(Manuscript received June 8, 1973)

The Automatic Intercept System must maintain a large data base from which information can be obtained pertaining to telephone numbers on intercept in the area. The file complex provides the mass storage medium necessary for this system. This article describes this subsystem, its operation, and how it is maintained.

I. INTRODUCTION

The Automatic Intercept System (AIS) must maintain a large data base from which information can be obtained pertaining to the hundreds of thousands of telephone numbers on intercept in an area. The file complex, consisting of a disc file and associated controller, provides the mass storage medium and data access mechanism for the AIS.

In establishing such a data base, two important points were considered. First, the file complex must be highly reliable. Second, the frequency of calls and the changeable nature of the data require that the data base be readily accessible and easily alterable.

In selecting a disc file it was determined that the disc file used by No. 1 ESS ADF would adequately fulfill AIS requirements. In addition, it would be possible to make use of much of the head accessing and motor drive circuitry already developed for No. 1 ESS ADF disc.[1]

High reliability is provided both through a coordinated hardware-software design which includes many hardware and software error detection and location techniques, and by duplicating the entire file complex—each file containing identical data.

Data interchange between the system control units and the two files is accomplished independently over the peripheral bus system under control of signals from the central pulse distributor (CPD).

Since the file subsystem is comprised of two identical units, much of the following hardware and software description considers just a single unit.

II. HARDWARE DESCRIPTION—FILE CONTROL

2.1 General

The file controller is a wired logic machine consisting of several synchronous sequencers and register circuits. The sequencers are individually associated with major file functions. Collectively, they provide the ability to retrieve, store, and check the data on the discs.

The file circuitry is composed of discrete-component, diode-transistor-logic (DTL) circuit packs, originally designed for the No. 1 ESS.[2] The choice of this type of logic provided a wide variety of available circuits, compatible disc-logic speed, and a favorable cost factor.

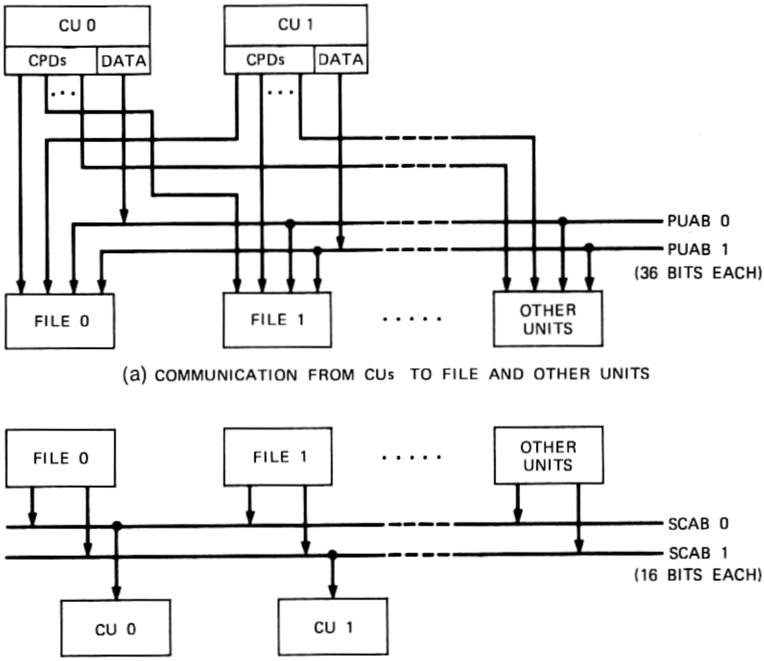
In the AIS, the file subsystem has duplicated files and each file operates independently of the other, allowing each to handle separate operations simultaneously. The data base on each disc, however, is the same with the exception of the recorded "call-counts" (the number of times the entry has been referenced) on the individual entries. This configuration provides both reliability and higher call throughput. If one file is out-of-service, the remaining file has the ability to handle all call processing requirements with minimum effect on service.

Although a file complex is a synchronous machine (using clocks recorded on the disc), it is independent of the No. 2 ESS Control Unit (CU) and performs most of its actions autonomously. It requires only the initial data and an instruction to perform a particular function such as looking up the status of an intercepted number. This enables the CU to perform other tasks while lookups and other functions are performed by the files.

2.2 Control Unit—file communications

A Control Unit communicates with a file over a 36-bit, ac, Peripheral Unit Address Bus (PUAB) and a 16-bit Scan Answer Bus (SCAB), as shown in Fig. 1. Each CU has its own PUAB and SCAB but each file connects to both sets of busses, allowing either CU to communicate with either file.

A file receives orders and data from the online CU via its PUAB and returns data to both CUs over the SCABs. Instructions either to



(b) DATA RETURN PATHS FROM FILES AND OTHER UNITS TO CUs

Fig. 1—CU communications paths.

accept data from a PUAB or to gate data onto the SCABs are received over separate leads termed CPD leads. Unlike the PUAB and SCAB busses which multiple to all peripheral units, separate CPD leads are assigned to each file.

2.3 Timing circuit

Timing in the file complex is provided by the combined use of three clock tracks recorded on the discs (see Section 3.3). One track is written as a single pulse, called track index, which is used to define the start of each revolution. The other two clock tracks are written as a pair. The first clock track of this pair defines the beginning of each bit. It is called the bit clock. The second clock track of the pair defines which bit within a word is present at any particular instant. It contains a pseudo-random pattern 46 bits in length which is repeated 1600 times around the track (a track contains sixteen hundred 46-bit words). This pattern is unique in every set of six consecutive bit positions and is, therefore, decodable into a 46-phase clock. The pattern is read from the disc into an open-ended shift register with the clock decoding circuit using the contents of the register to produce the proper bit time indication. Use of seven stages of register instead of six allows instant detection of an error in the reading of the pattern from the disc by providing a simple error detecting circuit. The bit clock provides a shifting signal for the shift register and is used for strobing data read from the disc.

This pair of clocks provides positional information within each word, while the word counter (Section 2.4.1) provides positional information with respect to the beginning of a disc revolution.

2.4 Register circuits

A number of special-purpose registers are provided to perform various file operations. A brief description of some of these is given in order to suggest the types of information needed to handle data on the file.

2.4.1 Word count register

The word count register has two separate sections, a counter and a register. The counter, which counts to 1600, is set to 0 at the beginning of each revolution of the disc and is incremented by one phase of the 46-phase clock. The associated 12-bit word count register can be loaded in parallel from either the word counter or directly by the CU. The contents of the counter and the register can be compared in a match circuit which will indicate when the counts are the same.

During an associative lookup (see Section V), the contents of the word counter section is gated into the register section to store the actual location of the called number word that was found. On the subsequent disc revolution the match circuit locates this word for call count rewrite. For block read or write operations, the CU loads the word count register with a location in the block ahead of the desired block to be processed. This enables the file to interrupt the CU when the disc is in the proper position, thus allowing the CU to be freed for other processing during the waiting period.

2.4.2 Called number register

The called number register is a parallel input, serial output 30-bit flip-flop register. During a lookup sequence it is loaded by the CU over the PUAB with the intercepted number to be looked up. During a search sequence this register is read out serially and matched against data read from the disc file. The serial output of this register is also used in rewriting a called number on the disc during call count rewriting.

2.4.3 Input-output register

The input-output (IO) register acts as a buffer between the CU and the file. When the IO register is used in lookup functions, the status (reason for intercept) of the intercepted number and new number, if one exists, are loaded into it serially from the disc. In block read and write operations, the IO register is used to buffer the word being transferred to or from the disc.

2.4.4 File status register

The 16-bit file status register is used to store certain indications for use by the CU, such as progress of file operations, file availability, and errors that might be encountered during a file operation.

2.5 Correspondence control circuit

This circuit consists of a group of subcircuits which perform the following functions:

(*i*) Control the associative lookup sequence.
(*ii*) Check for a serial match between data in the called number register and data read from the disc file.
(*iii*) Control block read and write functions.
(*iv*) Perform miscellaneous data checking and gating functions.

The combined use of these circuits is described in Section V.

2.6 Maintenance

The major portion of the maintenance circuit consists of a set of "crosspoints" (see Section VII) that can be accessed by the CU to determine the states of circuits internal to the file. The remaining portions of the maintenance subcircuits perform the following functions:

(*i*) Detect when the voltage output of any dc-dc converter in the file complex is incorrect.
(*ii*) Check the integrity of the timing circuits.
(*iii*) Check for proper cycling of the word counter.

III. HARDWARE DESCRIPTION—DISC MEMORY

3.1 Physical description

The disc file used in AIS is a modified Burroughs Corporation model BC475 disc file memory module. The BC475 is a head-per-track file using fixed air-bearing heads. The use of one head per track eliminates the need for head positioning. Two models have been developed for use in AIS, one with a storage capacity of 14.6 megabits, and the other with a capacity of 29.2 megabits. The lower-capacity file is equipped with two discs while the larger has four. Both models are equipped with the outer zone heads only, or one-third the heads normally supplied in the BC475. Pertinent characteristics of these memories are shown in Table I. The two-disc model omits the discs, heads, and other parts within the right-half enclosure.

Recording on the nickel-cobalt-coated discs is accomplished using the nonreturn-to-zero (NRZ) recording method (see Section 4.1). The read/write transducers, or "heads," are contained in assemblies of 13 heads each. When the discs are not rotating, the head assemblies are held retracted from the disc surface by head mounting springs. During operation, pneumatically driven pistons force the assemblies toward the disc surface. Opposing this action are the force of the spring and the force of the moving layer of air developed by the spinning disc surfaces. By applying a calibrated amount of gas* pressure to the pistons, the heads are forced toward the disc surface until they reach the point where all forces balance out. Thus, the heads literally "fly." In the KS-20512 unit, the head-to-disc spacing is approximately 75 micro-inches. Because of this close spacing, the discs are enclosed in an air-tight dust-free cover to prevent head instability problems that can be caused by dirt particles.

3.2 Data track accessing

In the disc file, only one data track can be accessed at a time. The 1-out-of-200/400-track selection is accomplished through the use of head switching diodes contained within the disc file. The information heads are grouped in sets of 100 heads per disc, 50 heads associated with each disc surface. The output leads from each set are connected to a head control circuit. The center taps of the 100 heads in each set connect in parallel to the 100 select lines in each head set. Selecting a head is accomplished by enabling one of the 100 head select lines

---

* AIS provides the option of using either nitrogen gas as in the No. 1 ESS ADF or air from a frame-mounted air compressor.

Table I — Disc memory characteristics

KS-20512	L1	L2
Disc Speed	1500 rpm	1500 rpm
Data Packing Density	1000 bits/inch	1000 bits/inch
Storage Capacity	14.6 megabits	29.2 megabits
Number of Discs	2	4
Number of Data Tracks	200	400
Number of Clock Tracks	8	16
Track for Track Index	1	2 (1-not used)
Cooling Fan	1	1

and one of the 4 head control circuits. With a center tap enabled and a head control enabled, the diodes in the desired head (1 out of 200/400) are forward biased, rendering the head active. For reading, the head control circuit connects the head to the read amplifier. For writing, the head is disconnected from the amplifier and a regulated current of 125 ma is switched to the head by the write switches in the head control circuit.

3.3 Clock head accessing

Each disc surface has a bit clock and a pattern clock recorded on it. These clocks are used to derive timing pulses used throughout the file complex (see Section 2.3). To minimize timing shifts between data and clock, the clock tracks selected are on the same surface as the data track being accessed. There is one center tap line for each pair of bit clock and pattern clock heads. The outputs of the bit and pattern heads on one surface of each disc are connected in parallel to amplifiers, one for pattern clock and one for bit clock. Similarly, the clock heads on the opposite surfaces are connected to two additional amplifiers. Enabling the appropriate center tap line forward biases the clock head switching diodes, connecting the head through to the amplifier. Two center tap lines are always activated, one to select clocks for timing purposes, and the other to provide bit clock for the disc frequency servo circuit.

The pair of timing tracks which are recorded on each surface consists of a bit rate clock and a unique 46-bit pattern. The bit rate clock is a rectangular wave having 73,600 cycles per disc revolution. The disc speed is nominally 1500 rpm, hence the bit rate clock is 1.84 megabits/second. The pattern is a pseudo-random arrangement of ones and zeros which is unique within any group of six adjacent bits (see Fig. 2).

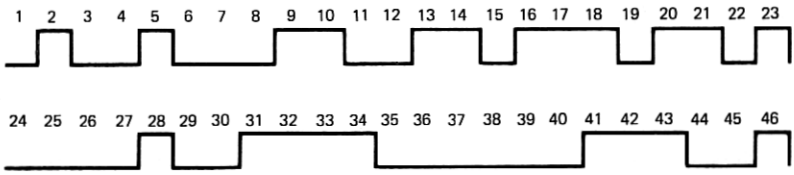


Fig. 2—Pseudo-random pattern for bit times.

Positional information is not contained in either the bit clock or the pattern clock. To provide this information a separate track containing a single "one" bit is used. This bit, called track index, defines the start of a disc revolution, and is used for many purposes in the file circuitry. The index track, bit clock, and pattern clock tracks are all written prior to use of the disc module in the system. Once the disc is in the system, these tracks can only be rewritten by special means.

3.4 Disc drive and servo

The discs are driven by a 2-hp, 208-volt, 3-phase, 60-Hz synchronous motor which is belt-coupled to the disc shaft. To insure the existence of a reliable power source that can be frequency controlled, ac power for the motor is converted from the normal office dc supplies by circuitry contained entirely within the file frames. A 360-Hz voltage-controlled oscillator, the frequency output of which is divided by a counter, supplies the required drive voltage. The voltage is transformer-coupled to the disc motor.

The disc speed is controlled by a frequency servo system to meet tight speed tolerances (the circuitry used is essentially a portion of that used in No. 1 ESS ADF).[1] The frequency servo circuit compares the bit clock frequency read from the disc to the frequency of a crystal-controlled reference oscillator and develops an error voltage proportional to the frequency difference. This voltage controls the frequency of the 360-Hz oscillator which, in turn, controls the speed of the drive motor. As a result, the disc speed is held to within ± 0.1 percent.

3.5 Disc temperature control

As with most telephone switching equipment, the file complex must be capable of operating over a temperature range of 35°F to 120°F. The disc file, however, should not be subjected to temperatures below 60°F mainly because of the shaft bearings. To overcome this limitation, the disc unit is enclosed in a temperature-controlled cabinet equipped with motor-controlled louvers and cooling fans. The operating range within the cabinet of 85°F to 120°F also serves to minimize shifts in data timing due to temperature effects.

3.6 Disc-related maintenance and protection

Numerous safeguards and checking features are built into the circuitry related to disc operation. The magnitude and duration of write current, proper switching of current between the 0 and 1 head windings, and correct head selection are a few of the more important checks performed. Failures detected by these monitoring circuits will result in the disabling of further head selection to prevent destruction of data. In addition to monitoring the data heads for multiple head selects, the clock head center taps are monitored for double enabling. Failures in clock output gating or in clock readout are detected by the bit clock checker which is capable of detecting the dropping of a single bit.

In the disc motor drive and servo areas, extensive checking is done both during motor startup and normal operation. Such factors as motor drive current, correct phase production, and servo operation are continually monitored. Certain failures will cause automatic power shutdown while others will result in the requesting of maintenance diagnostic programs.

Due to the nature of flying head disc memories it is possible under certain conditions for the heads to touch the disc surface causing damage. Very often damage to the surface can be prevented by detecting the presence of conductive dirt particles between the head and disc. This is accomplished by means of "touch" probes imbedded in each head assembly and a common touch detection circuit. The detection of a touch will cause automatic retraction of the heads from the discs.

IV. DATA ORGANIZATION

4.1 General

Intercept information is recorded on the disc serially by bit in binary-coded-decimal form. NRZ (nonreturn-to-zero) recording is used in which a change in magnetic flux occurs only when the logic level changes from one to zero or from zero to one. Data words consist of 42 data bits, 1 parity bit, and 3 guard bits. Three guard bits are used to allow single word alteration without data overlap problems.

Both surfaces of a disc are used for recording, with each surface divided into 50 tracks, all of which are located on the outermost "zone" of the disc surface. Thus, a four-disc unit has 400 data tracks. However, only 48 tracks per surface or a total of 384 tracks are used to store the actual intercept information. Of the remaining tracks, three contain "locator" information (indexing words) which expedite the associative number search; one track contains the nongeneric (installation-dependent) office parameters; and seven tracks are used by file maintenance programs for testing read/write circuitry. The rest are unassigned.

4.2 Word addressing

The disc surface is partitioned into 16 pie-shaped segments termed "sectors." Each sector contains 100 words divided into five groups of 20 words termed "blocks." A block is written in a five-word interlaced

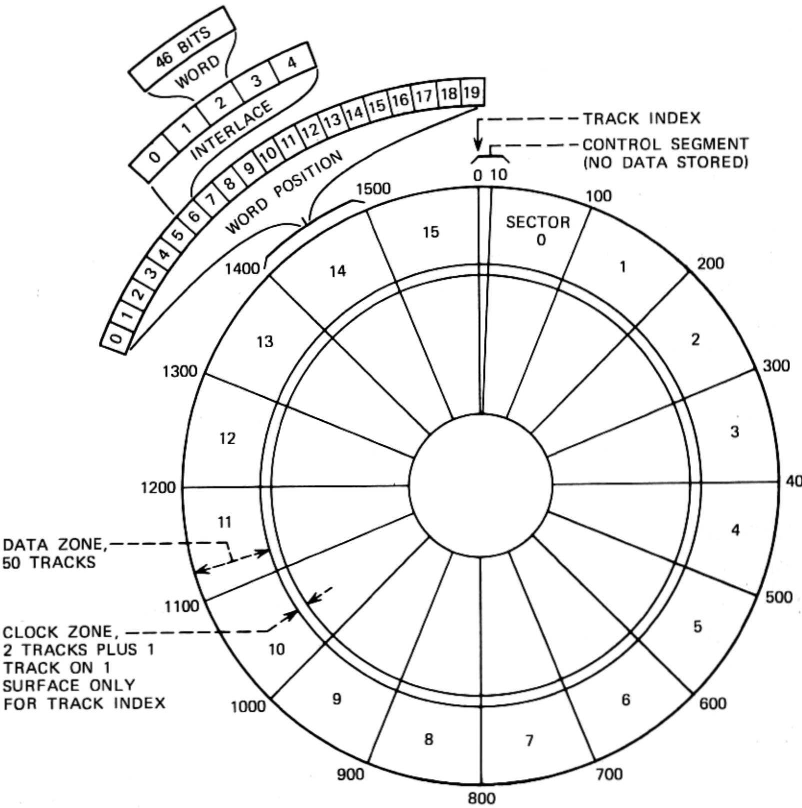


Fig. 3—Organization of data storage facilities on each disc face.

fashion where every fifth word belongs to a given block. However, data blocks are read or written as if the words were consecutive. The data are arranged in this way to match the data handling speeds of the file and control unit. The overall disc layout showing track locations, sectors, and interlaces is illustrated in Fig. 3.

Based on this data organization, a word location or address is defined by its disc (0–3), surface (0–1), track (00–49), interlace (0–4), sector (00–15), and word position (00–19). The transition from sector 15 to sector 00 represents the end of one disc revolution and the beginning of the next. This transition point is defined by the track index pulse, which sets the word count register to all 0's. Since clock and data head switching also take place at this time, the first two word positions of each interlace in sector 00 are not used to allow time for the head switching transients to subside.

4.3 Types of intercept record words

Intercept data words recorded on the disc fall into one of three categories: header or locator words, called or intercepted number words, and new number words. The formats for these three types of words are shown in Fig. 4. The tag on each word allows easy identification of the different types of words. The last bit of each word is odd parity computed over the entire word.

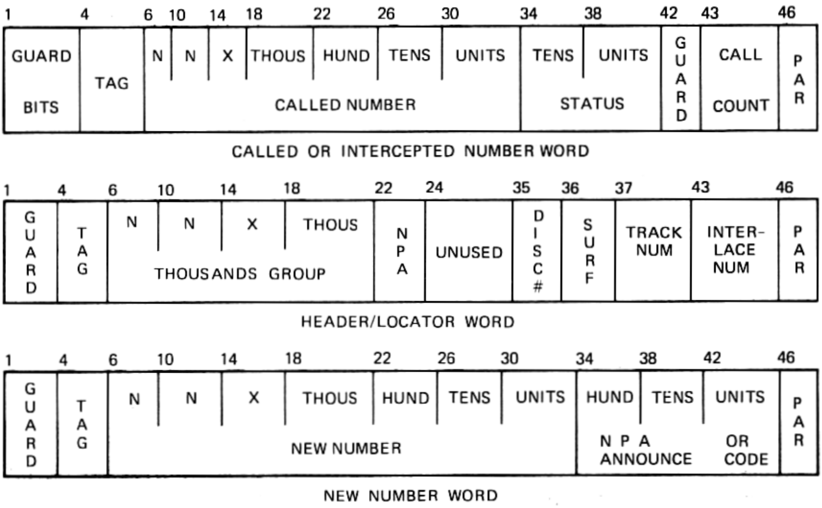


Fig. 4—Word formats.

4.4 Nongeneric data

Nongeneric (office-dependent) data stored in the CU call store memory are also stored on the disc file. This redundant storage serves two purposes. First, it provides a reference for audit programs to verify that the semipermanent call store data have not been changed erroneously. Second, in the event of a high-level system initialization, it is used in restoring the call store nongeneric data areas. These data contain information such as trunk assignments, announcement machine track assignments, lengths of queues, and type of AIS office (two- or four-disc, one- or two-network, etc.).

V. MODES OF OPERATION

5.1 General

The file has two modes of operation. The first is the associative lookup mode which is used for two purposes: (*i*) intercept number lookup by call processing programs; (*ii*) location of disc machine address information for use by file administration programs when inserting or deleting entries. The second is the block transfer mode whereby a block of data can be read or written under control of the CU. These operations are described in greater detail in the following sections.

5.2 Associative lookup mode

The associative lookup is initiated by programs in the CU passing to the File Control the seven-digit intercept number, the Numbering Plan Area (NPA) code of the intercept number, and a request code identifying the operation. The hardware sequencer then proceeds to control the lookup process using the intercept number as an index into the disc data base (see Fig. 5).

The sequencer selects a locator track at the start of the first revolution after the receipt of the command. This allows the reading of locator words, which consist of: (*i*) the first four digits of the intercept number and NPA code, and (*ii*) the track address of the data track where this grouping of numbers is stored. These locator words are compared serially against the corresponding portion of the intercept number being searched for and, when a match occurs, the track address portion of the locator word is loaded into a register.

At the start of the next disc revolution, the register containing the track address information is referenced by the track selection circuitry and this track is selected for reading. The sequencer directs the com-

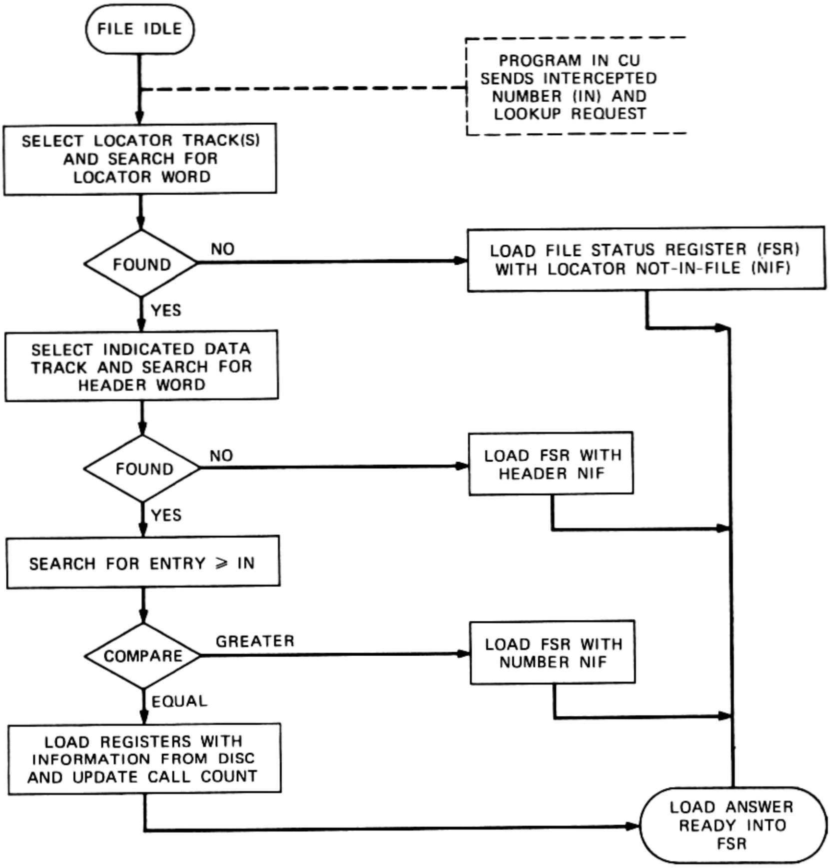


FIG. 5—Flowchart of intercept number lookup.

parison circuit to search for an entry on the data track which is identical to the locator word. This word is called a header word since it is the heading for all intercept numbers with the same first four digits (called thousands groups). When the header word is found, the sequencer begins searching the following data, comparing it with the full intercept number. The intercept number entries are sequentially ordered within each thousands group, allowing the comparator to search for either a match or a "greater-than" entry.

The match of the intercept number results in the following steps: (*i*) the status and machine location of the intercept number are loaded into registers for retrieval by the programs initiating the search; (*ii*) the sequencer waits for one revolution to rewrite the entry with a new call count (the call count is part of the entry indicating the number of times this number was referenced); (*iii*) if the next entry is a referral number (termed new numbers), it is also loaded into a register.

If a "greater-than" entry is found, the sequencer takes a different course of action. The machine location of the larger entry is stored away for use by file administration programs and the file status register is loaded with a "not-in-file" indication for the call processing program.

Upon finding either a match or larger entry, the file status register is loaded with an "answer ready" indication telling the requesting program that the action is complete. This indication is also set should the sequencer fail to find either the locator or header words.

The total associative lookup procedure requires from 80 to 520 milliseconds to be completed, depending upon point of entry, fill of data, etc. This is the most predominately used operation of the file.

5.3 Block transfer mode

The block transfer mode provides direct access to blocks of data on the disc. In this mode a block of disc words is either written or read in conjunction with a program operating in the CU. In order to accomplish this, the program (operating in the CU) and the file control (reading or writing on the disc) must establish synchronism. This is accomplished through the use of an interrupt structure in the CU.

The sequence (Fig. 6) is started when a program requests a block transfer, passing to the file the block address (track, sector, and interlace numbers) and the function request code.

The sequencer waits until the start of the revolution following the request, at which time the proper track is selected. The sequencer then compares the disc position with the address of the desired block. When the disc reaches the address, the file control sends the CU an interrupt signal. This calls in the program to transfer the block. This program reads the position of the disc and establishes synchronism with it. At the same time, the sequencer in the file circuit prepares to transfer the first word at the beginning of the block.

The block is transferred, utilizing the time provided by the interlaced data arrangement on the file for data manipulation. This manipulation includes error checking by hardware circuits in the file, transfer between CU and file, and storage in or retrieval from memory in the CU. The file reads or writes each fifth word on the disc in the selected sector, allowing 125 microseconds between successive words.

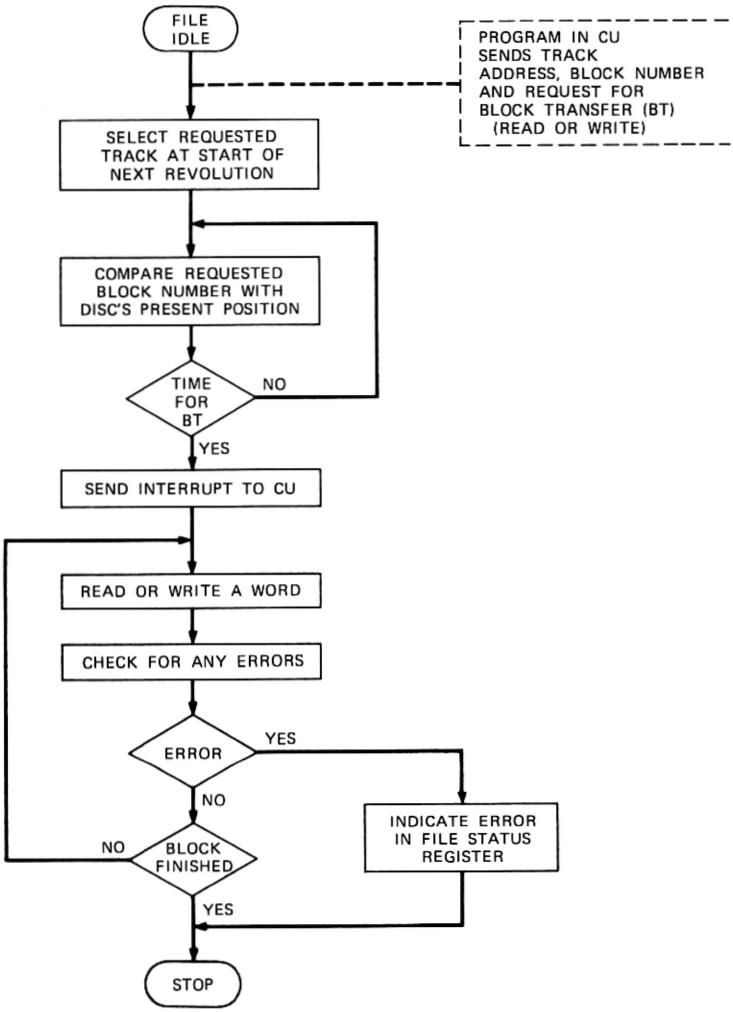


Fig. 6—Flowchart of block transfer.

When the entire block is transferred, the sequencer indicates in the file status register whether or not an error has occurred. The program reads the file status register at a future scanning time and takes action accordingly.

Block transfers, which are used by file administration to restructure data on the disc[3] take from 3 to 80 milliseconds to complete.

Other principal users of this mode are file audit programs, which need to read the disc in order to validate information stored on it, and the call store audit program to validate the nongeneric office parameters in call store.[4]

VI. ERROR DETECTION HARDWARE

Error detection hardware constitutes an integral part of the file design. This circuitry monitors the operation of circuits within the file and sets appropriate indicators in the file status register when an error occurs. Program action as a result of these indicators is discussed in Section VII.

Besides setting error indications in the file status register, certain errors, namely those associated with disc reading or writing, cause write-abort circuitry to operate. Activation of this abort circuit immediately stops disc writing operations. This is done to minimize possible mutilation of recorded data in the event that an error occurs in the read/write chain.

VII. RELIABILITY AND MAINTAINABILITY

Very high reliability is an essential characteristic of any real-time telephone system. This basic reliability objective dictated much of the file subsystem design philosophy. Hence, several important techniques were used to attain high reliability. Specifically, the file subsystem is completely duplicated with each file having sufficient call handling capacity to serve normal traffic loads with acceptable queuing delay. This means that one file subsystem can be down for repair without an appreciable effect on call processing performance. However, to keep the data bases in agreement, updating of the intercept records on disc is normally done only when both files are available.

In order for duplicated units to provide the desired high reliability, the repair time associated with each must be held to a minimum. In other words, the units must have high maintainability. Thus, another important designed-in feature of the file, intended to improve fault location and therefore reduce file down-time, is the incorporation of numerous test points accessible to the control unit. These points are of two types: the first type allows the control unit to examine the states of various internal file registers and flip-flops; the second type allows the control unit to set or reset selected flip-flops within the file complex. This circuitry provides the ability to stop and start clocks, read up to 16 flip-flops or gate outputs with each command, and set and/or clear most flip-flops and register stages. These features allow the programs to check both sequential and combinational circuits as if they were all combinational circuits.

A flip-flop register, called the Memory Register, is provided with multipled inputs, which can be selected by program from the CU. This register can be used to take "snapshots" of circuits while they are in operation. In this way, the occurrence of pulses can be detected, providing a powerful diagnostic tool in analyzing trouble symptoms.

Still other techniques applied to the file system to improve its reliability by minimizing its down-time are automatic error detection, location (also referred to as diagnosis), and system reconfiguration. These are accomplished by a combination of hardware and software methods. Checking circuits have been used in many areas of the design to provide immediate automatic hardware alarms in the event of component failure. Detection programs are provided which run periodically to establish whether the file is functioning properly and to take the proper action if a trouble is detected.

VIII. ERROR DETECTION AND LOCATION SOFTWARE

8.1 General

Diagnostic programs, used to locate faults within a file complex, can be requested either automatically or manually. Requests for automatic diagnostics may be initiated as a result of errors detected by any of the programs normally using the file or by system monitoring programs. Manual diagnostic requests can be made by maintenance personnel from the maintenance center teletypewriter.

8.2 User programs

All programs which use the file subsystem check for error indications in the file status register. These programs include the call processing program, the file administration program, the rapid scan audit program, and the routine exercise program.

Each of these programs is discussed below in somewhat more detail to show how they detect file troubles and request file diagnostic programs to be run automatically.

8.2.1 Call processing program

The predominant user of the disc file is the call processing program which may initiate several thousand intercept number lookups during the course of a busy hour. When a fault is detected by the file error detecting hardware, an error indication is set in the file status register. For example, if a parity error occurs during a normal lookup sequence, an appropriate bit is set. Another type of error that might be uncovered by call processing is a time-out, in which the lookup sequence has not been completed within its maximum cycle time. Either of the errors mentioned above would cause the call processing program to request a diagnostic test to be run on the file automatically.

8.2.2 File administration program

The file administration program which manages the data base performs error checks similar to those of call processing.

8.2.3 Rapid scan program

The function of the rapid scan program is to detect inconsistencies in file status indicators. The status of a file is kept both in file hardware and the control unit memory. When discrepancies are found, the rapid scan program attempts to reconcile the differences. As a result, diagnostic tests on a file may be requested automatically.

8.2.4 Routine exercise program

The routine exercise program tests not only the operational hardware but also the error checking hardware to insure that the latter will indicate a trouble should one occur. For example, one can cause a parity error to verify that the parity checker operates properly. These tests are run every 4 hours on each file, interleaved on a 2-hour basis. If an exercise program fails a test, it requests a file diagnostic to be run automatically.

8.3 Example of a file trouble

The following example illustrates the sequence of events which transpire when a fault occurs in a file. Consider the case where a flip-flop in the bit clock register becomes permanently "stuck-at-one." Any one of several programs could detect this condition, but in this case assume that an intercept number lookup is in progress. The bit clock error checking circuit causes an error bit to be set in the file status register. During each 25-millisecond interrupt, the call processing program reads the file status register to see if the lookup is complete. In the 25-millisecond interrupt following the fault, the call processing program finds the error indication. The sequence of events will then be:

(*i*) The file is marked "Maintenance Busy" so that no other user program attempts to use it.

(*ii*) The call processing program places a request with the system maintenance monitor for the file diagnostic program to be run.

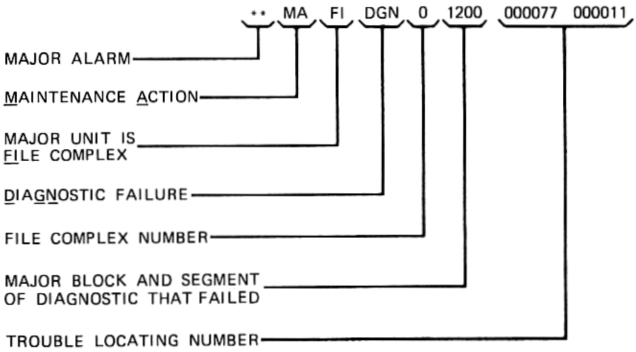


Fig. 7—Teletypewriter message for file trouble.

(*iii*) The file diagnostic is run and locates the faulty circuit.
(*iv*) The file is placed out-of-service and a message is printed on the maintenance teletypewriter (see Fig. 7).
(*v*) The next step requires action by the maintenance craftsman to correct the problem.

8.4 Call store mismatch due to file complex troubles

When the two CUs are running in synchronism, the CSI (*C*all *S*tore *I*nput) registers in the two units are compared at each call store read or write. If they differ, a mismatch interrupt will be generated. This means that if an attempt is made to write different data into the two call stores, a mismatch will occur. Certain file complex troubles can occur which could cause different data to be sent to the two control units. This type of fault will cause a control unit mismatch interrupt if:

(*i*) the received data are written into call store, or
(*ii*) a test and branch sequence is executed on the data that differ in the two control units.

The mismatch strategy used in the Automatic Intercept System is essentially that used in the No. 2 ESS.[5] The control unit mismatch recovery scheme, resulting from a control unit error, is discussed elsewhere. Recovery from a file error is based on this strategy. First the online control unit is tested. If it passes its tests, the communications links between the online control unit and each file are tested. If one of these tests fails, a low-level (Phase A) system initialization occurs. As a result, the "suspect" control unit is switched offline and diagnosed along with the file, since communications between the control unit and the file is in question.

If the online CU-file tests pass, but the tests between the offline control unit and a file fail, then the offline control unit–file communications diagnostics will be requested.

If a communications problem is found, a bit is set in a "bad access word" indicating that a trouble exists between the offline control unit and the specified file. When the problem is cleared and a control unit restoral is requested from the teletypewriter, offline CU-file diagnostic programs will be run automatically to insure that the access problem has indeed been cleared before the control units are put back in synchronism.

IX. FILE MAINTENANCE PROGRAMS

9.1 *File maintenance monitor*

A file maintenance monitor program controls the sequencing and priorities of all file maintenance programs.

In general, the priority structure in descending order is:

(*i*) Automatic requests based on suspected troubles either through hardware checking circuit activation or routine exercise detection.

(*ii*) Teletype requested diagnostics.

(*iii*) Periodic timed routine exercise programs.

If a diagnostic is being run when a higher-priority diagnostic is requested, the lower-priority diagnostic will be terminated gracefully and may be rerequested under program control once the higher-priority diagnostic is completed.

The function of the monitor is to:

(*i*) Determine if a test can be run.

(*ii*) Control the sequence of tests which are to be run.

(*iii*) Provide a common interface for the numerous file diagnostic blocks.

(*iv*) Screen the file input teletypewriter requests for appropriateness.

(*v*) Provide the appropriate teletype output messages.

9.2 *File diagnostic program*

The file diagnostic program consists of a sequence of diagnostic segments which start out testing the CU-file communications circuits and progressively test the more complicated internal parts of the file complex. This results in a systematic buildup of tested circuit elements which can be relied upon in subsequent tests. Because of this building-block approach, the tests must be run in a prescribed order. The major advantage of this approach is that the program can stop on first failure. The raw data results obtained, along with the number of the segment being run, are sufficient to locate the trouble to within several replaceable circuit packs. With this building-block type of diagnostic, one can find easily the particular program being executed when the failure was encountered and identify the particular test in the sequence that failed. This information may be useful to maintenance personnel in cases where additional analysis is necessary.

The sequence of tests forms a natural segmenting of the diagnostic programs. This is important since diagnostic programs are executed as part of the base level program and the length of each test is restricted to a maximum of 5 milliseconds in each base loop.

The tests are done in the following order. First, the control unit-file access circuits are tested. Next, the registers most closely associated with the peripheral unit address bus and the scan answer bus are tested. After the remaining static registers are tested, the dynamic registers (counters, etc.) are tested in two ways, first statically and then dynamically. This technique of testing dynamic registers in a static manner by stopping the internal file clock and providing simulated clock pulses under program control has proven to be a very powerful diagnostic method. It permits sequential circuitry to be tested as if it were combinational circuitry. However, dynamic tests are also run to detect marginal or speed-dependent failures which might not be detected in the static tests.

9.3 Automatic reconfiguration of file subsystem–control unit

Each control-unit-initiated order to a file is responded to with an enable verify (EV) pulse indicating that the unit received an order. For scan orders, where a response is expected, an all-seems-well (ASW) signal is sent with the response, again indicating that there are no known communications problems between the online control unit and the file. When the control units are running in synchronism, each unit executes identical code, including external orders, except that only orders from the online control unit are sent over the bus; the standby unit has its outputs inhibited.

The control unit tests for this immediate EV response (and ASW where applicable) and sets an indicator in the event a response is missing. Under software control, a reconfiguration program (also called a working mode program) may be called. In a series of steps, this reconfiguration program attempts to establish good communications between a control unit and the file which did not respond (Fig. 8). First, it retries the order since the failure may be of a transient nature. If the order fails a second time, the program then requests a switch to the other control unit. Once this is done it again retries the order; this time the new online control unit sends the order. Two separate actions are taken at this point depending on the outcome of the retry attempt:

(*i*) Success on retry after CU switch: this implies that either the now offline CU and its associated bus cannot send proper data, or that the related receivers in the file are faulty. In either case, the offline CU cannot communicate with the file and the offline CU is placed out-of-service. A program then tests the communications between the offline CU and the file to locate the trouble.

(*ii*) Failure on retry after CU switch: this means that neither CU can communicate with the specific file. Based on the "single-

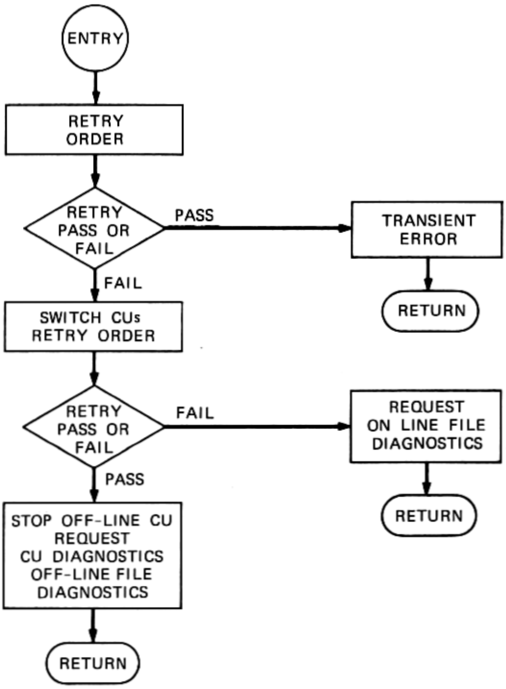


Fig. 8—CU-file reconfiguration procedure.

failure" philosophy, this implies that there is a hardware problem in the file. The file is removed from service and file diagnostic programs are requested to locate the problem.

X. FILE MAINTENANCE USING THE MAINTENANCE CENTER TELETYPEWRITER

10.1 General

A number of special-purpose programs, accessible from the maintenance center teletypewriter (MTC-TTY), provide the central office craftsman with a set of extremely flexible tools with which to maintain the file subsystem. From the MTC-TTY, such actions as determining file status, changing file status, and performing detailed or large-scale tests can be accomplished.

10.2 Teletype-requested functions

The following is a partial list of TTY-requested operations that can be performed on the file subsystem:

(*i*) Request a printout of status information—available, out-of-service, etc.
(*ii*) Request running of full or partial diagnostic tests.
(*iii*) Remove a unit from service.
(*iv*) Make a unit maintenance busy.
(*v*) Restore a unit to service.
(*vi*) Generate specific commands or orders to load or read most registers and/or particular flip-flops.
(*vii*) Request a file-look-up of a specified number and a printout of the information found.

10.3 Typical teletypewriter sequence

A typical teletypewriter interchange between a maintenance craftsman at the MTC-TTY and the file software system is illustrated in Fig. 9. In this example, the craftsman wishes to remove file complex 1 from service for the purpose of making a circuit change, performing preventive maintenance, replacing a suspect circuit pack, or some similar function. He proceeds as follows (the following steps refer to the message numbers in Fig. 9):

(*i*) The file status is requested. The system responds with "PF" (printout follows).
(*ii*) The file status is printed as "0 AVL" (complex 0 is available), "1 AVL."

```
M FI : SI ! PF                              (1) - (INPUT)

+tt MR FI SI O AVL 1 AVL                    (2) - (OUTPUT)

M FI : RMV : 1 ! OK                         (3) - (INPUT)

M FI : SI ! PF                              (4) - (INPUT)

tt MR FI SI O AVL 1 ØØS                     (5) - (OUTPUT)
•
•
•
•
M FI : RST : 1 ! IP                         (6) - (INPUT)
•
•
•
•
tt MR FI DGN 1 1500 000001 074200           (7) - (OUTPUT)


+ tt   MINUTES AFTER THE HOUR.
```

Fig. 9—Typical file maintenance TTY exchange.

Having ascertained that both files are available for system use, he reasons that file 1 may be removed for maintenance.

(*iii*) The system is requested to "remove file 1 from service." It responds with "OK."

(*iv*) The new file status is requested.

(*v*) The system now shows the status of file 1 as "1 OOS" (out-of-service).

Having isolated the file complex and prevented its active use by the system, the craftsman can now perform the desired maintenance function. Upon completion of the task, he attempts to restore the file to service.

(*vi*) The system is requested to "restore file complex 1." It responds with "IP" (in progress).

Before the system will restore the file, however, a complete diagnostic test will be automatically requested to insure that the file is functioning properly. Should the diagnostic pass, the system will respond with

tt MR FI RST 1 OK

and the file will be made available for normal use.

(*vii*) The file fails the diagnostic tests and the failing diagnostic segment (1500) is printed along with a "trouble number." The file complex is left out-of-service.

10.4 Isolating the trouble

To aid in locating the trouble, a Trouble Locating Manual (TLM) is referenced using the diagnostic segment and trouble numbers. This manual will supply the craftsman with a list of circuit packs to replace. Following the pack replacement, he may elect to restore the file—with resulting diagnostic—or he may simply request a full or partial (segment only) diagnostic directly.

Should the above procedure fail to effect a repair, other facilities are provided to aid in fault correction. One powerful feature, available from the MTC-TTY, is the ability to execute some of the teletypewriter requests on either a repetitive or a "one-shot" (i.e., repeat on request) basis. For example, a subtle or difficult-to-find trouble might be located by repetitively running the failing diagnostic segment, thereby enabling the craftsman to use an oscilloscope to trace the signals through the suspected circuit. Similarly, a repetitive lookup of an intercepted number may be requested. Using this method, the lookup would be repeated over and over, allowing the operation of the sequencers to be analyzed. Or if, as the result of a diagnostic, the trouble is localized to a group of several circuit packs, the diagnostic can be repeated on a "one-shot" basis following the replacement of each pack until the fault is corrected. To facilitate these features, a hand-held key/lamp assembly is provided which connects to the system Maintenance Center via a belt-line that strings through all the major frames in the office. Operating the key will stop and restart a repetitive test or simply trigger a one-shot test. The pass or fail results of the diagnostic are displayed on the lamps.

XI. GENERATION OF THE TROUBLE LOCATING MANUAL

In some systems, the Trouble Locating Manuals have been generated automatically by inserting faults, one at a time, running the diagnostic programs, and operating on the results with a data reduction program.[6] For the AIS file complex, however, the TLM generation was essentially manual, that is, it was formulated by the diagnostic programmer through circuit analysis. Both schemes have advantages and disadvantages. The principal disadvantage of the automatic scheme is that circuit changes require that a large part of the fault insertion procedure be redone. By comparison, the analysis method requires only the minor TLM changes that correspond to the actual circuit changes. The analysis method, on the other hand, requires that the TLM be prepared by experienced circuit technicians. However, these skilled people are usually required to write the detailed hardware diagnostic programs in the first place. Thus, the TLM is obtained with a small additional investment while the programs are being written.

XII. SUMMARY

The file subsystem has been designed to provide both fast and reliable storage and retrieval of intercept records. To achieve fast access, it was designed with two basic modes of operation. One mode, the lookup mode, allows the file to do associative searches of its contents independently of the CU. The second mode, the block transfer mode, permits the high-speed transfer of blocks of data between the control unit and the file.

To provide a reliable and maintainable system, a great deal of additional hardware and software has been incorporated in the design. This includes file duplication and the incorporation of error detection and checking circuits within each file. Furthermore, detection, diagnostic, and system reconfiguration programs are provided to quickly isolate faults in a subsystem to within a few replaceable circuit packs.

REFERENCES

1. E. J. Aitcheson, C. F. Ault, and R. G. Spencer, "No. 1 ESS ADF—Message Store —A Disk Memory System," B.S.T.J., *49*, No. 10 (December 1970), pp. 2887–2914.
2. W. B. Cagle, R. S. Menne, R. S. Skinner, R. E. Staehler, and M. D. Underwood, "No. 1 ESS Logic Circuits and Their Application to the Design of the Central Control," B.S.T.J., *43*, No. 5 (September 1964), pp. 2055–2095.
3. J. H. Carran, K. E. Greisen, W. G. Hall, and D. J. Wells, "Automatic Intercept System: Administering the Intercept Data Base," B.S.T.J., this issue, pp. 133–153.
4. H. Cohen, D. E. Confalone, B. D. Wagner, and W. W. Wood, "Automatic Intercept System: Operational Programs," B.S.T.J., this issue, pp. 19–69.
5. H. J. Beuscher, G. E. Fessler, D. W. Huffman, P. J. Kennedy, and E. Nussbaum, "[No. 2ESS] Administration and Maintenance Plan," B.S.T.J., *48*, No. 8 (October 1969), pp. 2765–2815.
6. R. W. Downing, J. S. Nowak, and L. S. Tuomenoksa, "No. 1 EES Maintenance Plan," B.S.T.J., *43*, No. 5 (September 1964), pp. 1961–2019.