THE BELL SYSTEM

Technical Journal

TSPS No. 1

No. 1 ESS ADF

THE BELL SYSTEM TECHNICAL JOURNAL

TSPS No. 1:

System Organization and Objectives

By R. J. JAEGER, JR., and A. E. JOEL, JR.

This article is an introduction to a series of detailed technical articles that describe the Traffic Service Position System No. 1. The organization and objectives of the system are given and the overall operation of the system is explained.

I. INTRODUCTION

At the outset of telephony, connections between telephones were established by operators through cords, plugs, and jacks at switchboards. Within the first decade, switchboard design and operating techniques soon settled into a pattern that is still followed today. The switchboards of 80 years ago are remarkably similar to those commonly used today. (See Figs. 1 and 2.) With rapidly increasing traffic, it became apparent that to use operators to establish local calls was not the best way to continue. Dial switching was invented.

Although the shift to local dial office operation started early in this century, it was in the years following World War II that the transition was rapidly completed. (See Fig. 3.) As the efficiency of transmission

2417
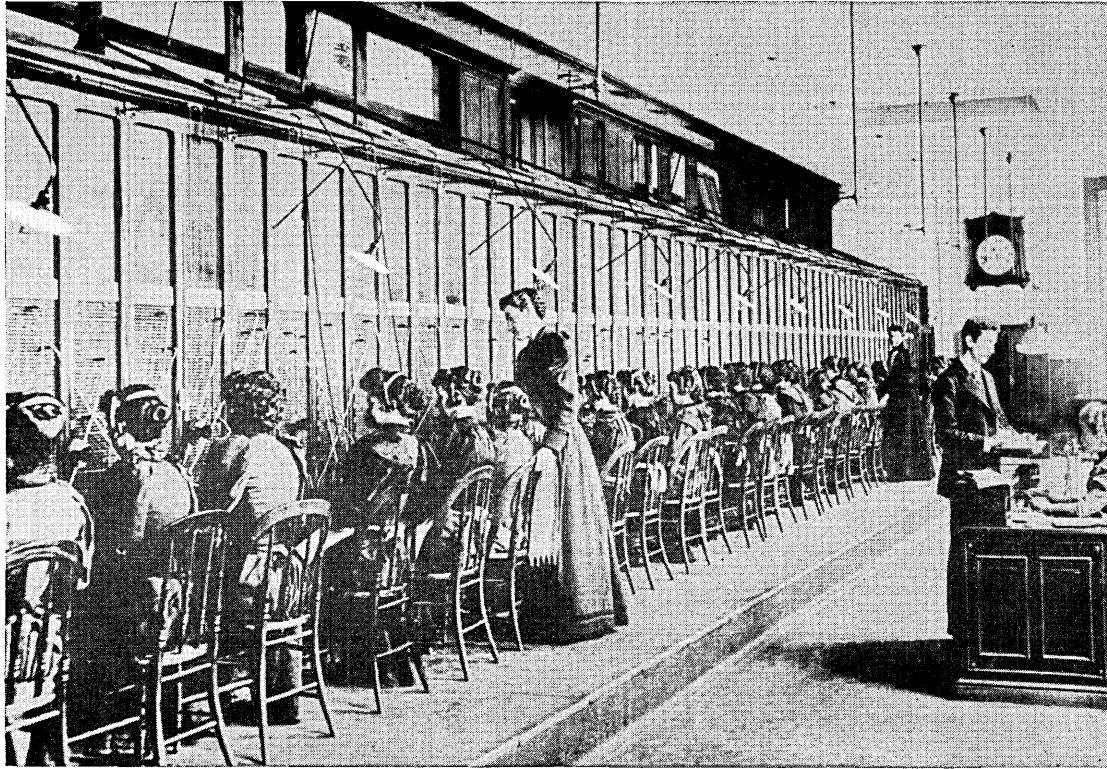
Fig. 1—Multiple switchboard in the Cortlandt exchange, New York City, installed in 1888.

Fig. 2—Switchboards today in New Jersey.
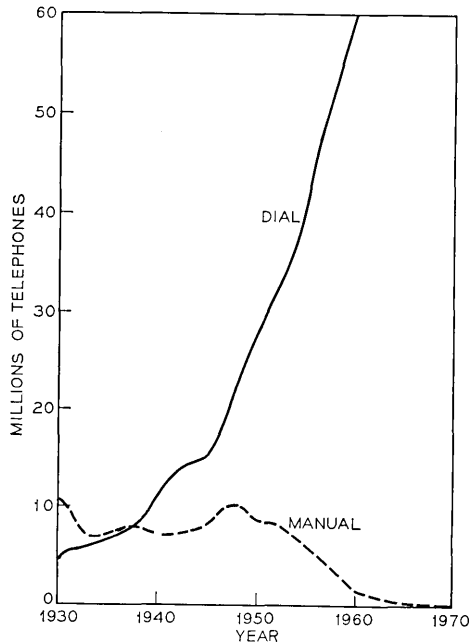
Fig. 3—Station growth.

and signaling facilities improved, the burgeoning use of the telephone for long distance calling permitted the development of arrangements by which operators dialed and supervised toll calls from their switchboards. These capabilities speeded call completion and improved operator charging accuracy.

In the 1950s customer direct distance dialing was introduced for station-to-station calling. This used the established toll dialing network and the new automatic message accounting facilities for recording the call details needed for charging. Thus, through technological advances the need for operators was greatly ameliorated by the customer's dialing both local and toll calls.

Operators were still needed, however, to handle local assistance calls, coin toll calls (to monitor coin deposits) and special toll calls, such as person-to-person and collect calls. The need for operators was eased only temporarily because the unabated growth of stations and traffic has been accompanied by an increase in the number of special toll calls, the expanding use of coin stations for toll calls, and the introduction of credit card calling. (See Fig. 4).

Coincident with the problem of obtaining greater numbers of operators was a changing labor market. It was becoming more difficult to hire

women to work in the urban switchboard locations, particularly for weekend and night duty. Competition from broadening job opportunities in business offices, sales work, and suburban factories was being felt. These factors combined to start a search to do the operator's job more efficiently and to make it more attractive.

Toward this end, the New York Telephone Company developed an operator assistance switchboard for customer dialed special toll calls to work with No. 5 crossbar system called PPCS (person-to-person, collect, and special) in 1958.[1] Based on this start, Bell Telephone Laboratories undertook the development of a standard cordless console with semiautomated call handling of customer dialed special toll calls through the crossbar tandem system in the early 1960's to provide a more comprehensive design to meet the broader needs of the Bell System. The new console was named the traffic service position (TSP) and the first standard installation went into service in 1964 in Cleveland, Ohio.[2] There are now some 21 installations in major cities. To provide TSP features with the other toll switching systems in the same manner would have required similar development for the No. 4 toll crossbar and No. 5 crossbar systems.

In order to provide a single system that could work with any toll switching system, present or future, the idea of an autonomous system to provide the automatic operator functions was conceived. That system, named the traffic service position system No. 1 (TSPS No. 1), has been developed. It is the purpose of this article and the six that follow to describe the new system.
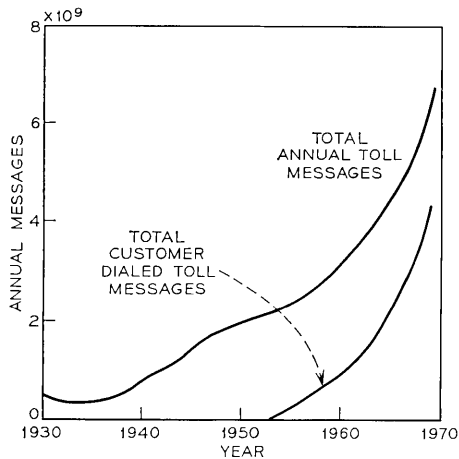
Fig. 4—How toll calls have increased.

II. OBJECTIVES

The basic objective of the system design is to relieve the operator of handling the routine switchboard operating functions so that she can concentrate on providing those functions which require judgment to be exercised and data to be put into the system. The initial design of the system provides for the types of calls and the operator functions described in Table I.

There are a significant number of call types now handled on toll switchboards that were not included in the initial design of TSPS in order to limit the size of the development and to meet the most urgent needs of the field. Typical call types not included in the initial design are conference calls, mobile and marine calls, inward to operator calls, and special handling of hotel and motel calls.

In order to handle the calls outlined in Table I and to obtain the desired service, administrative and operational improvements discussed in the introduction, these general design objectives were established:

(i) Autonomous system with standard trunk interfaces to other systems.

(ii) Modern attractive consoles.

(iii) Economical position remoting capability.

(iv) Stored program system design.

(v) Economical over a reasonable range of office sizes.

(vi) Improved administrative features.

(vii) Improved maintainability.

(viii) Flexibility in adding new features.

(ix) A generic processor-memory complex useable in other system applications.

(x) Use of "on-the-shelf" hardware.

In order to expedite development and field introduction, it was decided to adapt both the basic hardware components and the system structure of No. 1 ESS.[3] A number of fundamental modifications were made, however, and these are described later in this series of articles.

To establish bounds for the size of the design, a review of the potential market was made. Considered were trunking needs and the number and size of chief operator groups.* With the recognition that the system would receive wide application, equipment quantities were selected so that no one system component would be limiting in obtaining maximum

---

* A chief operator group is the location of a group of positions under a single administrator or chief operator.

TABLE I—OPERATOR FUNCTIONS

From Stations	Input Credit Card or Third Number	Called Customer Identification on Person-to-Person	Obtain Acceptance of Charges on Collect	Operator Identification of Calling Number*	Monitor Coin Deposits	Operator Assistance
Noncoin	0+	0+	0+	1+, 0+	—	0−
Coin	1+, 0+	0+	0+	1+, 0+	1+, 0+	0−

1+ Customer dialed station-to-station calls
0+ Customer dialed special calls
0− Operator assistance calls

* Needed only when calling number is not automatically identified and forwarded from the local office.

system utilization. As it now stands, the design is limited by data processing capacity. The maximum quantities of the major system elements shown below have covered the needs of all installations to date.

(*i*) 3000 trunks,
(*ii*) 310 positions, accessible as a single team,
(*iii*) 62 positions per chief operator group, and
(*iv*) 9 chief operator groups, remote, local or both.

The number of cities expected to require multiple installations of the system is small, indicating that a larger system with its higher start-up cost was not justified for the current potential market of several hundred systems.

III. SYSTEM DESCRIPTION

3.1 *100B Console*

Since the basic function of TSPS is to automate the routine aspects of the operator's work, the system description begins from the operator's viewpoint: the way calls appear on her console, her responses, and the features of the console. Against this background, descriptions of the software and hardware should be more meaningful.

Figure 5 shows a typical operating room. Each console section contains two positions in a desk-like arrangement. Figure 6 is a closer view of the console keyshelf with its lamps and keys. Except for the digital display in the uppermost portion of the section of the console, the remaining lamps are on the main panel. Some keys are equipped with lamps and are indicated in the legend.

A position becomes available to the system for service when an operator inserts her headset plug into its jack. Calls are automatically distributed to all attended positions in such a way that all operators receive an equal share of the traffic load. When a position is given a call, the operator hears a zip tone and is given a lamp display. A lamp indicates the type of the originating station, coin or noncoin, and whether the customer dialed 0 followed by 7 or 10 digits (0+), dialed 0 only (0−) or dialed a station-to-station call (1+). The use of a "1" prefix for station calls is not universal, but for purposes of description, station-to-station calls are often referred to as 1+.

With these indications the operator is able to respond appropriately. On calls received from coin stations, the deposit for the initial period and duration for the call dialed by the customer are also indicated in the numerical display when the operator takes the call. On 0+ calls she
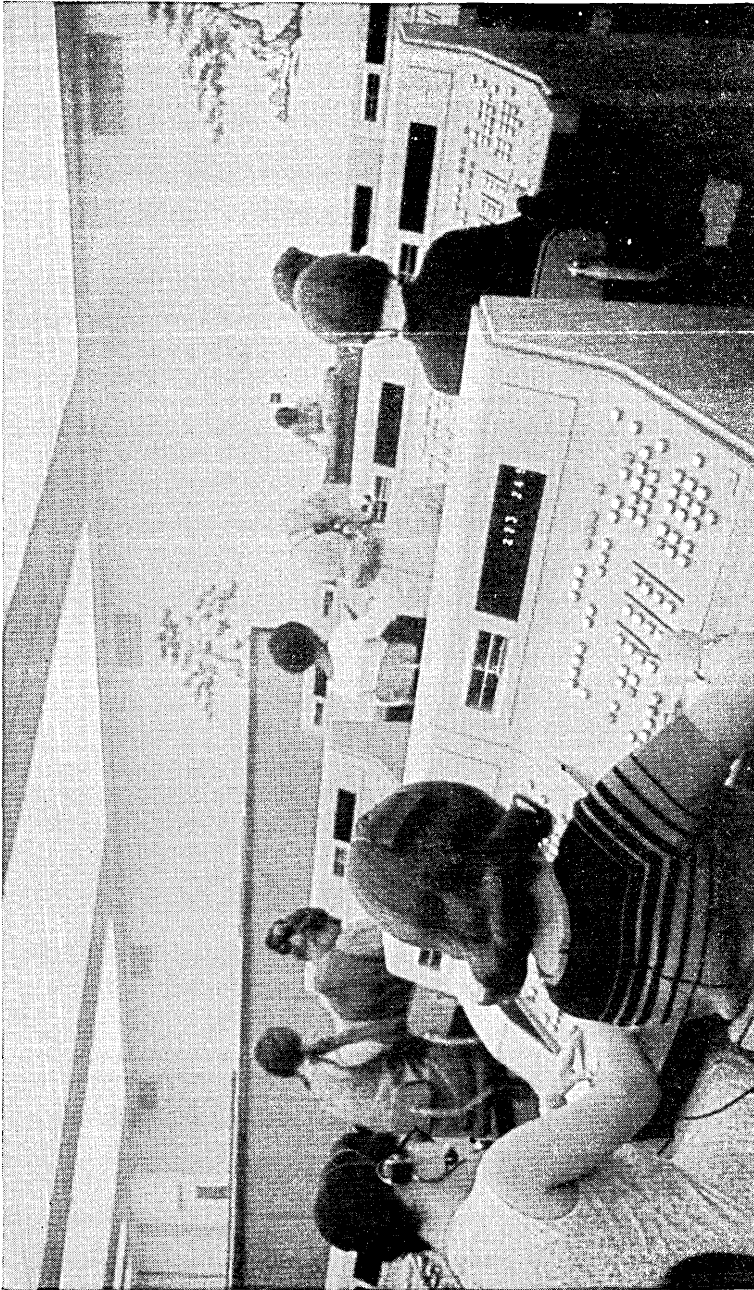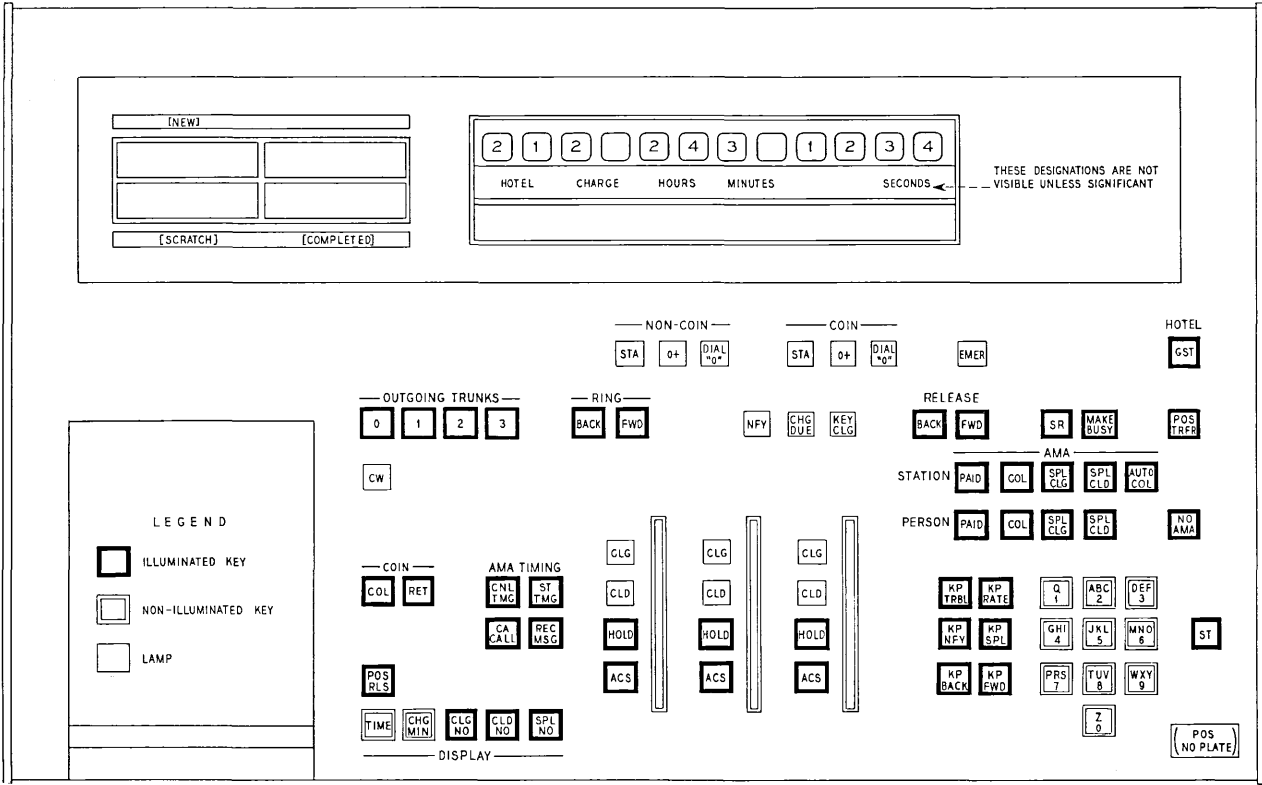
Fig. 5—A typical operating room.

Fig. 6—100B traffic service position key shelf.

asks what type of operator assistance the caller wants. When she determines the type of call being placed, the operator depresses a class of charge key. If it is necessary for the operator to enter a credit card number, the number of a third station for billing purposes, or to provide the called number or calling number, there is a keyset in the lower right corner which is conditioned by the operation of an appropriate key pulse key to indicate the type of number being entered into the system by the operator.

In the lower center part of the key shelf are three columns of keys which are referred to as "loops". When a call is connected to the position, it is associated with one of these loops. When the operator is in voice contact with the customer, the bottom key lamp designated ACS (access) is lighted. If the operator desires to keep this call associated with her position while handling other calls, she can push the HOLD key lamp. The upper two lamps of each loop indicate the switchook condition of the originating and terminating stations when a call is in either ACCESS or HOLD. The call is released from the position when both the ST TMG (start timing) key to the immediate left of the three loops and the POS RLS (position release) key in the lower left corner are operated.

Whenever a call is connected to a TSPS position, all call details are available from the system memory. These call details are directly equivalent to those that she would have written on a ticket if the call had been processed at a cord switchboard. She can display, under key control, the calling number, the number that is being called, a credit card number if keyed into the system, the number of a third telephone if one is being billed, and the charging rate on coin calls. Also, the operator can get the exact time of day being used by the system by depressing the TIME key. The displays associated with the two leftmost keys do not lock in and are only displayed as long as the key is operated. The time display is fixed at the instant the key is depressed.

Other operator controls include the ability to release connections forward or backward, ring the stations forward or backward, collect or return coin deposits and connect to special service operators over outgoing trunks.

Although the operator has some freedom in the way she handles calls, many of her key actions are automatically checked and flashing lamps used to indicate detectable errors. If, for instance, the operator depresses the START TIMING key before a class of charge has been recorded, the START TIMING lamp will flash. The operator must then determine what information is missing. Similarly, keying 11 digits on a 0— call will cause the KP FWD (KP forward) lamp to flash.

For all calls that are to be automatically timed by the system two key operations normally end the initial position seizure. These keys are ST TMG (start timing) and POS REL (position release). The START TIMING key is used to indicate to the system that the operator has all of the billing details on the call, and that when her position is released from the call, and the called station answers, the time of the connection is to be recorded. On 1+ coin calls the START TIMING key may be operated before the called party answers, but on person-to-person calls the START TIMING key is operated after the conversation starts. In either case, the time of connection is established when the operator is disconnected from the call. Timing for charges stops when a call is reconnected to a position for any reason.

3.2 *General*

Figure 7 is a basic block diagram for this system. All trunks have two two-wire appearances on the link network. The network connects the trunk to various service circuits—digit receivers, outpulsers, coin control circuits, tone circuits, and operator positions. Being a stored program system, the basic call logic instructions are in the memory and are executed by a processor. Changes of the supervisory state of trunk,
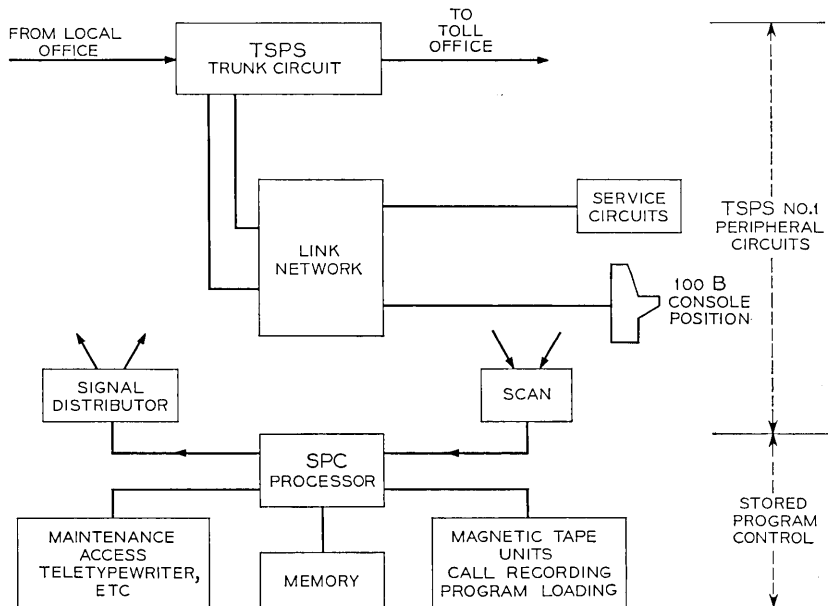
Fig. 7—TSPS elements temporarily connect operators and equipment to the trunk circuit.

service, and other peripheral circuits including the positions themselves, are detected by scanners. Output instructions via signal distributors and central pulse distributors control these circuits and the position lamps.

During the conversation period of a call, the customers are connected only to each other; there are no connections to operators or service circuits. It can be seen in Fig. 7 that the TSPS trunk circuit is a dedicated facility connecting the local office to the toll office. No switching capability nor concentration exists for this trunk circuit at TSPS. Thus, TSPS is a unique type of switching system because all the elements of a switching system are present only for the purpose of temporarily connecting equipment units and operators to the trunk circuit.

3.3 *Equipment*

The system equipment is divided into two principal parts, the processor-memory complex, and the TSPS No. 1 periphery which includes the position subsystem.

3.3.1 *Processor-Memory Complex*

One of the major points of divergence from the No. 1 ESS design is in the processor-memory complex. The processor-memory complex, including such supporting units as the control and display panel, a signal distributor, a central pulse distributor, a master scanner, the maintenance teletypewriter and the program tape unit for loading and unloading memory, constitute a subsystem called the stored program control No. 1A (SPC). The SPC equipment is covered by a separate set of engineering documents and has well defined interfaces so that it can be easily used in other applications. It is used with the electronic translator system for No. 4 toll crossbar as well as for the TSPS No. 1.[5]

The SPC No. 1A design incorporates several novel improvements. For example, the duplicated processors have fast dc matching circuits that compare the states of certain internal circuits and are able to abort orders within the failing cycle when a mismatch occurs. This provides for improved discrimination of errors and faults as well as better recovery from mismatches. Another difference is in the use of a simplified store bus arrangement to minimize the complexity and inherent hazards of having cross bus switching at every store frame.

For system simplification, a single type of store using the electrically alterable, nondestructive readout piggyback twistor was chosen to provide all of the memory required by the system applications.[4] The advantage of using this memory is discussed elsewhere in this issue.

3.3.2 *TSPS Periphery*

Many of the peripheral units of TSPS No. 1 are similar in function and appearance to those of No. 1 ESS. The plug-in circuit packs, the framework, and terminal strips are virtually identical. Details of the differences are described in Ref. 6.

The switching network used to connect the trunks to the service circuits and positions is a four-stage, two-wire space division network using ferreeds. In order to provide uniform traffic loading of the linkages for the widely differing traffic occupancies of TSPS trunks, a unique network configuration that involves build-out frames has been provided. The build-out frames are physically attached to the basic frames by cable connectors to ease the problem of network growth.

The plug-in trunk units and the universal trunk frames look like No. 1 ESS units. They are quite different in function and this is reflected in the internal design. Because TSPS works with any local office type, it must be able to receive multifrequency pulsing or dial pulsing over both loop and carrier facilities. Since the serving toll office may have a four-wire switching system, both two-wire and four-wire trunk circuits are provided. The four-wire trunk circuits are used when the toll office has four-wire switching and the incoming trunk facilities are four-wire.

3.3.3 *Position Subsystem*

While the position subsystem is really part of the TSPS periphery, it is sufficiently large and novel to warrant separate mention. Orders from the SPC to control the lamps at the consoles are decoded in a position signal distributor which operates or releases magnetically latched miniature wire spring relays. These relays are circuit pack mounted in the position buffer and, with a modest amount of wired logic, control the lamps and key lamps on the console. The console's numerical display is controlled directly from the position signal distributor.

Operators' key actions are detected by an autonomous scanner which codes and gates the keyed information along with the position number back to the peripheral master scanner.

When the positions are remotely located, the orders to control the position lamps and the data words for key operations are sent via the digital T1 carrier system which acts as an extender of the peripheral bus. The T1 carrier also provides the voice circuits for the operators. The manner in which the T1 system provides voice and data capability simultaneously is described in detail in the article on the TSPS periphery.[7]

3.4 *Software*

The program structure for TSPS closely follows that for No. 1 ESS.[3] An executive control program, interrupt levels, priority work lists, fault recognition programs and diagnostic programs are employed to provide the real-time, time-shared characteristics of the system. The individual programs are arranged to be part of either the SPC or TSPS similar to the equipment dichotomy described in section 3.3.1.

3.4.1 *SPC*

The SPC program package provides the system framework within which the TSPS programs function. It is composed of the executive control, maintenance control, interrupt and input-output programs. It also has those programs needed to maintain SPC equipment. The interfaces between the SPC and application programs have been carefully arranged to avoid significant penalties in overall program size and system real-time usage.

The SPC program package is covered by separate documentation that is coordinated with the SPC equipment. There are 35,000 words of SPC program of which 24,000 are for maintenance functions.

3.4.2 *Traffic Service Position System*

The TSPS programs operate within the SPC framework to achieve the desired total system functions. They are composed of both call processing and maintenance programs.

The call processing programs provide call handling logic and are concerned with the input-output functions of scanning and signal distribution, and the control of peripheral hardware to process calls. These programs detect changes of state, provide the data processing to interpret the changes of state, and issue orders to the periphery to change the status of the hardware to further the progress of the call. For example, call processing programs provide for the reception of digital information, establishing connections through the ferreed link network, the sending of information to control the lamp displays at operator positions, the reception of key signals from the operators, the computation of coin charges, and the placing of billing information on the magnetic tape. These programs operate under the stored program control executive control program structure which insures that all necessary work will be given an opportunity for execution within selected time periods.

For the programs that cover the operator functions, care has been

taken to insure that standard operating practices are reflected in the program design. The primary intent here is to give the operator maximum latitude in sequencing her actions while insuring that she properly completes all the necessary functions before releasing the call from her position.

There are 80,000 words of TSPS program of which approximately 37,000 are for maintenance of the peripheral equipment including the local and remote positions.

3.5 *Maintenance Features*

3.5.1 *General*

In this system as in other Bell System central office electronic switching system designs, the maintenance strategy is based on having vital hardware units in duplicate, using signals to indicate the successful execution of orders and using programs to test, detect faults, and diagnose trouble. The deductive and test strategies designed into the programs are relied on heavily to obtain fast and effective results. The use of teletypewriters and lamp displays to simplify the man-machine interface aids in an overall improvement in maintenance. The intent is to achieve greatest reliability through automation wherever possible.

3.5.2 *Equipment*

The most fundamental aspect of equipment maintenance is the extensive use of plug-in circuit packs to simplify replacement of defective units and to minimize the need to repair or adjust apparatus in the office. Duplication of important, commonly used equipment assures continuity of service while failing units are being repaired or additions are made. One method of duplication is achieved by having two separate and identical units, one active and the other a working standby operating in parallel. Only the active unit participates in system call processing. The stored program control processor and stores are examples. Another technique is to provide two halves of a unit and divide the work. When one half fails, the other half takes over the total work load; the link frame controllers are an example. There is also duplication in power and bussing of data signals.

As already mentioned, the teletypewriter is the primary means for a man to communicate with the system. Messages printed on the teletypewriter identify trouble and failing units. The messages provide

the needed detailed information for the maintenance craftsmen to take action. There are periodic teletypewriter reports given on system functioning so that a picture of the relative state of the machine can be obtained periodically. It is also possible via the teletypewriter to modify system operations. For example, the maintenance craftsman can request the system to test particular units.

In addition to the teletypewriter, there are adjacent display panels at the master control center for the SPC and TSPS equipment that give the status of the major equipment elements indicating whether they are in trouble or removed from service. These panels also have control keys which permit the maintenance craftsman to directly reconfigure the system. This permits him to manually associate certain equipment units to form a working system when automatic software defenses are inadequate or other troubles warrant.

The program tape unit for loading the electrically changeable memory via a magnetic tape is located in the master control center (see Fig. 8). This unit can also be used for recording on magnetic tape the contents of system memory to aid in the analysis of program problems. It is also used by the Western Electric Company to obtain a magnetic tape record of office data when engineering for growth.

Craftsmen gain access for testing service and trunk circuits, and to make transmission measurements over trunks through the control display and test panel. A voltmeter is available for making the normal ground and battery cross-checks on cable pairs that connect to the system.

The duplicated automatic message accounting magnetic tape units are in the master control center so that the maintenance craftsman can remove and replace the magnetic tapes on which call billing data are placed. The time of day circuit has key controls, also located at the master control center, for setting the time if commercial power interruptions cause slight time errors to occur.

Almost all frames in the office have the standard POWER OFF keys which are used to remove power from circuits when replacing faulty circuit packs. Because the position subsystem can sometimes have some of its equipment located at a remote site, special controls are provided at the remote location to aid the maintenance man in clearing troubles. Using these controls to enter requests, the system sends selected test orders to the position subsystem signal distributor and position buffer frames. These and other maintenance features unique to the position subsystem are described in Ref. 7.
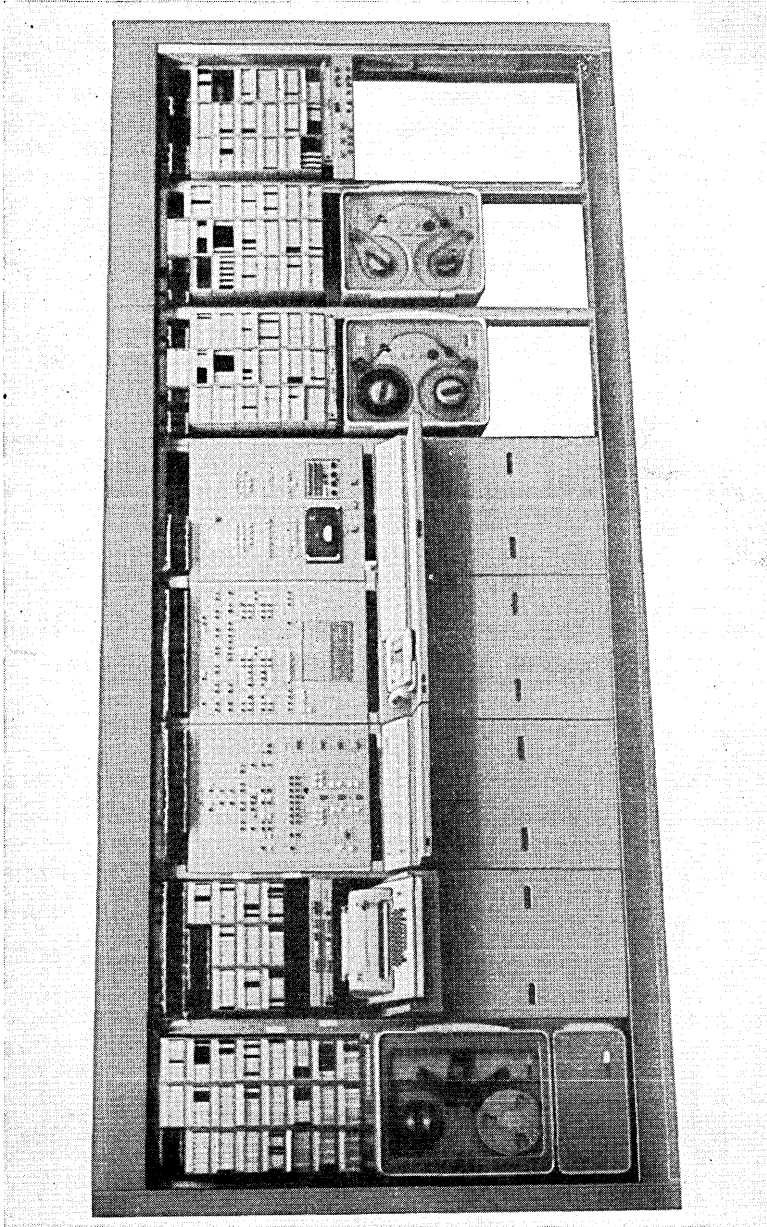
Fig. 8—TSPS master control center.

3.5.3 *Software*

Both the TSPS and the SPC provide fault recognition and diagnostic programs for all major circuit elements. The purpose of fault recognition programs is to quickly detect faulty equipment units and cause them to be removed from service. When the execution of an order fails in the processing of a call, an appropriate fault recognition program is called in to determine if the trouble is repeatable and if so, to locate the faulty unit and remove it from service. Later, on a lower level of priority, diagnostic programs are called in to test the removed unit to isolate the trouble to a small circuit area. A trouble test number is printed on the maintenance teletypewriter for each specific test that fails. The maintenance man looks this number up in a trouble locating manual which indicates the circuit pack or packs that are likely to be faulty.

IV. DESCRIPTIONS OF TYPICAL CALLS

The operation and purpose of the system is best understood by describing how it processes two of the many varieties of calls. The numbers in Fig. 9 might be helpful as a guide.

4.1 *Coin Call*

When a customer places a toll call from a coin station associated with a local office that is served by a TSPS, the instructions on the coin telephone direct that the complete called number, that is, the area code (if required) and the seven digit number for the desired party be dialed as soon as dial tone is obtained. If the customer wishes to place a station-to-station call, either a one prefix or no prefix is dialed prior to the called number depending upon local numbering plan arrangements.[8] As soon as the local office receives the digits, it determines that this call requires the services of a TSPS operator and the call is forwarded over a trunk to the TSPS. (See *1* in Fig. 8.) At the TSPS, as soon as the trunk circuit seizure is detected by scanning (*2*), a connection is established through the link network (*3*) to an appropriate service circuit.

Assuming the call is from a local office that has multifrequency outpulsing, a multifrequency receiver is attached and a supervisory signal is returned to the local office to indicate that outpulsing can begin. As each digit is received in the multifrequency receiver, it is detected by a directed scan (*4*) and placed in memory (*5*). When the complete called number has been received, a supervisory signal is returned to the local office requesting that the calling number be identi-
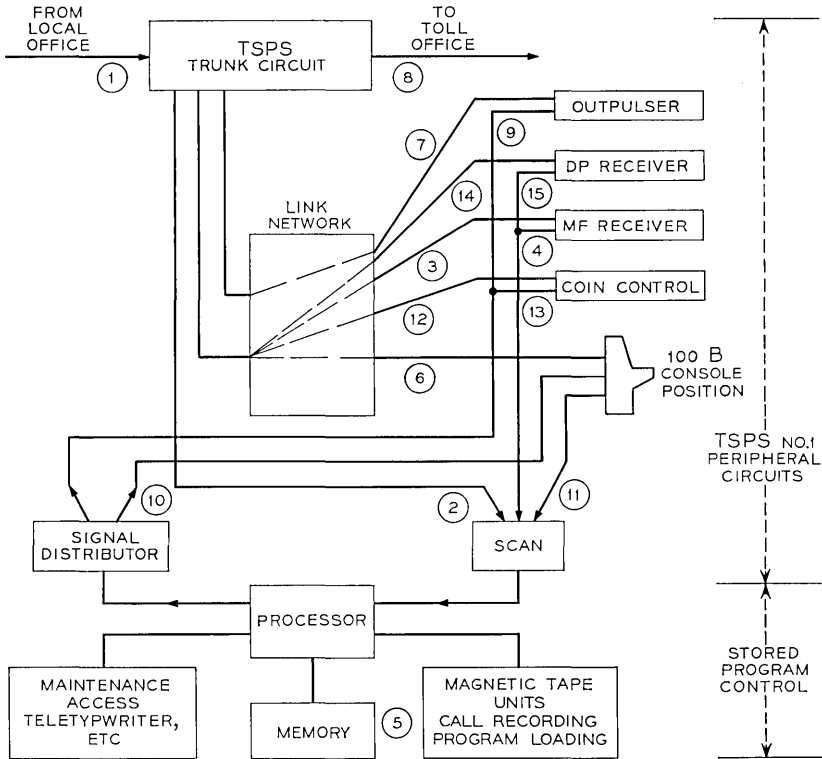
Fig. 9—To understand typical TSPS calls, study this drawing and its numbered parts in conjunction with the text.

fied and outpulsed to the TSPS office. In this case the multifrequency receiver remains attached to the trunk and receives the complete calling number and places each digit in memory.

When both the called and calling numbers are stored, the system simultaneously seeks an idle operator position (6) and an idle outpulser (7). When both are available, it establishes connections from the two appearances of the trunk circuit to these two circuits (6), (7). While the operator is responding to the call and talking to the originating party, the outpulser is connected to the toll office (8) and after an exchange of supervisory signals, outpulses the called number supplied to it a digit at a time by the central pulse distributor (9).

The operator's display, controlled by a signal distributor (10), indicates that this is a coin call and gives the initial charge and time period

for the call on a numerical display. She depresses the class of charge key to indicate to the system that a station-to-station type of call is being handled.

The customer, in responding to the request for money, deposits the coins. The operator monitors to see that the proper deposit has been made. In the meantime, the outpulser completes its function, is released, and the call is forwarded through the direct distance dialing network. An audible ring can be heard by both the operator and the originating party because the trunk has now connected the calling and called parties. Since this is a station-to-station call, the operator depresses the start timing key and releases her position as soon as the coins are deposited and she hears audible ringing. She is not required to participate in the call any further.

If the call is of the person-to-person type, she depresses a different call type key and remains with the call until the called station answers. After determining that the desired person is speaking, she depresses the start timing key and releases her position (6). All key actions are detected by a scanner and recorded in memory (11).

With the operator released, the call is timed by the system for the initial period. Eighteen seconds prior to the completion of the initial period, the TSPS sends a coin collect signal (12), (13) to the local office to cause the initial deposit to be collected automatically. This alerts the customer that the initial period is drawing to a close.

At the end of the initial period the system connects (6) an idle operator position to the trunk circuit. The position is not likely to be the same operator as for the initial seizure. This operator notifies the customer that the initial period is completed and that his call is going into the overtime period. If the call continues more than six seconds after the operator releases, the overtime period begins.

When the call is concluded, as indicated by on-hook supervision, the system seizes an available position (6) and indicates to its operator that this is the end of an overtime coin call. The overtime duration and charges are displayed. If the calling customer's phone is on the hook, the operator rings it by operating the appropriate key. After monitoring the deposit of the overtime charges, the operator depresses a key which signals that coins are to be collected and releases her position. Whenever charges are given to the operator for coin collection, appropriate taxes are computed and automatically added to the charge.

Studies have shown that normally it is not economical to provide memory space for coin rating tables for all calls. Most telephone companies attempt to provide automatic coin rating for about 95 percent

of their coin traffic. When calls are not automatically rated, the operator receives a display indicating that she must do the rating of the call either by using the bulletin on her position or by calling a special operator over one of the outgoing trunks. Having determined the rating information, she then keys it into the system and the system automatically computes the charge, adding appropriate taxes. Except for the manually rated cases, it is the responsibility of the system and not of the operator to use the proper rate period based on the time of day and to keep track of holidays and weekends.

4.2 *Person-to-Person Credit Card Call*

Calls made from noncoin stations are handled basically the same as they are from coin stations without the use of the coin features. To illustrate the difference in operation when dial pulsing is involved, let us follow a call from a step-by-step office.

Assuming a noncoin customer in a step-by-step office wishes to place a person-to-person call, a "0" will be prefixed before the 10-digit direct distance dialing number. The "0" indicates to the local office that the services of an operator are desired and a trunk to the TSPS is seized immediately after the initial "0" is received. Because there is no second dial tone with this type of service, the TSPS must be prepared immediately to receive dial pulse digits as soon as it detects the trunk seizure.

Because it may take 190 milliseconds to establish a connection (see *14* in Fig. 8) to an idle dial pulse receiver, the dial pulse incoming trunk circuits in the TSPS are arranged to receive and store up to two dial pulses to provide the needed time to establish the connection. When the receiver is connected to the trunk circuit through the network, a dc signal is sent from the trunk to the receiver indicating whether 0, 1, or more than 1 pulses have been received. If more than 1 pulses are indicated, the receiver signals (*15*) the processor to connect the call to reorder because it cannot assure the accuracy of the pulse counting. However, this occurs in less than five in 1,000 cases, which meets the designated service criteria. The remaining dial pulses are counted in the dial pulse receiver and at the end of each digit the processor (*15*) is given the digit for storing in memory as in the case of the multifrequency pulsing.

At the end of the called number dialing, the system releases (*14*) the dial pulse receiver and connects (*3*) a multifrequency receiver to the trunk circuit through the network. A supervisory signal is sent to the local step-by-step office to request identification of the originating

number which is sent to the TSPS and recorded in memory as in the previously described coin call. In this instance, when the position is seized, the indication to the operator is that it is from a noncoin station and that the customer dialed "0" followed by 7 or 10 digits. She asks the customer what service is required and the customer indicates that he is making a person-to-person call using a credit card. The credit card number is then given to the operator and she keys it into the system. When the called station answers, she seeks the desired party. If the party is not at the station that was called and she determines that the person is at another station, she can release the forward connection (8) and key a new called number into the system (11). An outpulser is again connected (9) to the trunk to send forward the new called number. When she determines that the right parties are connected to each other, she depresses the start timing key, releases her position and the call is now "floating" (not associated with a particular operator) with the system monitoring the switchhook supervision of both the called and calling parties.

If for any reason the calling customer desires the assistance of an operator, a switchhook flash is detected by the scanner (2) and an idle position is connected (6) to the call. The operator determines the problem or the reason for being brought in on the call and takes the appropriate action. If there is a noisy condition, poor transmission, or wrong number, she can disconnect the called number and try again.

All positions in a TSPS are treated as members of one large team of operators to gain efficiency in service. There are several benefits. When the team size exceeds 120 operators, it is possible to achieve the 92 percent occupancy which has been established as an objective for operator services. Also, the full access of all trunk units to all occupied operator positions is valuable during light loads because all traffic can be sent to a small group of operators. This permits all but one group to be completely shut down during the lightest loads.

V. ADMINISTRATIVE FEATURES

5.1 Recent Changes

As in most complex telephone systems where many of the features are dependent upon the local conditions, call routing and charging must be appropriate to that area. Therefore, it is necessary to provide information known as "office data" for a particular installation to function properly. It is often necessary to modify or change this information from time to time.

In TSPS, because it uses electrically alterable memory, it is possible to make all changes through teletypewriter messages and for these changes to become a relatively permanent part of the system memory as soon as they have been verified. In order to minimize human errors, the TSPS recent change messages use an easily understood alphanumeric format. The TSPS programs insure that the information being provided for recent changes is rational within the context of the machine. It is also necessary for a person making a recent change to specify the existing information that is being changed. This acts as a double check and minimizes the possibility of errors getting into the system. In order to provide a simple method of operation, a separate teletypewriter channel is provided for recent changes giving the telephone company the option of placing the machine in the most convenient location. Frequently it is located in traffic quarters since most recent change information originates from the traffic department. The greatest recent change activity involves the coin rating tables and the size and classes of the trunk groups.

5.2 *Traffic Facilities and Force Administration*

Of fundamental importance in the proper use of any switching system is the administration of the traffic dependent facilities of the system. Adequate quantities of hardware such as receivers, outpulsers, and coin control circuits, as well as software buffers, hoppers, and registers, must be maintained. In addition, a system like TSPS requires a sufficient number of operators to meet certain service criteria such as are set for speed of answer. To assist the traffic people of the operating companies in monitoring the adequacy of these facilities, traffic data is printed out periodically on a teletypewriter at the central administration group.

Part of this data is printed hourly to record peg counts and traffic usage measurements. Peg counts of certain service circuit seizures, usage measurements of in-service and out-of-service circuits, and usage measurements of many software facilities are provided. Additional information is also provided at 15-minute intervals on the load level of the system, the number of trunk seizures, trunk seizures requiring dial pulse receiver use, dial pulse calls encountering reorder, and the number of trunks made busy during an overload. Using this information, the facilities engineer can make adjustments in response to changes in the telephone traffic patterns and loads.

The central administration group also receives data concerning operator usage every half hour. These data include the number of position seizures, work volume, average number of positions occupied,

and actual call value or the average holding time/position/call with the figures summarized by chief operator group and for the entire complex. With this data the forcing groups determines the total size of the operator work force and works out the requirements for each chief operator group. Central force administration is particularly important in the TSPS because all operators are a part of one large team but they are divided into as many as nine groups that can be separated by many miles.

Each chief operator also receives operator usage data every half hour for her group as well as for the complex. This information allows her to assess the effectiveness of her own group in comparison with the team and modify the training and supervision of her group.

5.3 *Operator Keyed Trouble Reports*

The TSPS operator, like most operators, is in a unique position of being able to assess the general quality of operation of the telephone network. She can detect operational problems when calls fail to be completed and can judge noise and transmission quality. Moreover, she frequently can detect whether the problems are toward the called or toward the calling party.

Because TSPS is a stored program system, it has a unique opportunity to record these observations efficiently. Any time she encounters trouble, the operator can, by key operations, indicate to the system the problem with the call then connected to her position. She operates the keypulse trouble key, two digits signifying the nature of the trouble, and the start key. The system upon receipt of the two digits makes a simple translation to determine whether the trouble report should be forwarded to the master control center teletypewriter in addition to the direct distance dialing service bureau teletypewriter.

When the printout is made, the trouble code, the reporting position identity, the identity of the trunk and the calling and called numbers are given. It is possible by analyzing repeated reports to locate such units as faulty trunks or to indicate areas that should get increased maintenance scrutiny. Results to date have indicated that this is a most valuable tool and means are being sought to automatically process the trouble information supplied by the operators so that effective response to these reports can be made in minimum time.

VI. PLANNED NEW FEATURES

It is the long range intention of the Bell System to ultimately replace all cordboards with TSP operation. The tedious and repetitious functions

that can be best accomplished through mechanization will be automated. In keeping with this intention, it is expected that additional features will be placed on the TSPS No. 1 for handling those calls that still remain on cordboards. For example, mobile and marine calls, conference calls, and inward traffic to a toll center are likely candidates for addition to the TSPS.

There are plans to further automate certain calls now handled on the TSPS with manual ticketing. Included are automating time and charges record keeping on noncoin calls, semiautomated handling of hotel and motel calls, and providing operator assistance for international direct distance dialed calls.[9] It is likely that features that are not foreseen at this time will be added to the TSPS in future years. The stored program concept facilitates the addition of such features to existing offices.

VII. CONCLUSION

Nine TSPS installations are now in service in the United States. About 30 more offices have had equipment shipped which is now being installed. It is anticipated in the decades ahead that several hundred TSPS installations will be in service.

Experience in the early installations has shown that the TSPS has met its design objective. Those telephone customers who have expressed opinions about TSPS service have been most commendatory because of the speed of service and of the ability of the operator to give her undivided attention to the call. The telephone companies are pleased with this system because it provides simplified engineering, improved maintenance, adequate and timely data for traffic management and significant savings in traffic operating costs. The operators like TSPS because of the attractive decor employed in operating rooms, the modern consoles, and the fact that they can give customers their undivided attention when calls are connected to their positions.

The articles that follow describe many of the features of the system in more detail. Those items that are unique to the design of TSPS are stressed. The material covered in these articles represents the work of many people. This project was supported by many areas of Bell Telephone Laboratories and built upon the techniques of earlier developments such as No. 1 ESS and the crossbar tandem TSP. It also represents a close cooperative effort with the AT&T Company, Western Electric, and those telephone companies that received the early installations, all of whom played a most vital part in establishing requirements and implementing the design.

REFERENCES

1. Freericks, L., "Special Positions for Person-to-Person, Collect and Credit Card Calls," Elec. Eng., *79*, No. 10 (October, 1960), pp. 814–817.
2. Morris, R. M., "Crossbar Tandem TSP," Bell Laboratories Record, *42*, No. 5 (May 1964), pp. 146–151.
3. "No. 1 ESS," BSTJ, *43*, No. 5 (September 1964), pp. 1831–2609.
4. Clemons, D. G., and Little, R. S. "The Piggyback Twistor Store," Bell Laboratories Record, *47*, No. 11 (December 1969), pp. 359–364.
5. Fought, B. T., and Funk, C. J. "Electronic Translator System for Toll Switching," IEEE Trans. Commun. Technology, 3, 1970, pp. 168–174.
6. Baldinger, W. R., and Clark, G. T. "TSPS No. 1: Physical Design," BSTJ, this issue, pp. 2685–2709.
7. Comella, W. K., Day, C. M., Jr., and Hackett, J. A., "TSPS No. 1: Peripheral Circuits," BSTJ, this issue, pp. 2561–2623.
8. Keever, R. J., "Telephone Numbering Plans—Keeping Pace With Domestic and World Service Needs," IEEE Trans. Commun. Technology, *14*, No. 10 (December 1966), pp. 863–864.
9. Scott, L. J. "Overseas Dialing: Yesterday, Today and Tomorrow," Bell Laboratories Record, *48*, No. 5 (May 1970), pp. 142–149.

TSPS No. 1:

Stored Program Control No. 1A

By G. R. DURNEY, H. W. KETTLER, E. M. PRELL,
G. RIDDELL and W. B. ROHN

The Stored Program Control No. 1A processor complex was conceived as a program sequenced control for the Traffic Service Position System No. 1 and for the Electronic Translator Systems. It was designed to be a highly reliable complex using a conservative discrete component hardware design with excellent fault detecting capability and with exceptionally good automatic recovery of call processing when faced with hardware faults.

The design aims, order structure, hardware features, fault detection, diagnostic, and recovery aspects of the system are described. Stress is placed on those features which are felt to be improvements over previous program controlled systems.

I. INTRODUCTION

New telephone services are regularly being conceived which place large demands upon existing switching facilities—demands which are increasingly difficult to satisfy with electromechanical techniques. The use of large high-speed memory and stored-program logic permits the modernization of existing switching functions[1] and implemenation of new services more effectively and at lower cost than by electromechanical means. The stored program control (SPC) No. 1A has been developed as a general purpose stored program electronic processing system to provide a flexible control for implementing new or modernized telephone services. It is a system which is independent of application but with generalized interfaces to which hardware and software may be readily applied in the development of specific application systems.

Bell System electronic switching machines must work reliably in an environment in which substantial noise and temperature variations may be encountered. System processing capacity must economically meet traffic handling needs over the projected life of the system. However, this requirement is generally met more readily with state-of-the-

art technology than the reliability objectives. Thus, the SPC No. 1A represents a conservative design in device technology and processing speed, but an advanced and sophisticated organization for automatic trouble recovery. The emphasis in this article is on the hardware and software organization for maintaining system operation with particular attention given to the maintenance features which depart from those used in other Bell System stored-program processors.

The SPC was developed concurrently with the traffic service position system (TSPS) No. 1, which is designed to improve operator assistance facilities.[2] The SPC is also used in the electronic translator system to replace the present electromechanical call routing translators in the 4A toll crossbar system.

II. A COMMON SYSTEM

2.1 Hardware Organization

The basic concept of the SPC is that of an electronic data processor operating with a stored program to control all functions of its associated application system on a time-shared basis. The general organization of the SPC complex is illustrated in Fig. 1. The complex consists of the arithmetic-logic unit known as the processor, the memory system using "piggyback twistor" stores,[3] several peripheral units required for maintenance of the processor-memory system, and a master control center complex for man-machine interaction.
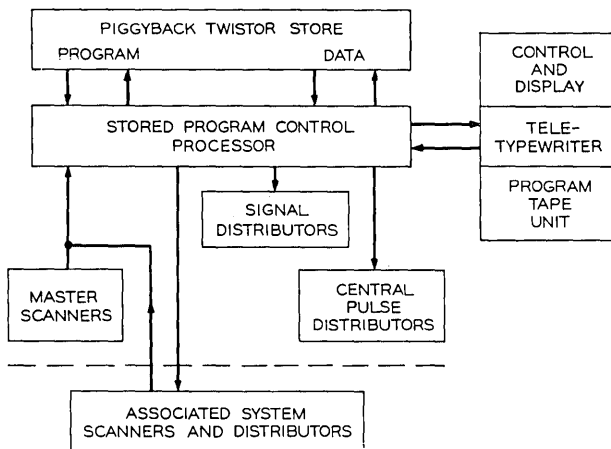

Fig. 1—Basic SPC complex.

To permit the widest application for the SPC, careful consideration was given to the structure and definition of the instruction set. The final choice consisted of 19 basic processing operations and 14 maintenance operations. These represent a compact but powerful order set that is well-balanced for either logical data processing or for control of peripheral hardware. This was particularly important for the SPC which in its inception was not aimed specifically at either one of these fields of application.

The first two application systems using the SPC, the electronic translator system and TSPS, are examples of these two quite distinct processing orientations: logical data processing and peripheral hardware control.

For basic data manipulation, memory access is organized for both word and character. In addition to access to full or half words, individual memory access instructions can, through an option field, specify the reading or writing of any byte. (A byte is an item of i bits positioned j bits from the least significant bit of any half word.) Seven processor index registers are fully flexible, that is, any may be specified for the functions of accumulator, return address, and so on. Transfer tests may be made on the total data or any specified bit of any register. The basic hardware oriented orders are scan and distribute orders. Special purpose orders provide combined operations to permit real-time efficiency for highly repetitive input-output functions.

Data transfer to and from memory always consists of a full 40-bit word plus seven bits of error detection and correction code. The stores are nondestructive readout memories which are used to store both semipermanent and temporary data. Semipermanent data locations are protected from inadvertent writing by lockout circuitry. Protection is defined in increments of 1/16th of a store through wiring straps which may be easily changed in the field. The lockout function may be overridden by program control to permit updating or loading the semipermanent data.

Since the stores are electrically alterable, they provide a single-memory-device system for the generic programs, parameter data, office translation data, and the temporary data scratch pad.

Maintenance of the central processor requires the ability to distribute control signals (that is, to set flip-flops or operate relays). This dictates the need for a central pulse distributor and a signal distributor.[4] To maintain units of the SPC, the system must also perform scan operations to read the states of internal points which requires the use of a master scanner.[4] Points in the matrices of these units which are not required

for the SPC are available for use in the application system. Communication to these units is via private SPC peripheral and central pulse distributor address bus systems.

As a common system, the SPC design does not attempt to anticipate the peripheral addressing requirements of the application systems. Thus, provision is made for a standard binary output to an application system peripheral unit address bus. It is assumed that each application system requiring peripheral unit communication will provide an external translator for any desired translation function.[5] Application system scanner type units share the scanner answer bus system with the SPC master scanner.

The master control center consists of a control and display panel, a teletypewriter and a program tape unit. The control and display provides visual and audible indications of major and minor alarm conditions, as well as lamp displays of the status of the various equipment and buses that make up the SPC complex. It also provides keys for data input to the system and for control of the system configuration of on-line equipment units when manually aided recovery is required. These lamps and keys represent additional functions of the SPC that are implemented through the central pulse distributor, signal distributor, and master scanner.

The teletypewriter is the device by which the maintenance craftsman receives detailed information concerning trouble detection and isolation and through which he instructs the system to perform specific tests.

The program tape unit serves as a secondary memory system that is used initially to load the piggyback twistor memory and subsequently to make large changes in the generic program or office translation data. The capacity of the tape memory is such that many application system functions requiring bulk storage with slow access can be accommodated. For example, it can be used to collect dumps of office data for off-line verification and to retain the original office data during changes. It thereby serves as a backup should restoration be required. It can also be used to contain a library of infrequently used application routines, such as growth testing routines.

Flexibility for growth is quite important. In the SPC complex, the impact of growth centers primarily on the stores and central pulse distributors. As previously mentioned, the signal distributor and master scanner are on a private SPC address bus system and contain adequate matrix capacity for all SPC functions up to a maximum size. Assignments in the signal distributor and master scanner have

been made to allow for maximum growth in the SPC area before allowing assignment points to be used by the application system. Growth for stores is considerably simplified by providing full bus connectorization.[6]

2.2 Software Organization

The SPC generic program provides the necessary maintenance functions for recovery from troubles and for fault isolation in the various equipment that comprises the SPC hardware complex. (Generic program refers to the set of instructions common to all SPC systems.)

Since maintenance of the SPC hardware involves a man-machine interface, the programs which relate to the control and display, tele-typewriter, and program tape unit are also part of the generic program. The maintenance test programs and input-output programs for the SPC equipment function as an integral part of the surrounding software-environment and make very specific assumptions regarding this environment. This environment consists of job administration and certain common service programs. These provide not only a cohesive SPC maintenance system but also the framework for executing a broad range of application programs.

Figure 2 is a general block diagram of the SPC generic program and its interfaces with the application system program. The blocks are arranged from top to bottom to show descending priority levels of program execution. Certain blocks are labeled SPC APPLICATION. This indicates that for this class of program functions generalized interfaces have been established that readily allow integration of application system programs with SPC programs.

Of highest priority in the automatic handling of troubles are the fault recognition programs which are entered by interrupts generated when trouble detection circuits recognize a system malfunction. These programs are required to distinguish non-repeating from repeating circuit malfunctions and, in the latter instances, to remove the faulty unit from service, establishing a working system configuration so that the system's normal tasks may be resumed. In priority ranking, they are concerned with processor, store, and peripheral unit malfunctions. The peripheral unit fault recognition programs at this priority level are concerned with symptoms of hardware trouble which appear im-mediately after signals are sent to the peripheral system. Failure to receive an enable-verify signal or an all-seems-well signal from the peripheral unit indicates such trouble.

Hardware emergency action is another high priority program that is required to facilitate recovery from circuit malfunctions. It functions
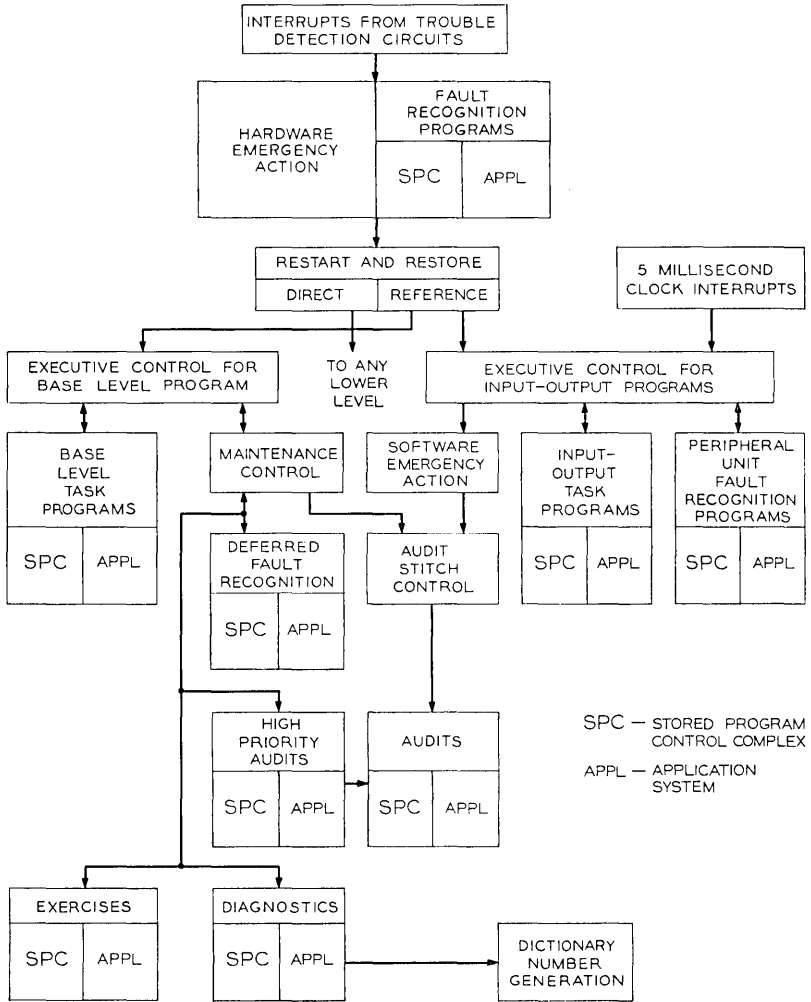
Fig. 2—SPC generic program.

in conjunction with a wired logic hardware sequencer to establish and test new combinations of active processor and active base store until a combination is found which is trouble free. (The base store is the store unit containing the emergency action recovery programs.) It is essential to cope with troubles that destroy sanity of the active control processor system first.

All maintenance interrupt programs return to normal call processing

via the restart and restore program. In most cases, the return is made
to pick up processing at the point at which the interruption occurred.
However, under some conditions this is not safe or feasible; the return
is then made to an appropriate restart or reference point in the job
administration stream where carryover of index register data is not
required. Several of these reference points are defined in the two ex-
ecutive control programs.

An executive control program for input-output work is entered
periodically from interrupts generated by a 5 ms clock.[7] This repre-
sents the next lower major priority class of work. Both SPC and ap-
plication system input-output tasks are scheduled by the executive
control. The SPC tasks include teletypewriter and program tape unit
input-output, execution of signal distributor orders to operate lamps
and alarms, and scanning for alarm conditions.

Another program activity scheduled at the input-output priority
level is an extension of peripheral unit fault recognition. These programs
look for peripheral controllers that have failed to reset to an idle state
in the normal time interval. The SPC program of this class is required
to handle signal distributor controller troubles.

A software emergency action is also scheduled by the executive
control for input-output programs. It makes several tests for the sanity
of system processing to detect degradation that could have been caused
by data mutilation in unprotected memory. The SPC system and
any appropriate application programs normally detect and correct
such data errors by frequently running a series of memory audit pro-
grams. However, in some cases, mutilated data will produce an avalanche
effect that requires immediate and drastic action. The emergency
action program is designed to detect these cases and immediately
execute a series of memory audit and initialization routines to restore
normal operation.

The lowest priority class is that consisting of all the programs executed
at the base level of processing, that is, the noninterrupt level of pro-
cessing. These are scheduled by an executive control for base level
programs which has flexible arrangements for assigning priority and
scheduling task programs. These administrative functions are developed
through the use of parameter tables which provide a simple interface
for application programs.

In general, base level task programs process data collected by input
programs and buffer appropriate data to be subsequently distributed
by output programs. Return address linking is maintained in the output
buffer tables so that the executive control programs can arrange the

stages of processing in sequence for each call while providing time-shared processing of many calls.

At the lowest priority level in the base level of processing, the executive control enters the maintenance control program. This program arranges in sequence all maintenance task programs whose work may be performed on low priority relative to other system tasks. These deferrable maintenance tasks are executed by maintenance control according to priority.

The highest priority group consists of the deferred fault recognition programs. These programs are intended to bring as many units as possible into operation, since the interrupt level fault recognition program (because of constraints on interrupt time) may have to put together a minimum working configuration. The next level consists of high priority memory auditing routines designed to check the integrity and consistency of the data used in the various bookkeeping operations of the program system. The routines are called in at this priority level by programs encountering processing anomalies or hardware failures which might lead to data mutilation.

The diagnostic programs are next in the priority structure. They perform exhaustive tests on the units removed from service by the fault recognition programs and save the pass-fail data for the tests. These data are processed by the dictionary number generation program to produce a number printed on the maintenance teletypewriter. The number is used to enter an appropriate trouble location manual which will identify a circuit pack or packs suspected of causing the trouble. The lowest priority of maintenance programs are the exercises, which include routines that respond to manual requests, routines that periodically test circuit functions which are not in continuous use, and memory audits which are run when all high priority work has been completed. The tables used by the maintenance control program make generous allowance for adding application routines in each of the various priority categories.

Another important function of the maintenance control program is known as audit stitch control. This mode of system operation is requested by the software emergency action program when it detects symptoms of software insanity. Under the audit stitch control, a series of memory audit and software-hardware initialization routines are executed (that is, "stitched") to restore sanity. All call processing activity is suspended during this operation.

Figure 3 depicts the program control plan specifying the functional assignments for the maintenance interrupts (A through G plus K),
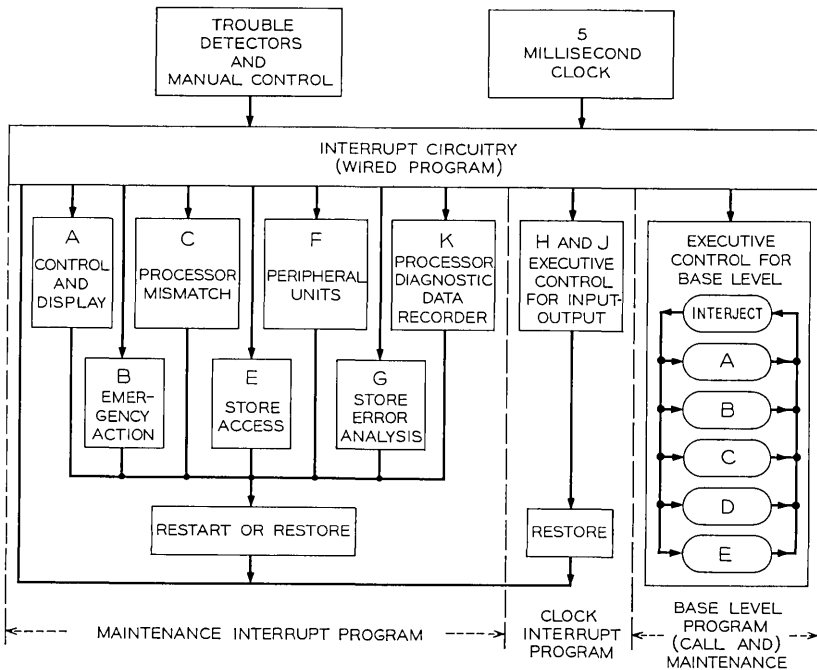
Fig. 3—Program control plan.

the input-output interrupts (H and J) entering the executive control for input-output, and the executive control for base level with its several priority classes of work. Most of the blocks in this figure have been referred to generally in the preceding discussion. However, a few further comments are required to clarify some points.

The maintenance interrupt blocks are designated in terms of the hardware subsystem whose trouble detection circuits generate the particular interrupt level. An appropriate fault recognition program for a given hardware subsystem is entered from the interrupt level associated with the control and display panel where facilities exist for the craftsman to configure the system when automatic reconfiguration is unsuccessful. The G and K levels are associated with specialized functions discussed in detail later.

All base level task programs are assigned to one of the five priority classes (A through E). A priority class of work known as "interject" represents an intermediate priority between interrupt and noninterrupt (base-level) programs. Section IV says more about this class. The

lowest priority base-level class (*E*) is primarily dedicated to maintenance control and the programs which it schedules.

Several techniques have been used extensively to provide a clean interface between the SPC programs and the application system. One approach in the administrative programs is heavy reliance on table control. The tables of these programs allow very flexible task program scheduling and priority assignment. The SPC and application task programs may be mixed together in the tables with SPC assignments ordinarily remaining fixed.

Transfer vector tables (transfer orders to link interprogram communication) are used as a fixed interface for certain necessary linking of SPC programs and application programs. Since the application programs may be located differently in the various systems, this technique avoids consequent variations in the SPC programs.

These techniques make it possible to provide a standard SPC load tape. This is of considerable value to development organizations that use the SPC in an application system development, since they can freely alter the structure of the application programs without having to reassemble SPC programs.

III. HARDWARE

3.1 *General*

To obtain economic advantages from standardized hardware it was decided early in the development of the SPC complex to use low level logic and the existing ESS No. 1 circuit packs wherever possible.[8] Only when there was no existing circuit pack was one designed. Where possible, entire circuits and circuit functions were used from ESS No. 1 with the result that several units are nearly identical.[9] Because of this decision, the central pulse distributor, signal distributor, master scanner, teletypewriter circuits, and communication bus systems are so similar to ESS No. 1 circuits that only minimal descriptions are given here because detailed descriptions appear in Ref. 10.

3.2 *Basic Logic Gate*

Throughout the entire system the basic No. 1 ESS "and not" low level logic gate was used (see Fig. 4). The characteristics are as shown. Worst case circuit design techniques were used in order to insure reliability over working voltage and temperature ranges during the life of the office.

For a moderate sized SPC complex with 20 piggyback twistor stores,

(a)
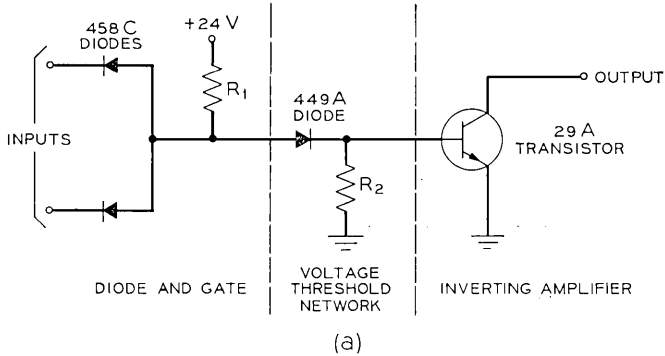
PARAMETER	RANGE	TYPICAL VALUE
SIGNAL LEVELS		
HIGH (1)	4.0 – 5.1 V	4.6 V
LOW (0)	0 1 – 0.5 V	0.25 V
FAN – OUT	1 – 8	4
FAN – IN	1 – 20	4
TURN – ON DELAY	10 – 65 nSEC	35 nSEC
TURN – OFF DELAY	10 – 65 nSEC	35 nSEC
NOISE MARGINS		
HIGH	1.6 – 3.6 V	2.5 V
LOW	1.0 – 1.5 V	1.2 V
CIRCUIT MARGINS		
RESISTOR TOLERANCE	± 20 %	
VOLTAGE TOLERANCE	± 10 %	
POWER CONSUMPTION	37 – 140 mW	76 mW
OPERATING TEMPERATURE RANGE	0 – 55°C	

(b)

Fig. 4—Basic low level logic gate (a) and characteristics (b).

about 50,000 low level logic gates are required. Forty-two percent are used in the stores, 40 percent in the duplicated processors and the other 18 percent in the rest of the SPC complex. In addition to low level logic gates there are many circuit packs requiring higher powered transistors for power supplies, memory drivers, etc.

3.3 *Processor Description*

3.3.1 *General*

The processor, which provides the control for the system, is a computer-like circuit which executes program instructions received from the piggyback twistor stores. The processor cycle is 6.3 μ long. Execution of an instruction requires from 1 to 10 processor cycles. During the last cycle of each instruction the processor reads memory to obtain the next instruction to be executed. Figure 5 depicts the essential portions of the processor which are required for this instruction fetching. The memory address of the next instruction is contained in the 19-bit program address register.

The five most significant bits specify the name of the store to be approached, and the remaining 14 bits indicate the internal address within that store. The processor is thus capable of addressing any one of 32 stores and approaching any of 16,384 locations in that store. The total memory capacity for the system exceeds one-half million directly accessible 47-bit words.

The address image register is a 20-bit register which temporarily retains the address of the most recent communication with the store. If an error occurs during the store communication, the address to
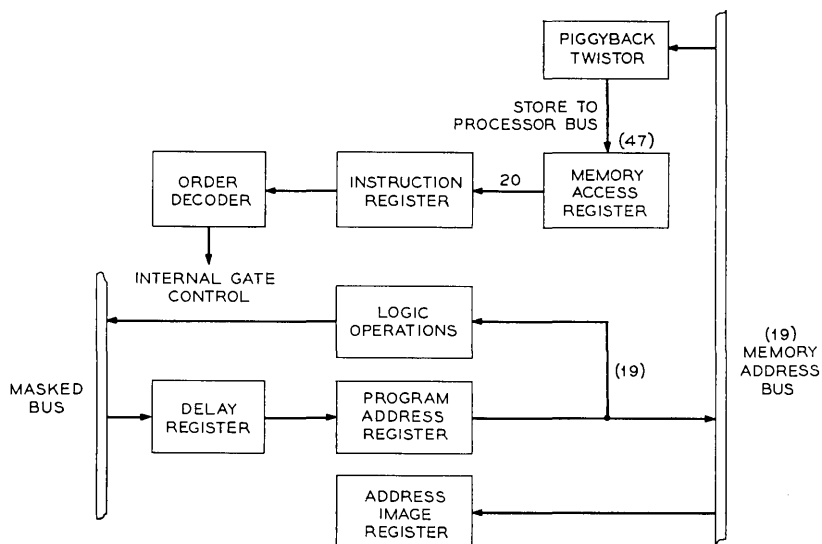

Fig. 5—Program fetching.

which readdressing is required is available in the address image register. The least significant bit of the address image register is used internally by the processor to select a half word on some instructions.

The processor performs logic with a 20-bit combinational parallel subtractor which is also capable of addition and the logical operations AND, OR, and EXCLUSIVE-OR.

The result of a logic operation may be gated to the delay register via a 20-bit bus called the masked bus. The delay register provides temporary storage for instructions which require gating of information from the program address register (or an index register) through the logic operation and back to the program address register (or index register).

The memory access register is a 47-bit register which accepts information from the piggyback twistor stores. The 47 bits consist of a 40-bit word generally containing a 20-bit operation code and a 20-bit address or data field as well as a 6-bit Hamming error correction code and an overall parity bit. The instruction register is a 20-bit register which stores the operation code for the instruction to be executed, and the order decoder controls the gating and information signals necessary for its execution.

In memory fetching, the contents of the program address register are gated onto the memory address bus, and a read instruction is sent to a store. While the processor is waiting for the store response, the program address register contents are gated to the logic circuits where they are incremented to the next higher store address to prepare for reading the next program instruction. The resultant address is gated to the masked bus and into the delay register.

When the store has completed its reading cycle, it gates 47 bits of information onto the store-to-processor bus and subsequently into the memory access register. The most significant 20 bits are gated into the instruction register where they provide inputs to the order decoder until the instruction is completed. During the beginning of the execution of the instruction, the updated program address will be gated from the delay register to the program address register in preparation for the next fetch.

There are five main classes of SPC instructions which make up the SPC program repertoire: internal data manipulation, transfer, memory reading, memory writing, and peripheral input-output orders. These instructions are designed to provide a powerful set of orders for use in a broad variety of application programs. To accomplish this objective, virtually every instruction can be executed with any of the internal

index registers. Any index register may be used for operations such as address indexing, return address storage, logic operations, and peripheral unit enables. Special circuits were designed to permit some of the frequently performed tasks to be accomplished efficiently. For example, circuitry was added to provide access to a particular bit or set of adjacent bits (that is, a byte) with just one instruction. A corresponding packing option is available through another circuit for memory writing orders.

3.3.2 *Internal Data Manipulation Instruction*

An internal data manipulating instruction is used for operations in which data from an internal register are placed into a register after some arithmetic or logic operations are performed on the data. There are seven general purpose registers each consisting of 20 bits which provide storage for information manipulation. For convenience and efficiency all data handling inside the processor is done in the half word size of 20 bits. Two 20 lead digital converted bus systems provide easy access to and from these registers.

Figure 6 indicates the arrangement of the registers and the buses. Access from the registers to the logic operation circuit is provided by the unmasked bus and the argument bus. With this arrangement it is possible to perform a logic operation on any two of the registers by gating one on the unmasked bus and the other on the argument bus and requesting the desired function from the logic circuitry. The results are gated onto the masked bus and into the delay register. From there the result may be transferred to any of the registers.

Although the data words are 20 bits long, operational data themselves normally consist of much smaller bytes. A byte is a quantity defined by a programmer which can vary between 1 and 20 bits. Several of these bytes are then assembled into a 20-bit word for efficiency in storage, and a common task is to unpack a particular byte. The byte of interest is isolated by performing an ANDing operation of the 20-bit word with a register.

Figure 6 also indicates circuits which have been added to permit unpacking of the most commonly used bytes without using a register. The shift and rotate is a combinational circuit which permits shifting or rotating data to the left or the right by any number of bit positions from 0 to 19. The wired mask inhibits transmission of all information except a selected group of right adjusted bits. There are 16 different mask sizes which may be selected. These mask sizes are 1–12, 14–16, and 20 bits. Unpacking is accomplished by gating contents of a register
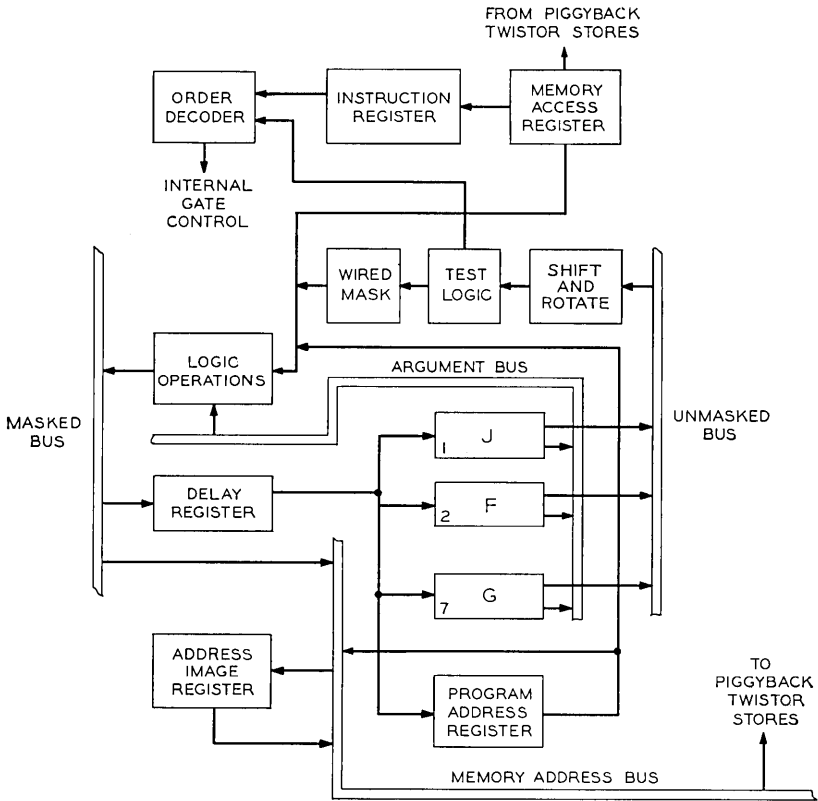
Fig. 6—Internal data manipulation and transfer instruction information flow.

onto the unmasked bus into the shift and rotate where the data is rotated until the byte to be isolated is adjusted to the right. The byte can then be passed through the wired mask and gated into one of the general purpose registers. Some instructions of this type have sufficient bits in the instruction code to specify a logic operation with a second register after the wired unpacking has taken place.

3.3.3 Transfer Instructions

Transfer instructions require a single machine cycle, and are basically fetching operations. If the transfer is conditional the address at which the fetching occurs depends on whether the transfer conditions are satisfied or not.

The least significant 20 bits of the memory access register contain

the address to which program control is to be transferred if the transfer is consummated. This address may be augmented by adding to it the contents of any one of the seven general purpose registers in an operation called indexing. Indexing is permissable on all orders which involve store or peripheral communication.

An address is indexed by gating the right half of the memory access register to the logic operations and at the same time placing the contents of the selected general purpose register on the argument bus. These two quantities are added in the logic operations, the sum is gated onto the masked bus for subsequent gating to the memory address bus, and to the piggyback twistor stores. When a transfer is executed, the present address may be gated into any one of the general purpose registers and later be used as a link back into the present program.

When a conditional transfer instruction is executed on contents of a register, the register is tested in test logic at the same time as the indexed address is being formed. The result of the test is transmitted to the order decoder where the proper gating is established to transfer program control to the new address or to perform a normal fetch for the next sequential instruction.

3.3.4 Memory Reading Instructions

Figure 7 depicts the flow of data associated with a half-word memory read instruction. In a fetching instruction the 20 least significant bits in the memory access register contain the basic store address. During the first part of the first cycle this address can be combined with the contents of any general purpose register to initiate a reading sequence in the store selected.

The five most significant bits in the indexed address specify the name of the store to be approached. Name code 00000 is permanently assigned to 22 sets of registers within the processor circuit known as buffer bus registers. A particular register is specified by the remaining bits of the address. These registers are read by the maintenance programs in monitoring the operation of the processor. These registers contain as many as 24 flip-flops.

The reply from the store or buffer bus register arrives in the memory access register near the end of cycle 1. The least significant bit in the address image register is used to specify which half of the 40 bits in the memory access register is to be used. In the second cycle this data filters through the wired mask any one of the general purpose registers.

A full word memory reading is also available which can read two half words into two separate general purpose registers, but no unpacking is permissable.

3.3.5 *Memory Writing Instructions*

The half-word writing instruction consists of three main operations:

(*i*) Preread data of the memory location which is to be changed,

(*ii*) Operate logic to form the information to be written back into the location, and

(*iii*) Transmit new data to the store.

During the first machine cycle, address indexing is permitted and the resultant address is transmitted to the store over the memory address bus (Fig. 7). The proper mode bits are also transmitted to cause the store to perform a preread cycle. This results in reading a location, the address of which is retained by the store so the subsequent
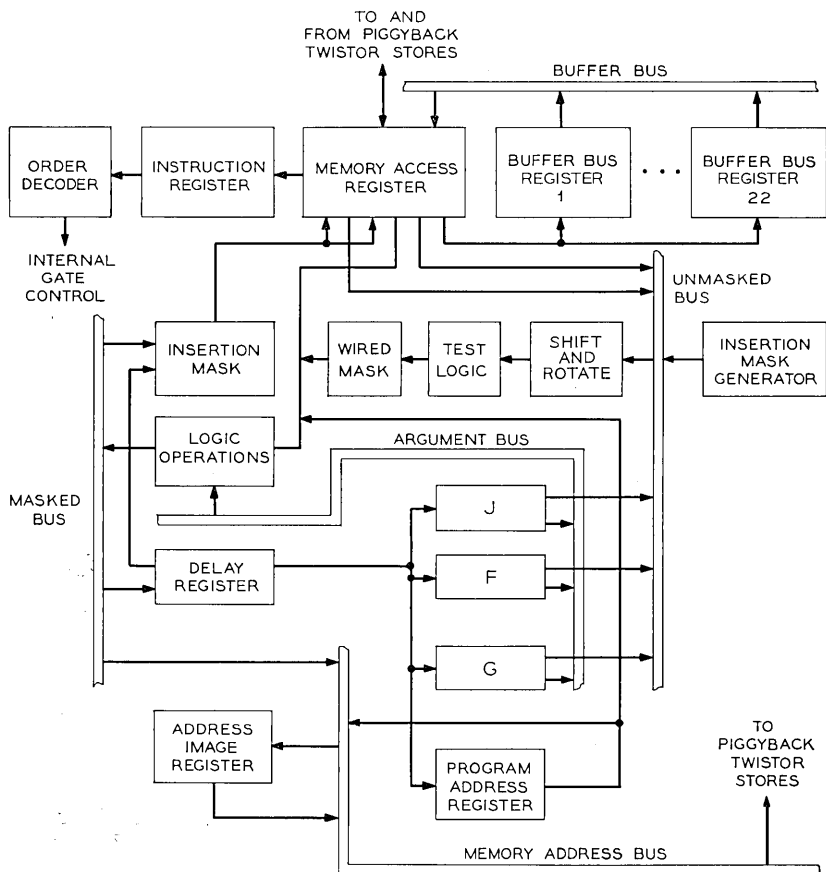

Fig. 7—Information flow for memory reading and writing instructions.

writing will occur at the same store location. The response from the pre-reading arrives in the memory access register near the end of the first cycle.

Writing often requires a packing process in which a byte is inserted in an existing word without disturbing the rest of it. This packing process (insertion masking) is accomplished with dedicated circuitry. The size of the byte is selected by program from the available wired masks.

In the second cycle the register containing the right adjusted byte or word to be written into memory is gated to the unmasked bus through the shift and rotate where it is left rotated to the proper bit position. It then flows unchanged through the test logic, wired mask, and logic operations to the masked bus and into the insertion mask. Here it is combined with insertion mask contained in the delay register and is gated into the memory access register in those bit locations specified from the delay register. This can occur in either half of the memory access register as selected. Hamming and parity bits are generated over this new 40-bit word and the internal store address which is present in the address image register. The data are then transmitted to the store.

The third cycle consists of the normal fetch and the transmission of a "write go" signal to the store after the data to be written has been verified.

When the indexed address contains the store name code 00000 the preread and subsequent write will occur at the buffer bus registers specified by the remainder of the address. Thus these registers may be controlled in the same manner as a store location.

3.3.6 Peripheral Orders

There are four main ac buses associated with information flow between the processor circuit and a unit of peripheral hardware. The SPC-peripheral unit address bus is dedicated to the units which are a part of every SPC installation; the application system peripheral equipment is located on a separate peripheral address bus. The central pulse distributors are located on a dedicated central pulse distributor enable bus, and one of them is activated on every peripheral order. The scanner answer bus is provided for responses from scanner-type units to the processor. This bus serves both the SPC and the application system peripheral units.

Figure 8 illustrates the flow of information for the normal two-cycle peripheral instruction. The address to be transmitted on the address buses is formed by the normal address indexing of the right half of the memory access register and one of the general purpose registers. This

TO AND
FROM PIGGYBACK
TWISTOR STORES

BUFFER BUS

| ORDER DECODER | INSTRUCTION REGISTER | MEMORY ACCESS REGISTER | BUFFER BUS REGISTER 1 | · · · | BUFFER BUS REGISTER 22 |

INTERNAL GATE CONTROL

UNMASKED BUS

| INSERTION MASK | WIRED MASK | TEST LOGIC | SHIFT AND ROTATE | INSERTION MASK GENERATOR |

| LOGIC OPERATIONS |

CENTRAL PULSE DISTRI- BUTOR ADDRESS TRANSLATOR

MASKED BUS

J

DELAY REGISTER
(20)

F

SCANNER
ANSWER BUS

1/n

G

ADDRESS IMAGE REGISTER

PERIPHERAL ADDRESS BUS

PARITY GENER- ATOR

PROGRAM ADDRESS REGISTER

TO PIGGYBACK TWISTOR STORES

STORED PROGRAM CONTROL

TRANS- LATOR 1/n

PERIPHERAL UNIT ADDRESS BUS

MEMORY ADDRESS BUS

Fig. 8—Information flow for peripheral instruction.

address is gated from the masked bus to the address image register where it is stored until it is gated onto the peripheral address buses.

Concurrently with the address indexing operation, the central pulse distributor enable is gated from a selected general purpose register onto the unmasked bus and into the central pulse distributor address translator. The enable is then translated into a 1/8, 1/8, 1/16 code and transmitted to the central pulse distributor. This encoding is performed in the processor to reduce the number of logical units required in the various central pulse distributor circuits to decode the information

to a particular 1/1024 selection. The internal peripheral address in the address image register is pulsed onto the two address buses in the proper time frame so the signal from the central pulse distributor and the address bus reach the chosen peripheral unit in coincidence. The data for the SPC-peripheral unit address bus is translated into a 1/8, 1/8, 1/4, 1/2, 1/2, 1/2, 1/2 code while the data on the peripheral address bus is transmitted without translation. An overall parity bit is generated over the data on the latter bus. During the second cycle of the instruction, checks are made for receipt of enable verify signals and other indications of correct transmission of all data.

3.3.7 Features to Increase System Efficiency

A write operation into the piggyback twistor store requires about 37 microseconds from the time the store receives the information to be written until it is ready to be accessed again. If the store passes internal tests made during the write operation, the store pulses a "write all seems well" signal to the processor about 17 microseconds after receiving the information. A write overlap sequencer enables the processor to begin processing the next instruction while the store is completing the write operation. This sequencer monitors the store for the signal and requests an E-level interrupt if it is missing. The sequencer is active until the store has sufficient time to complete its write operation. If an attempt is made to access a store which has not completed a write sequence, it will not return an "all seems well read" signal. The processor will reaccess the store each cycle until the write overlap sequence returns to normal or access is successful. All seems well read failures that occur with the write overlap sequences inactive result in normal store error sequencer action.

When an interrupt sequence is activated, control is transferred to an interrupt program. Once this interrupt program is in control, it requires the contents of the seven registers to be placed in memory so the state of the processor may be restored just prior to returning to the main interrupted program. To write this information in the piggyback twistor stores would require a relatively large amount of system time for the J level input-output interrupt since it occurs every five milliseconds. In order to save this time, the seven general purpose registers were designed to have two flip-flops for each bit of data. When a J level interrupt occurs, the contents of the registers are gated in parallel into the auxiliary registers. When the J level interrupt is completed and control is being returned to the interrupted, program, the contents of the auxiliary registers are gated back into the main register with no loss of system time.

3.3.8 *Matching Between Processors*

The principle means of detecting malfunctions is through the inter-processor matching system. The processors are normally locked in step executing the same instructions and matching selected data between the two units. The matching signals are transmitted between processors over dc connections to minimize signal delay. Three matches can be taken in each cycle.

Figure 9 indicates schematically the two independent matching units, matcher A and matcher B, which comprise the matching system in the processors. The match control is a buffer bus register which is controllable through a memory write instruction. This register is used to control the match sources which are to be used, the times at which the matches will be made, and the action to be taken on an abnormality. The A match unit in each processor is capable of matching any one of five groups of data against the same information in the other processor. The B matcher operates identically with the exception that it can sample any one of six different groups.
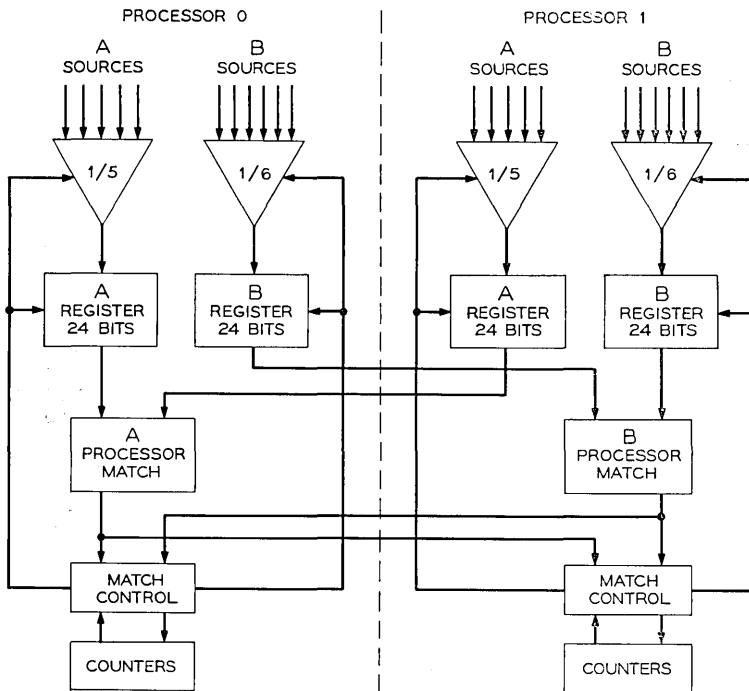
Fig. 9—Matching system.

The matching system may be operated in either the sample match mode or the directed match mode. In the sample mode, a single match is taken in one or both of the matcher units at a previously designated cycle and phase of an instruction under direction of match control circuitry which specifies when matching should occur. This sample match mode is terminated as soon as the match is taken with the match registers containing the sampled data.

In the directed mode, match control establishes a source and times for sampling for each matcher unit. Any of the sources may be specified and the sample may be taken in any or all of the three phases each cycle. These particular sources are then matched at the specified phases of each cycle until the mode is terminated by an abnormality or by program control.

The directed mode is active during normal operation with the unmasked bus and masked bus specified as the sources and a match occurring during each of the three phases. Virtually all the pertinent data associated with the execution of an instruction passes over these buses so a malfunction in a unit will be quickly detected.

Three automatic matching features are available while the matching system is operating in the directed mode:

(i) match on write instruction,
(ii) match on store error sequencer, and
(iii) match on store error correction.

When any of these automatic features are activated by setting a buffer bus register bit, the normally selected source will be automatically overridden for a particular cycle and phase and the memory access register substituted as the source.

When the results of the "matching" operation indicate a malfunction, the match registers and the counters are inhibited from further operation. All of these circuits are registers on the buffer bus and may be interrogated with a memory read instruction. The match registers are frozen with the data which caused the abnormality, a 4-bit cycle counter indicates the cycle of the instruction at which it occurred, and a phase counter records the phase. This information is used by the maintenance programs in detecting and isolating hardware faults.

3.4 *Control and Display Unit*

The control and display circuit is patterned after the No. 1 ESS circuit.[11] It provides a visual indication of the system status through indicator lamps and allows manual control of the system through keys

and switches. Figure 10 depicts the arrangement of the apparatus used in its operation.

There are trouble lamps associated with the system as a whole and with the separate SPC units. When a system or unit malfunction occurs, an alarm is sounded and the corresponding trouble lamp is lighted. Lamps are used to display trouble active and power status. If the system becomes inoperable and cannot recover, manual control may be assumed. The craftsman may force any combination of stores and active processor for the first eight store frames. He may also isolate power from a particular store bus, peripheral bus, or central pulse distributor bus system.

3.5 *Program Tape Unit*

One of the major advantages of the piggyback twistor stores is their ability to change stored program and data electrically from the program tape unit by programmed means.

To load information into the piggyback twistor memories, the program tape unit, under system control, reads characters from its magnetic tape and places corresponding signals on groups of ferrods in the master scanner as shown in Fig. 11. The processor then reads and assembles the characters into a 40-bit word, combines this with the address of the store location which is to be changed, generates a hamming and parity code, and then writes the information with the hamming and parity bits into the proper store. Synchronization of the program tape unit output with the SPC processors is controlled by clocking pulses originating in the processor. The program tape unit also can be used to write information contained in memory on tape. For this type of operation the processor reads the piggyback twistor memory. The 40-bit word received from memory is broken up into five-eight-bit characters, a parity bit is generated over each character, and the characters are transmitted in sequence to the program tape unit over the peripheral bus system.

The format of information on the magnetic tape is shown in Fig. 12. Identity words are used to identify the blocks of data. The tape is prepared and written at 200 BPI density with non-return to zero writing mode. Following each block of characters is a longitudinal parity check character to verify the validity of the data forwarded to the processor. Each block of data is followed by a $\frac{3}{4}$ inch inter-record gap. Reading and writing is done at five inches per second tape speed to synchronize transfer of information with the five millisecond J level interrupts.
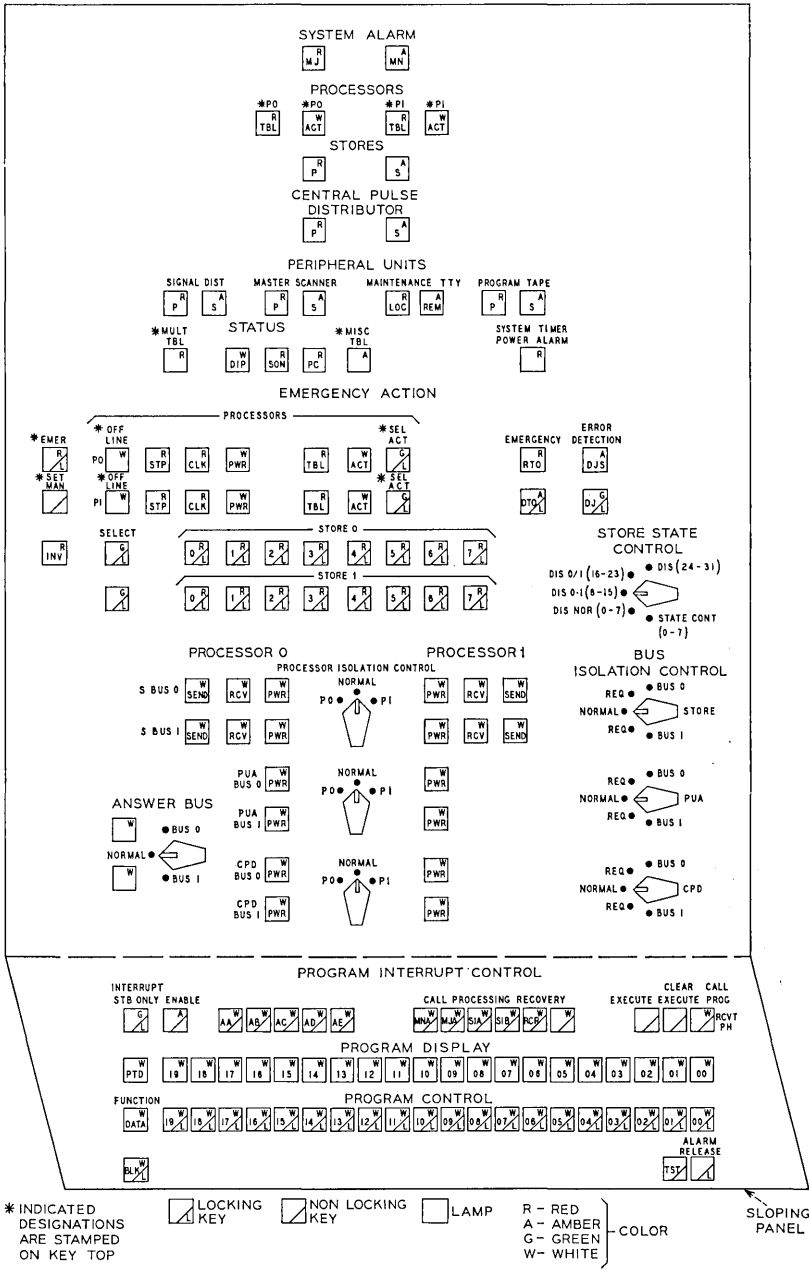
Fig. 10—Control and display.
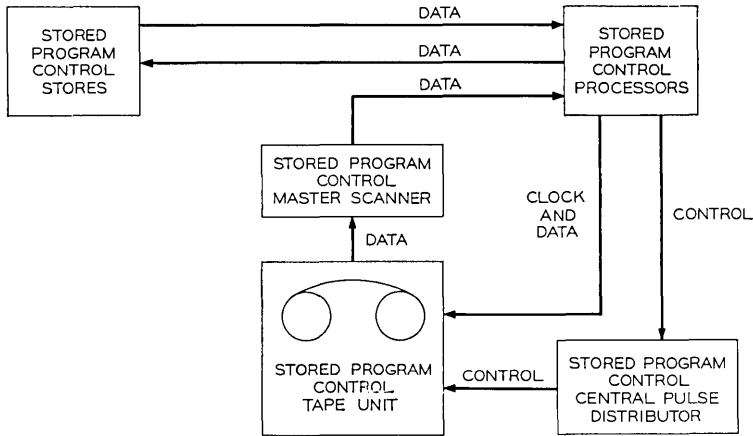
Fig. 11—Block diagram of program tape unit.

3.5.1 *Read Mode*

Since the magnetic tape is written in the non-return to zero mode, the presence of a logical one is indicated by a flux reversal. The read head output consists of bellshaped bipolar pulses corresponding to appropriate flux reversals. These pulses are linearly amplified and then sent through an analog-to-digital converter (Fig. 13) which produces unipolar square pulses with leading edges at the corresponding peaks of the original bipolar pulses. After delays for bit deskewing and synchronization, the contents of the buffer are gated into a normal register slot; a general reset then occurs which resets the buffers, and the distributor is advanced to provide a new slot for the next character to be read from tape. The register has six slots each containing the nine bits of one character. Words are picked up by the processor at five-millisecond intervals.

3.5.2 *Write Mode*

The write mode is initiated by the processor. When the processor desires to transfer information from the store to the program tape unit, it signals via central pulse distributor points to condition the program tape unit circuit for writing and starts the tape transport running in the forward direction. The processor then transmits unit five or six characters sequentially to the program tape. These characters are held in the six normal register slots. The processor then signals the program tape unit to begin writing a record block. One character is written on tape

FORWARD TAPE MOTION

DIRECTION OF READING

Track No.	INTER-RECORD GAP	IDENTITY WORDS			FIRST DATA WORD					INTERMEDIATE DATA WORDS	LAST DATA WORD							INTER-RECORD GAP	
8		F	F	N19	A11	A3	035	027	019	011	03	035	027	019	011	03	←—0.020"—→	P2	←---0.750"---→
7		F	F	N17	A9	A1	033	025	017	09	01	033	025	017	09	01		P2	
6		F	F	F	N15	A7	039	031	023	015	07	039	031	023	015	07		P2	
5		F	F	F	A14	A6	038	030	022	014	06	038	030	022	014	06		P2	
4		F	F	F	A13	A5	037	029	021	013	05	037	029	021	013	05		P2	
3		P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1		P2	
2		F	F	F	A12	A4	036	028	020	012	04	036	028	020	012	04		P2	
1		F	F	N16	A8	A0	032	024	016	08	00	032	024	016	08	00		P2	
0		F	F	N18	A10	A2	034	026	018	010	02	034	026	018	010	02		P2	

RECORD GAP NO. 2

RECORD GAP NO. 1

RECORD GAP NO. 3

SUCCEEDING RECORD BLOCK

TRACK NUMBERS

REFERENCE EDGE

TRANVERSE PARITY TRACK (000 PARITY)

LONGITUDINAL CHECK CHARACTER (EVEN PARITY)

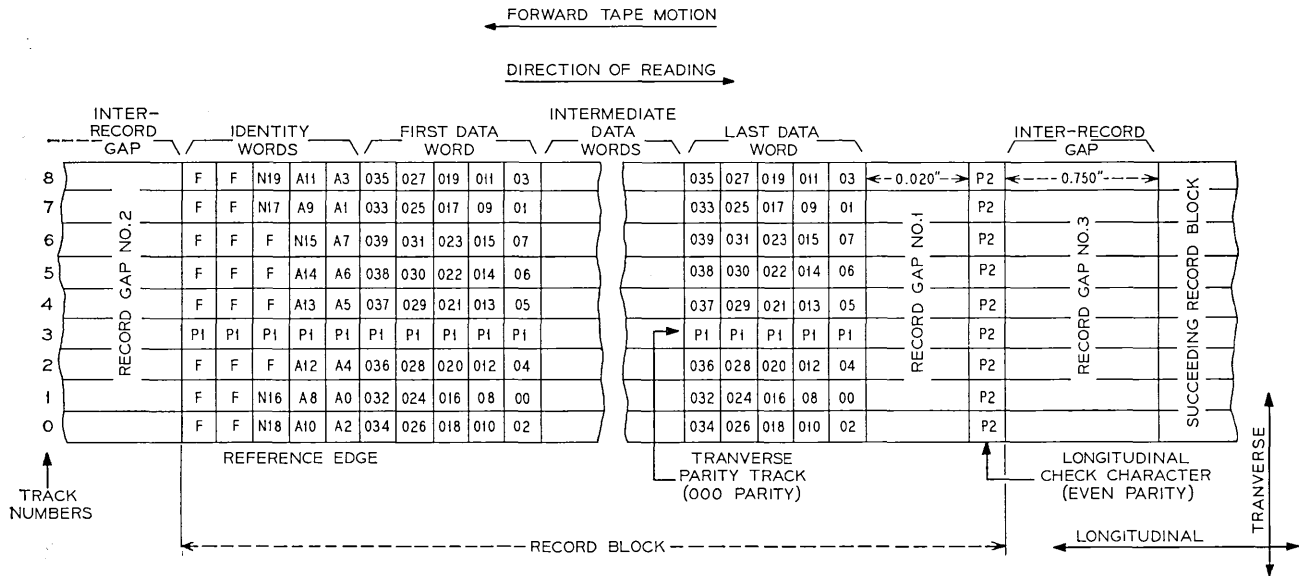TRANSVERSE

RECORD BLOCK

LONGITUDINAL

Fig. 12—Format of magnetic tape.
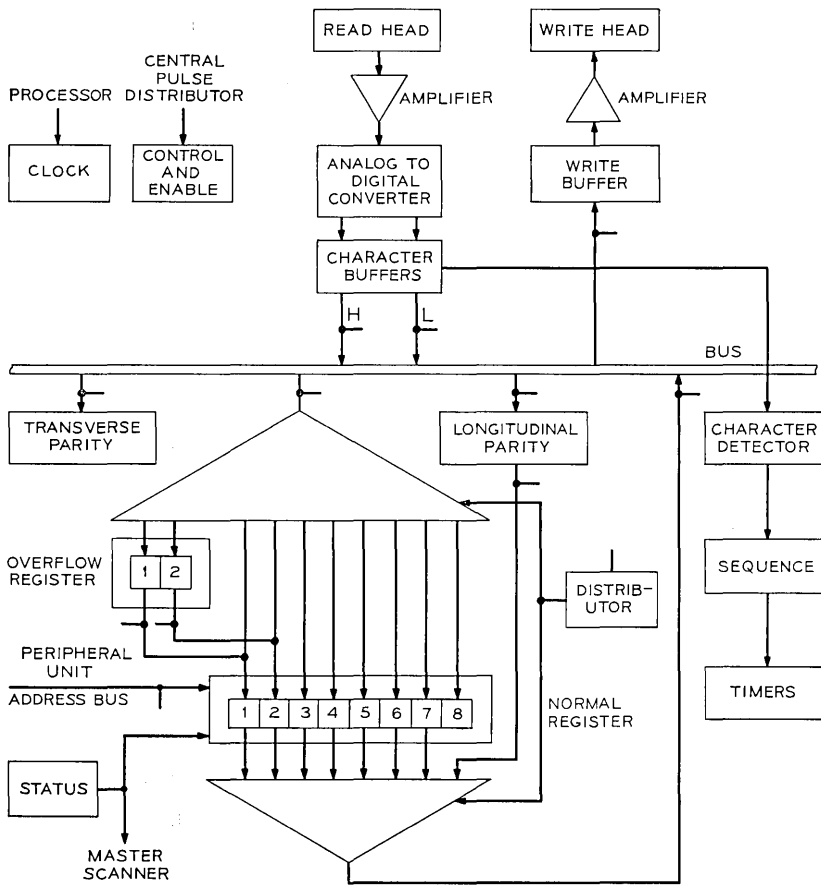
Fig. 13—Program tape unit.

each millisecond. Before each character is written, it is checked for transverse parity. Five normal register slots are emptied each five milliseconds, and the processor fills emptied slots with new characters. After a maximum of 151 words have been written, the processor will signal the program tape unit to end the record block. The processor times for the interrecord gaps.

3.6 Teletypewriter Circuits

Teletype circuitry as used with the SPC is almost identical to that used in No. 1 ESS and needs little explanation. The interface buffer

between the teletypewriter and the processor is a low level logic circuit connecting to the peripheral bus. In operation, an enable signal is received from the central pulse distributor to enable the teletypewriter to accept the information which appears on the address bus. The character or control information for the teletypewriter is received in parallel from the peripheral bus and is shifted out serially toward the teletypewriter at a pace compatible with the mechanical speed of the machine. The teletype buffer receives a character when a message is being typed. The design of the interface buffer is arranged to simplify diagnostics of both the electronic and electromechanical parts of the teletypewriter.

3.7 Communication Bus Systems

To provide communication between the component parts of the system, an ac bus transmission system is used. These are parallel twisted pairs running from unit to unit in particular groupings called buses. They carry information words in the form of 0.45 microsecond pulses. Pulse generating and receiving circuitry is designed to produce recognizable signals at the receiving circuits of all the units connected to the bus. Bus impedances are about 50 ohms at the driving point. Since only one order may be sent in a 6.3 microsecond cycle, the duty cycle for pulsing is less than 10 percent thus allowing operation of the bus receivers without dc restoration.

The ac bus driving transformers are well shielded and the system is balanced with respect to ground. Contact protection networks are extensively used in relay circuitry to prevent generation of transient voltages. Paired leads are used wherever possible to minimize coupling of noise into the bus system.

Peripheral unit bus lengths can be as great as 450 feet. Store buses are restricted to 100 feet in order to minimize propagation delays for proper store/processor communication.

3.8 Central Pulse Distributors

The central pulse distributor is used to provide the SPC processor with directed access to many points within the system which require fast response to the instructions being transmitted. The unit is a translator decoder which takes an equipment address and provides a unipolar or bipolar pulse output on one out of a possible maximum of 1024 points. Address storage registers at the input of the central pulse distributor connect to parity error detection circuitry which check the input information for accuracy. The address registers simultaneously prepare a path through a matrix to an output point of the central pulse

distributor. If the error check is satified, a pulse is applied to the apex of the matrix and appears at the selected output point. If internal checks for correct operation are satisfied an all seems well pulse is returned to the processor indicating the SPC has functioned properly.

3.9 Signal Distributors

The signal distributor is accessed from a peripheral bus and therefore uses low level logic circuitry as an interface with the system. Internally, however, nearly all logic is performed by relays. The selection matrix consists of relay contact trees which form a selected path under control of the input address information. The signal distributor is intended to operate magnetic latching relays, in response to instructions distributed by the processor. The apex of the relay tree matrix can connect to a −48 volt signal or a +24 volt signal to operate or release magnetic latching relays connected to the output points. Operation of the magnetic latching relay causes a pulse to be generated along the energizing path back through the matrix. When this signal is detected, the current to the relay is interrupted and the input registers are reset.

3.10 Master Scanner

Master scanners are used throughout the system for supervisory input from the various equipments to the SPC complex. A master scanner contains a matrix of ferrods connected in such a manner that they can be interrogated in groups of 16 to determine whether or not the ferrite material is magnetically saturated. Saturation of the material occurs when the point to be monitored causes current to flow through a winding surrounding the ferrite rod. A saturated ferrod produces a very low output in the output winding when interrogated, a non-saturated ferrod produces a high output. The output of the ferrods is transmitted to the SPC on the scanner answer bus for the use of the processor, in the handling of calls or execution of other work.

IV. ORGANIZATION FOR RELIABILITY AND MAINTAINABILITY

This section is concerned with those aspects of the SPC No. 1A hardware design and organization which are provided specifically to facilitate maintenance and achieve the over-all system reliability objective[12,13]. Thus, it describes the equipment redundancy scheme, the arrangements for switching, access to the various equipments, special circuit facilities for emergency recovery from troubles, maintenance instructions, and so on. Sufficient description of the workings of each

function are given to indicate the basis and validity of the choices. However, the section does not discuss the complete strategies for recovery from trouble conditions. Rather it is intended to provide the background for such a discussion which will be given in the following section on the SPC program.

4.1 Redundancy

Duplication of subsystem equipment units in the SPC is provided for dependability rather than for an increase in the traffic handling capacity of the system. The memory subsystem consists of two store groups, each of which contains numerous information blocks (or stores). Each store of a particular store group has its corresponding image in the opposite store group. The stores contain two distinct categories of information: semipermanent data (program and parameter), and temporary data which may be intermixed in a store. The processors normally run in parallel, synchronously executing the same program, each from its own associated store group. Basic processor store communication is achieved with a partially dedicated bus system using a hamming code plus parity scheme for error detection. This system is arranged in such a manner that a store can only receive and answer over one bus, and each bus of the duplicated bus system has access to only half the stores. At the processor end of the bus, the static control of send and receive modes between each processor and the store buses is augmented by dynamic switching of the store answer bus by either or both processors using a store name match circuit. This program controlled circuit informs a processor when only one of the duplicated stores is in service and from what bus the answer can be expected.

4.2 Store Communication

4.2.1 Control

Generally, all communication buses are completely switchable at the SPC processor. Hence, various SPC processor communication bus modes can be established by means of unique configurations of the buses and processors. The output from a flip-flop in each processor dictates whether or not that processor is active. Communication channels to peripheral units and central pulse distributors are normally established only for the active processor. Since a change in the status of the system is established by means of the central pulse distributors, the active processor is in command of the system configuration.

The active processor selects the bus configuration for the appropriate

processor-store communication mode and thus dictates the treatment of instruction and data communication to and from the stores. It can send on either or both buses while the standby can only send on a bus not being used by the active unit. Normally, each processor communicates with a separate copy of memory (fully independent access); but store communications control is provided to allow flexibility in choosing different bus configurations.

Some of the functions of the store communications control are:

(i) to specify the active processor store bus,
(ii) to allow each processor to independently communicate with a bus,
(iii) to allow the standby to receive from the active processor store bus and deny the standby write access,
(iv) to allow the active processor to send over both buses (implied denial to standby), and
(v) to allow both processors to receive from both store buses.

4.2.2 Store Failure Bus Modes

When the system has a single defective store, the system is reconfigured to establish a mode in which the active processor communicates with the good store group. The standby processor performs as dictated by the store communication control except when communicating with the defective store on its associated bus. When the standby processor addresses the defective store, it will momentarily switch its receiving bus so as to receive from the good copy. This switching mode is established by a program action that places the name of the defective store in a name code recognizer register in each processor. A program controllable bit is provided which specifies the receive bus that should be used by both processors when a match of the defective store name occurs.

If a second store failure occurs when one store in the system is out of service, the system can still function as long as the second failure is not in the mate of the off-line store. When there are two or more stores inoperable and all are on the same bus, the name code recognizer circuits will not be used. The processor-store bus mode which is used causes the active processor to transmit over both buses with both processors receiving from the active bus, i.e., the bus containing the full complement of stores.

When a single store is out-of-service on each bus (not mates), each processor uses its name code recognizer. Each processor transmits and receives over its respective bus except when an inoperable store is ad-

dressed. At this time, the processor which is addressing the defective store will receive from the other processor's bus.

When there are more than two stores defective with only a single store defective on one bus, the name code recognizer circuits of each processor are used to specify this store in both processors. The active processor is configured to transmit over both buses; and both processors are configured to receive from the bus containing the single defective store, except when that store is addressed. Both processors then receive over the other bus, i.e., the bus which has the good copy. This mode keeps the good stores on the standby bus updated although their data is not used.

When there are two or more stores defective on each bus (not mates), it becomes necessary to operate the store complex without redundancy. The action prescribed is to shut down one copy of the remaining redundant stores and to place the system in a mode in which the active processor transmits over both buses and both processors receive over both buses. In this mode of operation, the store name code recognizer circuits are not used. Any subsequent store outage during this mode will necessitate use of the emergency action facility to recover a working system if sufficient working stores remain. This mode of operation is used as a last resort to keep the system running.

4.2.3 *Selective Store Addressing*

The operational programmer who uses the basic instructions for writing data processing routines assumes troublefree operation and ignores subsystem duplication. The maintenance programmer, however, requires some other orders in addition to the operational instructions. These additional orders are used in fault recognition, diagnostic and routine exercise maintenance programs.

Maintenance instructions provide access which is either inconvenient or impossible to obtain with combinations of normal orders. Indirect paths to the desired points sometimes exist via the normal instructions but require reconfiguration of the interconnections between subsystems.

To test store buses, stores, or processor access to a particular store bus, selective addressing of either the 0 or 1 subsystem is accomplished within the processors by a special routing control. This control works in conjunction with a special set of instructions to allow selective addressing of stores, bus testing, selected access to internal points within a store or processor, etc. The order control signals are transmitted from that processor selected by the maintenance programmer in setting up the special routing control. Subsequently, both processors receive the reply.

Hamming, parity, and all-seems-well (ASW) checks are made by the processor(s) when executing maintenance instructions. But, since troubles may be expected to occur during these tests, no corrective action is taken. All responses to store errors are inhibited so that data containing errors may be analyzed.

Special bus testing maintenance instructions are provided in conjunction with special store hardware to test the input-output control leads between the processor(s) and the store complex.

4.3 *Processor Reactions to Store Errors*

In the fully duplicated SPC system, single errors are corrected using a Hamming error correction facility. The results of correction are then compared by the processors, and, if the match is successful, processing continues. If not, a reread is performed, because failure to match after correction indicates the original error was an odd, multibit error.* Reread failure now leads to a processing interrupt, resulting in store system reconfiguration. With any configuration, double errors, address errors, and ASW read errors immediately cause a reread; and reread failures lead to a store interrupt. Failures to write immediately lead to an interrupt.

With at least one processor or store out-of-service, the Hamming error correction facility is inhibited, because, on some or all reads, matching is not available to detect false error correction. Consequently, single errors are treated like double errors. If store errors which cause interrupts occur and the failing store cannot be removed from the system because its mate is faulty, the system's ability to function with single error correction must be checked. If it can be proved that the system is making only single errors (through use of test word reads), the system maintenance program will attempt to run the system with single error correction. Otherwise, manual intervention is required.

4.4 *Processor(s) Sending to the Store System*

A preread feature is provided on all write operations so a Hamming check can be made to insure that the internal address read from the store is the one intended for the write. The store retains the preread address for the write operation. Protection against erroneous writing into another store is provided by a parity check on the store name (higher-order bits of the store address). Because of the preread check, erroneous writing in the desired store is essentially limited to writing

---

* The Hamming code detects and corrects single errors and detects all double and some higher order even errors. It detects all higher order odd errors but treats these as though they were single errors.

erroneous data at the correct address. Invalid data put into the correct store at the correct address should be detected by the Hamming or match circuitry on a later read of that address. If the other copy exists, the processor(s) can continue without difficulty. In the fully duplex mode of operation, the Hamming check over the data and address field is backed-up by a processor match of the data returned independently to each of the two processors.

4.5 *Peripheral Communications*

SPC peripheral unit input-output orders use an enable code to identify the unit which is to receive the order. Certain bits of the code select a central pulse distributor and provide address data used by the central pulse distributor to select an output lead over which the central pulse distributor will send an enable signal to the particular peripheral unit. The data being sent to peripheral unit (which includes an internal selection code) is routed to a particular peripheral unit controller of a duplicate pair, depending upon which peripheral unit and central pulse distributor address buses are selected in the enable code. The enable signal from the central pulse distributor allows the information to be gated from the address bus to the proper peripheral unit controller.

Should any central pulse distributor, bus, or peripheral unit controller fail, there is sufficient redundancy to allow access to the peripheral unit via another route. This requires a change in the associated enable code(s). Most of the bits for the enable code in a peripheral unit are fixed. However, the two bits which control the choice of the peripheral unit address bus and the choice of central pulse distributor (and thereby indirectly the choice of a peripheral controller) vary as trouble conditions affect the status of these units. These two bits are often referred to as the "routing" bits. Since they reflect the status of the peripheral unit system, maintenance programs are required to keep the entire complex of enables updated with each change of the system.

4.6 *Maintenance Control of System Sanity*

When an error in the store containing the interrupt programs leads to an interrupt, there exists the possibility of system insanity. Two defensive mechanisms are used here. First, store communication at the interrupt level is limited to the store containing the interrupt program until the program has established that the system is operating properly. This protects system data. Second, an attempt is made by hardware logic to provide the system with the good copy of the interrupt program. This function is performed by conditional bus switching at the time of

interrupt to the store bus *not* indicating the error. This function can be inhibited (under program control) if there is only one copy of the interrupt program available to the system and if that copy is in the store on the active bus.

4.7 *Emergency Action Functions*

The purpose of the emergency action circuit is to monitor system sanity through a variety of independent timing circuits, and to provide a sequencer for selecting a new hardware configuration if these circuits detect trouble.

A timer is provided for monitoring system sanity at all times. This timer, called the long timer, is active whenever its associated processor is active. This circuit times for 630 milliseconds; if, at the end of this interval, the timer has not been properly administered, it will cause an emergency action B-level interrupt. Two independent program controllable timers are activated by the emergency action circuit when a system malfunction is detected. These timers must be reinitialized periodically or the emergency action circuit will create an emergency action B-level interrupt. These timers can also be used to provide defensive protection in cases where a program action might result in an insane system configuration.

The emergency action sequencer controls the reconfiguration of the system during an emergency action resulting from the emergency action B-level interrupt by sequentially selecting combinations of processor and store buses. If all combinations of processor and store bus have failed to pass the tests administered by the emergency action program, an emergency action alarm will be activated. This indicates to the maintenance personnel that the system cannot recover.

An emergency action program is provided to complement the function of the emergency action sequencer. This program administers a series of hardware tests to determine the acceptability of the sequencer established configuration. These actions are discussed in Section 5.3.3.

V. DESCRIPTION OF THE SPC PROGRAM

5.1 *Maintenance Functions*

This system depends primarily on hardware-checking circuits for trouble detection during operation. When a trouble detection circuit locates a problem in the system, it notifies an interrupt circuit. The interrupt circuit immediately stops operational program processing and transfers control to a fault recognition program associated with the

particular type of trouble indication. The functions of the fault recognition programs are to distinguish nonrepeating troubles (errors) from repeating troubles (faults) and, in the case of faults, to quickly determine an operational system configuration, establish it by switching out faulty units, and then return to operational program processing.

Fault recognition is the most important maintenance function in a real-time system. The minimization of time taken from the operational processing function and an extremely high probability of returning to processing with an operational (sane) system are the critical objectives which any automated maintenance technique must satisfy. Diagnostics with "fine" resolution are of secondary importance.

5.1.1 Processor Errors

The processor retrial program is designed to qualify a processor trouble indication as an error or a fault. The program also interrogates critical areas within the processors; and, if they are found to be faulty, it isolates the fault to the active or standby processor, records pertinent processor error information, and initiates actions required to return the system to normal operation at the conclusion of the program.

Since errors are expected to be more frequent than faults, and, as it requires considerable time to completely check both processors, the mismatch is first assumed to be an error. As the processors inhibit destination register gating on mismatches, it is possible to re-establish the state of the processors at the beginning of the interrupted order and re-execute most instructions. To determine if the mismatch was caused by a fault, the failing instruction is unwound and control is returned to the interrupted program. In the process of preparing the processors for re-executing the failing instruction, a hard fault may again make itself evident. In such a case, retrial of the order is unnecessary and actions will be initiated immediately to remove the faulty processor from service. If the instruction retrial causes another C-level interrupt, the mismatch is presumed to be a fault, and control is transferred to a complete check control routine in the processor fault recognition program.

The processor retrial program maintains error counters based on the results of processing mismatches. If any of the error counts become excessive during a given time period (implying an undetected processor fault), control is transferred to the complete check control routine. Otherwise, control is returned to the interrupted program for retrial of the failing instruction.

When a mismatch occurs, the C-level interrupt hardware sequence

starts the 2-millisecond and 40-millisecond emergency action timers (See Section 4.7). If the active processor has a fault which impairs its ability to properly execute the retrial program, a time-out will occur and control will be switched through a B-level interrupt to the emergency action program.

5.1.2 Store Errors

Two basic facilities are provided in the maintenance programs for dealing with store errors. One is essentially a data collection facility for gathering pertinent information regarding the nature and source of store errors. It is expected that this will be useful to the maintenance craftsman for correcting problems that are causing infrequent but generally consistent store errors. The second facility is an automatic error analysis and reconfiguration strategy that is aimed at the problem of error bursts. In this case, the error rates are so high that immediate and automatic recovery actions are required to avoid severe system penalties.

The G-level interrupt is used to invoke the program that performs store error bookkeeping. This interrupt occurs after any single error is successfully corrected or after any multiple error is eliminated on a reread. The G-level interrupt mode is normally active. However, it is deactivated when certain maintenance programs are running to avoid undesirable interaction. Also, a program governor action is present which limits the number of G-level interrupts to approximately two every two seconds. Allowing more interrupts would probably only generate redundant data and cause an excessive overhead (real-time) penalty.

The G-level program records counts of the various types of errors (repeating single, nonrepeating single, multiple) for each store, internal store addresses for the first 15 errors of each type, and the failing bit for the first five occurrences of repeating single bit errors. The tables, if nonempty, are printed out on the teletypewriter every hour and then cleared.

If transient single errors are prevalent, the craftsman can request a special matching mode that matches the received data before error correction occurs. The resulting mismatch on an error will invoke the C-level interrupt program which generates a printout that identifies the store, the address, and the failing bit.

The error analysis program which is concerned with error bursts or sustained high error rates performs a periodic scan of hardware single and multiple error counters. If either of these counters overflows in a

specified interval, the error analysis mode is initiated. The analysis program collects error data for an additional period of time and attempts to determine from this data whether the errors are caused by store, store bus, or processor trouble. If the analysis is successful, it removes the noisy unit from the system configuration and identifies the unit by a printout. It also causes an immediate dump of the G-level store error tables which should provide useful information for localizing store troubles. A diagnostic is requested for a suspected noisy store, but the store is not automatically returned to service if the diagnostic passes. Because of the intermittent nature of errors, it is likely that the diagnostic will pass. Thus, the system provides printouts of all pertinent data on the problem and leaves the decision to place the unit in service to the judgment of the craftsman.

5.1.3 *Peripheral Unit Errors*

Malfunction of an order sent to a peripheral unit is detected through immediate checks made on the communications with the units and by a gross check of whether the unit successfully completed the desired operation. The immediate checks include: a central pulse distributor execute return signal indicating that the proper central pulse distributor was enabled; a central pulse distributor all-seems-well signal indicating normal functioning of the central pulse distributor; an enable-verify signal indicating that the peripheral unit received an enable signal; a scanner all-seems-well signal indicating successful readout of a scanner row (in the case of a scanner); and, in the case of the application peripheral address translator, an all-seems-well indicating successful address translation for data transmitted to the application peripheral unit bus. The signal distributor provides a gross check of normal operation by indicating through scan points that it has returned to an idle state after executing the order.

Failure to receive the immediate check signals result in an F-level interrupt. The fault recognition program at this level handles all problems associated with high-speed peripheral orders; i.e., those not directed to units involving relay operations or other such long-work-cycle operations. Discrimination of errors from faults is derived by repeating the high-speed order in the F-level and observing whether the failure indication is repeated.

For the slow operating units such as the signal distributor or similar applications system equipment, retrials are performed by a J-level program. At a later time, the order that caused the F-level is retried in J-level to determine whether the failure was due to an error or a

fault. Troubles detected by checks within the signal distributor or similar frames are indicated to the programming system through Master Scanner ferrods. These ferrods are checked in J-level every 25 milliseconds. This check occurs just before an order is sent to the unit by the peripheral order buffer execution program. The failing order is then retried in J-level. This retry will determine whether an error or a fault occurred.

The procedures described above apply to all SPC peripheral units except the teletypewriter and the program tape unit. Maintenance on these units is initiated by the maintenance craftsman. The strategy applied to application peripherals will depend on the characteristics of the device.

With any of the units discussed above, a successful retrial will result in immediate termination of activity. The program involved in error detection may increment a counter or print a message indicating the nature of the error, but no further actions will be performed. The suspect unit is returned to normal service, and program execution returns to its normal schedule.

5.1.4 *Processor Faults*

In the SPC No. 1A system, the processor fault recognition and processor diagnostic programs have been integrated. This method assumes that, with sufficient match access, fault recognition can be a subset of diagnostics. That is, those procedures which test the logical capability of a processor also provide results for diagnostic resolution. The programs will be discussed as separate entities, but one should keep in mind that the test sequences are one and the same. Only the control programs are separate entities.

For repeating troubles, the processor fault recognition program determines which processor is faulty, switches this unit out of service, records significant error information, and places a request for a diagnosis of the faulty unit. Because it runs during the C interrupt level, the fault recognition program checks only those circuits which could have caused a mismatch and those peripheral system communication circuits which may not have been in use but will be employed in the new active processor. A complete analysis of the faulty processor is made by the processor diagnostic program at a later time. It is important to understand that the objective of fault recognition is to find a working configuration. This means that the faulty processor might be active and diagnosing the good processor. This is possible when faults are in areas such as the following: processor-external bus
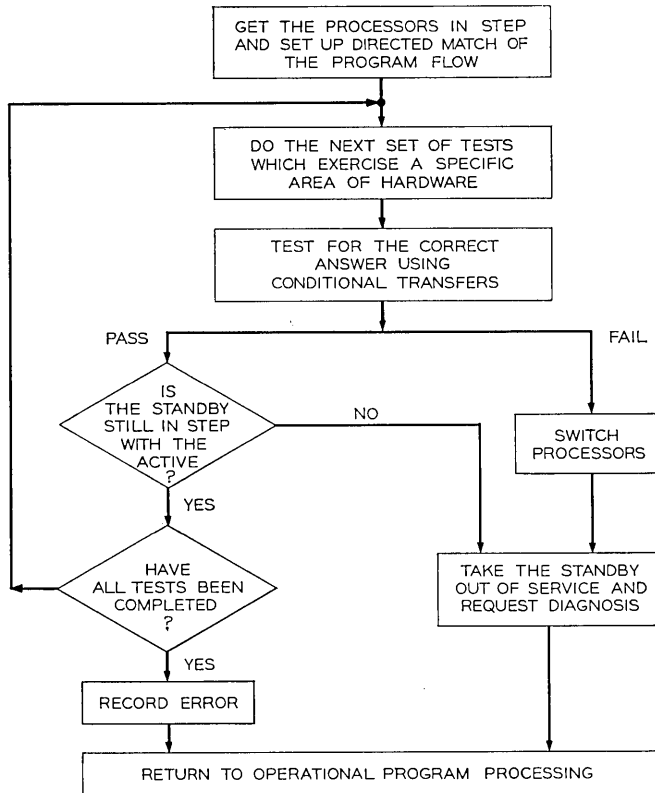
Fig. 14—Processor fault recognition test mode.

communication circuits not presently in use by the existing configuration, the match circuits, and processor-processor interfaces. The diagnostic program has been designed to detect and correct for this anomaly.

The basic program flow technique used within test phases in the fault recognition program is shown in Fig. 14. The tests are designed as if each processor is testing itself. Tests consist of data manipulation operations which check for the proper circuit response followed by conditional transfer orders. If the active processor fails, it will switch processors (which it alone can do by program). The faulty unit, now standby, will be removed from service and the diagnostic program requested. If the active passes, checks are made to see if the standby is following the operations of the good active unit (by examining the match circuits). If the standby is found to be out of step, it is removed

from service. This testing process is continued until all tests have passed or a faulty unit is found.

The fault recognition program is entered by the retrial program when that program decides that a hard fault exists in the system (i.e., two consecutive C-level interrupts occur at the same program location) or when the retrial programs' error counters have exceeded their specified thresholds. The processor fault recognition program is also entered via a B-level interrupt from the emergency action program after a normal processor switch to test the new configuration. In this case, the processor fault recognition program performs internal tests (sanity), establishes and tests new store configuration, tests peripheral unit communications and configures it, if necessary, and returns the system to normal operation.

If the program was initiated by a normal processor switch or by a processor mismatch, the standby processor is made to operate in step with directed matching of the masked bus and unmasked bus. The standby processor is stopped whenever it disagrees with the active processor. If the program was initiated by an emergency action, only the active processor is tested and only the basic sanity tests are performed.

The sanity tests include arithmetic and logic functions, transfer decision, writing and insertion mask, store error detection and correction, and instruction decoding. The completion of the above tests indicates that the active processor has passed the tests. The store name clamp, which limits processor access to the base store, is removed before further tests which check auxiliary index registers, store addressing to the store with a complementary address name of the base store, and store address interaction.

When the fault recognition program is ready to release the system from interrupt control, measures are taken to determine if the return to normal system operation can be to the interrupted point (the point at which the mismatch occurred) or must be made to an arbitrary reference point. If the return is to the interrupted point, the retrial program will attempt to unwind the interrupted instruction. If the retrial program cannot unwind this instruction, it transfers to the fault recognition program, which forces a return to an arbitrary starting point (reference return).

5.1.5 *Store Faults*

Normally, a store reread or write failure creates an E-level interrupt and calls the store fault recognition program. The basic steps the program follows are:

(i) Find the faulty store, bus, or processor. Isolate the failing unit from the rest of the store-processor complex, and request an appropriate diagnostic program.

(ii) Update status records.

(iii) Reconfigure the remaining system in order to create a system configuration with maximum store duplication for call processing.

(iv) Return the system to normal processing.

Program transfers to out-of-range addresses or memory reads or writes into out-of-range addresses will also create an E-level interrupt which initiates the store fault recognition program control program. The basic steps the program follows in this case are:

(i) Recognize that the error data saved by the processors on all store errors indicates an out-of-range condition.

(ii) Recognize that the communications address register contained an unequipped store address at interrupt time.

(iii) Request a selected set of high level audits in an attempt to find and correct the error in temporary memory.

(iv) Request a return to a reference point in the executive control program.

(v) Request the maintenance restart program to turn on special timing because of possible data mutilation in the system. If this event occurs repeatedly in the next 16-second interval, this will automatically trigger a call processing recovery phase (or phases). The latter is discussed in Section 5.3.3.

(vi) Establish the store and processor configuration existing prior to the interrupt.

Store fault recognition program programs are, in general, organized so that an overall control program, uniquely associated with some particular broad functional task, links together a number of routines and tests. These routines and tests range in scope from simple test routines to comprehensive testing and analysis routines. Also, many contingencies can arise which do not cause E-level or G-level interrupts and, hence, do not directly engage the store fault recognition program programs. Nevertheless, they do require the testing of the store-processor complex. For the aforementioned reasons, it was desirable to organize the store fault recognition program to create a pool of routines and tests useful to the maintenance and system programs which must handle such contingencies.

The primary purpose of the store fault recognition program is to establish in a minimum amount of time a workable set of stores that provide

access to at least one copy of all data. Determination of the faulty store is of secondary importance. In order to accomplish this objective, the store fault recognition program keeps a record (or maps) of the present set of in-service, and off-line but usable stores. After determining the failing unit and updating the maps, the store fault recognition program uses a generalized routine which uses the maps to determine and establish a maximum store configuration. Once this configuration has been established, this program executes a quick access check to each store unit. If this passes, it is assumed that the faulty unit has been successfully isolated. Then, on a deferred basis, the faulty unit is diagnosed.

If a configuration cannot be established because history indicates mate* stores are in trouble, or if the access test fails, this program starts a bootstrapping operation in which a bus is chosen and all stores on that bus are tested using test pattern techniques. The set of stores not passing on this bus are then tested on the other bus. If a complete configuration is obtained, the system is restarted as outlined above, and, on a deferred basis, the remaining faulty and/or untested stores are tested. Those that pass the tests are returned to system operation; those that fail are diagnosed. If the basic bootstrapping operation fails, an attempt is made to establish a working set of stores using the single error correction features; that is, previously failing stores are tested to see if they have only a single-bit oriented fault. If successful, a working system is recovered. Otherwise, manual intervention is required.

A program controllable (on/off) feature is available in the interrupt sequencer to insure that, on an E-level interrupt, the active bus after the interrupt sequence was not the bus causing the E-level interrupt (i.e., conditional bus switching on interrupt). This feature is activated by the store fault recognition program, if the store containing the E-level program is in service on both buses. It is useful in minimizing recovery time. When it cannot be used, the emergency action timers provide a back-up mechansim.

5.1.6 Peripheral Unit Faults

Peripheral unit faults are detected by the same checks used to detect peripheral unit errors. Some of these checks are central pulse distributor all-seems-well, enable verify, and peripheral unit all-seems-well. An error is distinguished from a fault by the success of the initial retry. All work performed on faults detected by an all-seems-well failure or

---

* Both copies of the same unit.

an execute reply failure is performed in F-level. The failing order is retried using a different central pulse distributor from the original order. If these retries fail, a retry is made utilizing the off-line processor. Control of the central pulse distributor and peripheral unit buses is given to the standby machine and a retry is executed. When any retry succeeds, the faulty unit is removed from service and control is returned to call processing. The procedure described for central pulse distributors is also followed for the application peripheral unit address translator. However, in the case of this unit, control is transferred to an application subprogram to complete the final cleanup work.

The SPC system contains only one unit, which returns a scanner all-seems-well; viz., the master scanner. Fault recognition for the master scanner progresses in the following manner. The order is retried using the mate master scanner controller. If the retry is successful, the original scanner controller is marked in trouble and all enables for the unit are updated. If the orders to the mate fail, an attempt is made to determine if the scanner row presently being addressed is faulty; this is done by reading other rows in the same master scanner. If other rows can be successfully read, the all-seems-well check is deleted from all enables for this scanner, and control is returned to normal processing. If neither of the above strategies has been successful, the order is executed through the standby processor in an off-line mode. The active processor will be removed from service if this procedure is successful.

When an enable verify failure occurs and the unit involved is of the fast* type, all possible routes will be tried in an effort to achieve a working configuration. Here, again, enables will be updated, the faulty equipment will be removed from service, and control will be returned to the normal processing programs.

Fault recognition for the slow units is also a logical extension of error recovery. When the initial retry procedure has failed, a series of retries using other routes is attempted in order to achieve an operational configuration. The retrial procedure is implemented in J-level on a time-shared basis to provide proper order timing. All routes are attempted to insure recovery. The first successful retry causes termination of the procedure and a request of the appropriate diagnostics. Although the details of the retry strategy may vary depending on the unit involved, the following is representative of the procedure followed. If central pulse distributor and bus status indicate a good route exists to the mate controller, the failing controller is removed from service and the order is

---

* A fast unit can receive and process orders at electronic speeds, whereas a slow unit requires about 25 milliseconds to process an order.

retried using the mate. When this fails to achieve success, all possible routes are tried in an arbitrary fashion. There are four possible routes to each of the slow units. The routes are determined by using either central pulse distributor with either peripheral unit bus. If none of these retries succeed, the order is executed from the standby processor in an off-line mode. If this does not succeed, no further action is taken.

The SPC peripheral fault recognition programs contain a great number of control tables to allow easy adaptation to application demands. An application system may easily provide special routines and programs for handling non-SPC equipment. In this way, the SPC peripheral system has the flexibility necessary to adapt to a wide variety of application needs.

5.1.7 *Processor Diagnostics and Exercises*

The primary objective of the processor diagnostic program is to isolate a fault in the standby processor to a small number of replaceable circuit packages. This objective is met by performing a series of rigorous tests on the standby processor, recording the pass-fail results of each of these tests, processing the resultant pass-fail test data by a dictionary number generating program into a fixed-format trouble number, and transmitting the trouble number to the maintenance craftsman via the maintenance teletypewriter. The maintenance ·craftsman refers to a trouble locating manual to associate the trouble number with the corresponding circuit package(s). Replacement of the circuit package(s) associated with the trouble number is expected in most instances to remove the fault from the standby processor.

5.1.7.1 *Matching Features.* The primary means of trouble detection for the processor is the matchers. They do, however, have a secondary function of providing a means for diagnosis of the processors; i.e., they provide access to many points within the processors.

Basically, the match system must be able to perform both a directed match and a sampled match. In a directed match, the matchers are directed to look at match sources* at specified time segments on a continuous basis.† In the sampled match, the matchers are directed to look at specified sources at a specified time segment, a given number

---

* A match source is comprised of a set of internal points within a processor; e.g., an internal bus, decoder points, sequencer status, etc.
† A variation because of machine complexity is for the matcher(s) to be directed to look at a set of match sources on a sequential basis. The source to be matched may also become a function of the instruction(s) being processed.

of machine cycles from the point of initialization. This is frequently called selective, discrete, or snapshot matching. This is a powerful tool and it is used in certain "diagnostic only" tests in the integrated program method to be discussed in the following sections. However, the integrated tests use the directed match mode with the flexibility provided by varying the match sources and/or the segments of the machine cycle when matching occurs. In this mode, the specified sources are matched continuously on every machine cycle at one or more of the three time segments.

In the processor diagnostic program, it is assumed with a high level of confidence that the standby is faulty and that the active is fault-free, or that the active is faulty in circuitry not effecting the normal system operation in the system's present configuration. Thus, the active can be used to test the standby. The assumptions of single failure and no redundant logic are applied when designing tests. The execution of a test follows a strict pattern:

(*i*) initialize circuits external to the circuit being tested so their interaction with the tested circuit will be consistent,

(*ii*) apply inputs to the circuit under test,

(*iii*) deactivate or reinitialize the tested circuit when necessary,

(*iv*) compare outputs with expected outputs, or the active processor results, and

(*v*) record results indicating whether the test passed or failed. If it failed, record results about how it failed.

5.1.7.2 *Integrated Approach.* In the integrated approach, a single functional block (or phase) of tests performs both diagnostic and fault recognition functions. The majority of the tests are "integrated" tests. The basic difference between using the program for fault recognition or diagnostics is the collection of data. When this program is run as a fault recognition program, data from the standby processor is not collected. When the program is run as a diagnostic, a failure in test *i* will provide data which "points" to the *i*th executed instruction and indicates the bit pattern of mismatch. An interrupt technique invokes data recording for diagnosis (Step 5 in the preceding outline of the program structure) so that this function can be omitted when the tests are performed for fault recognition. A general flowchart for the integrated program for a test is shown on Figure 15. Note that the test now includes all instructions including initialization. Thus, this is a continuous sampling technique.
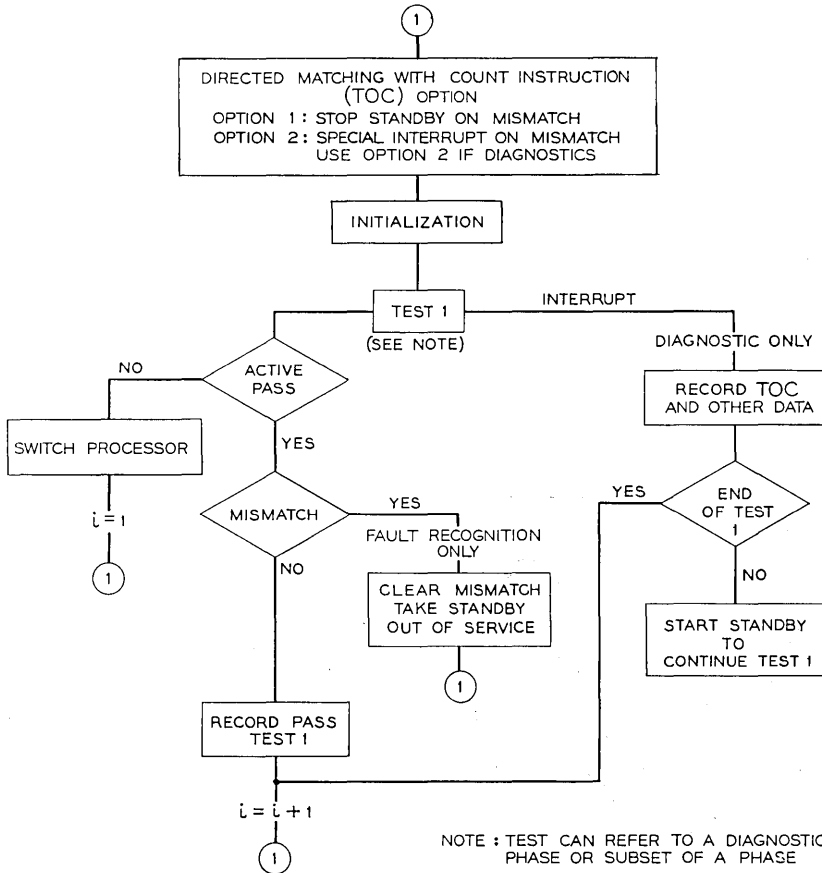
Fig. 15—Integrated technique.

5.1.7.3 *Processor Diagnostic Control Program.* The processor diagnostic program may be entered to run a complete diagnosis as a result of a fault recognition program detecting a processor fault, for a demand exercise to perform one or more phases (or subphases), or for an automatic exercise of one or more phases (or subphases). The processor diagnostic control program must be able to determine the nature of the request and base control decisions on the particular function being performed.

The processor diagnostic control program performs general initialization, phase and subphase initialization, linkage between phases and subphases, and termination functions.

A diagnostic program is broken into logical testing entities called subphases or phases. A phase is the series of tests performed between entries to the dictionary number generation program which consolidates this information. This dictionary number generation program preprocesses a maximum of 40 raw data (pass-fail or bit pattern of mismatch) words at a time. A subphase is used to describe a functional block of tests and comprises a series of tests between entries to the diagnostic control program. Several subphases comprise a larger functional entity which is used to form a phase and which satisfies the maximum raw data words constraint.

Processor diagnostic program general initialization includes the marking of control bits to determine which diagnostic phases should be run, the disjoining of processors, and similar details. In addition to some common initialization required by all phases and subphases, the control program uses a control table to determine special initialization necessary for the subphase to be run. The control table concept is illustrated in Fig. 16.

The control program uses a pair of ordered-bit words for linkage between phases (phases 1 through 20 are represented in a 20-bit word and phases 21 through 36 in the other). A "one" is marked in the bit position representing each phase to be run. The processor diagnostic program linkage of phases and subphases is performed by the control program which scans the two control words for the next "rightmost one". The bit position of the "one" indicates the phase number to be run. A head table is indexed by the phase number and contains the address of the first word of a control table related to the phase to be run. Subphases within a phase are sequentially placed in the control table depicted in Fig. 16, with the last subphase of a phase containing an "end-of-phase bit." This bit signals the completion of a phase and indicates a need to request the dictionary number generating program to process the data obtained. This process continues until all requested phases have been processed. Then, the dictionary number generating program is entered to produce a trouble number.

5.1.7.4 *Tests and Test Phases.* For testing purposes, the processor circuits are organized into functional blocks that are tested as independently as possible. A hierarchy of tests was established based on the critical nature of the particular circuits and processor functions that are being checked. The hierarchy of tests is the basis of the selective phase skipping and early termination procedures which will be described in a succeeding section.

CONTROL CONTROL
SEQUENCE TABLE
INDEX

		SUBPHASE ADDRESS	M	L	K	J	I	H	G2	G1	F	E	D	C	B	A
— —	— — — — — —															
— —	— — — — — —															
— —	— — — — — —															
— —	— — — — — —															
— —	— — — — — —															
— —	— — — — — —															

CONTROL PROGRAM ACCESS TO CONTROL TABLE

1. THE CONTROL PROGRAM MAINTAINS A POINTER (CONTROL TABLE INDEX) INDICATING THE CONTROL TABLE LOCATION BEING ACCESSED.

2. THE CONTROL SEQUENCE IS ALWAYS EXECUTED IN ASCENDING ORDER OF CONTROL TABLE ADDRESSES.

METHODS EMPLOYED BY PDIAG USING CONTROL TABLE CONTENTS

1. USE BYTE A–D AS ORDERED BITS BUFFER:
   (a) LOCATE RIGHTMOST "1".
   (b) EXECUTE HARDWARE OPERATION.
   (c) LOCATE NEXT RIGHTMOST "1".
   (d) EXECUTE HARDWARE OPERATION.
   (e) ETC.

2. USE BYTE E AS AN INDEX TO A TRANSFER VECTOR TABLE TO PERFORM FIXED SUBROUTINES: AN EXAMPLE IS MATCHING INITIALIZATION.

3. USE BIT L TO DESIGNATE THE FINAL SUBPHASE WITHIN A PHASE.

4. USE THE SUBPHASE ADDRESS AS THE POINT OF ENTRY INTO THE SUBPHASE PROGRAM TO BE EXECUTED.

Fig. 16—Control table arrangement.

Control program access to control table:
(*i*) The control program maintains a pointer (control table index) indicating the control table location being accessed.
(*ii*) The control sequence is always executed in ascending order of control table addresses.

Methods employed by processor diagnostic program using control table contents:
(*i*) Use byte A–D as ordered bits buffer:
    (*a*) Locate rightmost "1."
    (*b*) Execute hardware operation.
    (*c*) Locate next rightmost "1."
    (*d*) Execute hardware operation.
    (*e*) And so on.
(*ii*) Use byte E as an index to a transfer vector table to perform fixed subroutines: an example is matching initialization.
(*iii*) Use bit L to designate the final subphase within a phase.
(*iv*) Use the subphase address as the point of entry into the subphase program to be executed.

An individual test might consist of any one of, or a combination of, the following actions: (*i*) examining various state indicators for specific conditions, (*ii*) completely exercising circuits to verify the ability to assume all possible states, or (*iii*) applying inputs to circuits to effect expected actions. Groups of tests form subphases. A subphase may test one or several circuit areas.

5.1.7.5 *Recording Results of Processor Diagnosis.*   There are essentially two methods of evaluating test results. In one method, the active and standby processor test results are compared. In the other method, the test results are compared with expected values. It should be noted that various logical functions are used to compare and record results within the framework of the two aforementioned methods of test evaluation.

An interrupt mechansim is used by the processor to record diagnostic test failures in a general purpose scratch area (see Fig. 15). The interrupt does not occur until the completion of the instruction causing the mismatch, so that the active processor will contain the complete data necessary to continue at the next executable instruction. This interrupt causes the diagnostic recording program to be entered. After recording the data the diagnostic program is re-entered at the next executable statement. This process continues until the end of the subphase.

Periodically, a check of accumulated test results is performed in order to isolate certain active processor faults. An active processor test failure causes the diagnostic to be aborted. A diagnostic message with data pertinent to the active processor failure is printed on the teletypewriter. This message is also useful in isolating faulty circuits being used as test tools by the processor diagnostic program.

Recording of data for all phases uses a standardized record area. Figure 17 shows the recording for a typical phase. Normally, a match source point is associated with each test area. The first four words are used for storing the mismatch bit pattern. Word five is used to record the point of execution (cycle and phase) *within* the instruction sequence at which the failure was detected for the first three failures. Each independent test is recorded "pass" or "fail" in words six through forty as needed for a given phase.

5.1.7.6 *Selective Test Skipping.*   The running of phases is subject to a selective test skipping mechanism which is controlled by a table. Each phase has a control table entry which is applied in one of two ways: (*i*) the entry specifies a set of phases to be skipped if this phase fails; or (*ii*) the entry specifies a set of phases to be skipped if there have
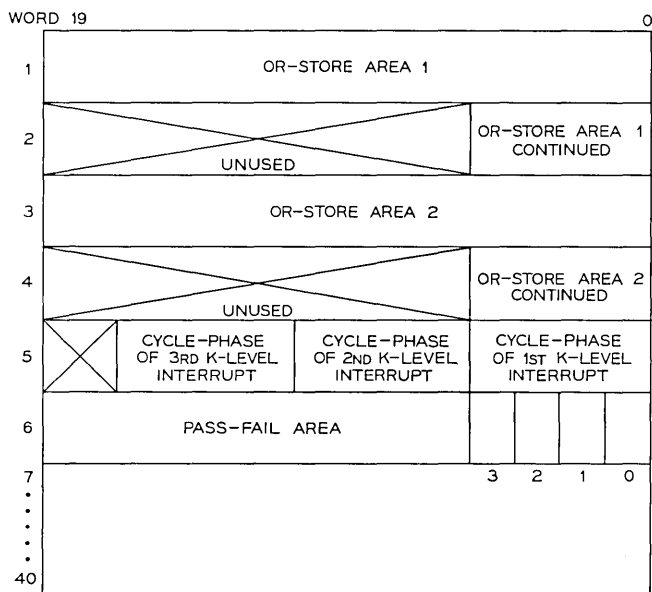
WORD 19   0

1	OR–STORE AREA 1
2	UNUSED / OR–STORE AREA 1 CONTINUED
3	OR–STORE AREA 2
4	UNUSED / OR–STORE AREA 2 CONTINUED
5	CYCLE–PHASE OF 3RD K–LEVEL INTERRUPT / CYCLE–PHASE OF 2ND K–LEVEL INTERRUPT / CYCLE–PHASE OF 1ST K–LEVEL INTERRUPT
6	PASS–FAIL AREA
7	3 2 1 0
40	

Fig. 17—Typical phase recording.

been any failures recorded by the diagnostic up to and including the present phase. The skipping rules were derived logically based on the axiom that diagnosis should continue only if verified test circuits are available to test the next circuit. For example, if phase seven, matcher access tests, fails, then phase eight, matcher function testing, is skipped. The skipping table entries were later modified considerably as a result of sample fault analysis.*

An additional abort rule compares the accumulated number of distinct tests which have failed over a number of related phases with a threshold. If an unacceptable number of failing tests have been recorded, the diagnostic is terminated. This rule is used to minimize inconsistent results by ending the diagnosis if many basic tests fail, thereby preventing the running of more sophisticated tests which could lead to confusing results. A second purpose of this abort rule is to avoid giving the standby processor control of a store bus and/or peripheral bus if a basic fault has already been uncovered. This prevents adverse interaction with normal data processing.

---

* Thousands of faults were physically inserted into the processor(s) to evaluate the diagnostic program.

Finally, individual tests are skipped or the diagnostic is terminated within a phase when certain critical tests within a phase fail.

5.1.7.7 *Some Results.*   The techniques discussed previously produced a considerable reduction in the number of faults which resulted in inconsistencies when diagnosed repeatedly. Previous comparable diagnostics have resulted in approximately one of twelve faults producing differing results during repeated fault simulation. In contrast, only one out of sixty faults fell in this category during fault simulation with the SPC processor diagnostic program. The improvement in consistency can be attributed to differences in hardware design (which uses a "bit-oriented" hardware layout), recording techniques, abort rules, and program structures.

5.1.8 *Store Diagnostic Program and Exercise Program*

The store diagnostic program attempts to locate a fault within a store which has previously been found faulty. The tests are performed using both processors when they are in service. Most of the testing is performed using special maintenance orders provided for this purpose. Using these orders, only the store on the bus specified by the order responds. The route (i.e., bus 0 or bus 1) for these maintenance orders can be specified by a control flip-flop within the processors. Since stores are dedicated to a particular bus, selectivity is achieved. In addition, a pair of special maintenance orders were designed with associated special store circuitry to test the bus leads associated with the store complexes. These two orders provide complementary mode control bits to the store, directing it to transfer its input directly to its output. Thus, a minimal amount of store control is required to test the store interface with the bus system.

The store tests are divided into related groups called "phases" which test various portions of the store circuits. A special store monitor bus is used for observing internal dc points; a store "snapshot" register is available to sample internal functions in a manner similar to a processor match circuit when it is being used in a sampling match mode. Control of this "snapshot" register is provided by special maintenance instructions called control reads and control writes.

If the all-tests-pass diagnostic of the store occurred after a faulty circuit pack was replaced, certain sections of memory may have become outdated (i.e., they do not contain recent changes in variable data) before the power was restored to the frame. Therefore, before a store is returned to the working system after an all-tests-pass diagnostic

result, steps must be taken to ensure that the store memory contents are current. The contents of the diagnosed store should be the same, or be made the same, as the contents of the mate store on the other store bus.

If the all-tests-pass diagnostic occurred as a result of a scheduled, automatic exercise, no rewriting should be necessary. The store diagnostic program keeps stores updated while diagnostic tests are being run.

The checking and updating of a store which has passed diagnosis is performed in two millisecond segments, with call-processing interrupts inhibited. Each segment, whether checking or writing, deals with eight 40-bit words. The checking includes two parts: comparing 40 bits of each word with the corresponding word in the mate store, and a check of a processor buffer bus flip-flop bit. The flip-flop can be set by a parity failure, a double bit error, an address error, or a read or write all seems well failure.

If one or more 40-bit mismatches occur during a checking segment, the entire eight-word block will be rewritten from the mate in the following segment. The eight-word block is then rechecked in the next two millisecond segment to see that the writing was performed properly. As stated previously, the buffer bus flip-flop is checked only if 40-bit mismatches have not occurred in the current checking segment. If the flip-flop indicates an error, the eight-word block will be rewritten and rechecked in the following two segments.

Stores are automatically diagnosed and updated once a day to insure that maintenance circuits are excercised and that residual single errors in memory are not allowed to accumulate.

5.1.9 *Peripheral Unit System Diagnostic and Exercise Programs*

The SPC System provides diagnostic programs and exercise programs for all its major peripheral units. The exercise routines keep records on manually induced equipment states and initiate routine diagnostic testing of their respective frames. The diagnostic programs provide testing in response to fault recognition requests and manual teletypewriter requests. The five SPC peripheral units with diagnostics are: the master signal distributor, the master scanner, the central pulse distributors, teletypewriters, and the program tape unit.

The teletypewriter and program tape unit diagnostics do not conform to the scheme described above. The teletypewriter diagnostic can be requested manually and is also run as a scheduled exercise. This is because there is not a fault recognition program for the teletypewriter.

The program tape unit diagnostic is only run in response to a manual request; no fault recognition or routine exercise is available. Since both of these frames are under manual supervision, this scheme does not present any problems.

Although the basic functions of the diagnostics and exercises are those mentioned above, certain special features do exist. These factors provide the flexibility necessary to allow the system to be used for many applications. The scanner and signal distributor diagnostics are specially designed to allow use of many of their program segments in diagnostics provided for similar application peripherals. An application system with frames similar to the scanner or signal distributor can eliminate a great deal of additional program by utilizing these features. The diagnostics use tables and transfers to special application sub-programs to achieve this flexibility.

The central pulse distributor diagnostic also provides transfers to application sub-programs to provide tests of central pulse distributor features that are dependent on frame assignments. This means that individual applications are not restricted to any special scheme in assigning central pulse distributor points for their equipment.

5.1.10 *Off-Line Testing*

The off-line control programs provide testing tools to assist the maintenance operator. They are generally used to locate the cause of hardware problems which cannot be readily detected by the diagnostic programs (due to inconsistent results, intermittent faults, etc.).

Off-line testing is limited to processors, stores, and store buses. No special provision is made for off-line testing of peripheral equipment since test facilities are available at the master control center and independent use of the peripheral buses would pose major problems.

5.1.10.1 *Off-Line Functional Tests.* Off-line functional tests fall into two basic categories: those which are repetitively executed, and those which are performed in a single operation. Repetitive tests include:

(*i*) Execute a specified instruction or group of instructions at a fixed repetitive rate.

(*ii*) Write a specified 40-bit word repetitively into one or more consecutive unprotected store locations.

(*iii*) Write a specified 20-bit word repetitively into one or more consecutive unprotected store locations.

(*iv*) Read 40-bit word(s) repetitively from one or more store locations.

(*v*) Read 20-bit word(s) repetitively from one or more store locations.

One-shot tests include:

(*i*) One-shot initialization of a processor buffer bus register.

(*ii*) One-shot initialization of an unprotected store location.

5.1.10.2 *Non-Off-Line Functional Tests.* Non-off-line functional tests provide a means of printing data on the maintenance teletypewriter concerning the state of the active system upon execution of any specified program instruction. This data is useful in analyzing hardware troubles or verifying program changes.

To obtain the data printout, the maintenance operator sets the switches on a plug-in auxiliary matcher unit (which will generate a G-level interrupt on a match condition) to the octal address of a desired program instruction. He then activates this monitor function by typing in an appropriate input message.

Activation of the plug-in unit will not affect normal system operation or matching. The first time the program instruction specified on the plug-in unit is executed, a G-level interrupt takes place. The G-level program then enters a dump routine which produces a teletypewriter printout of all active processor buffer bus registers. In addition, any 12 memory locations specified in the input teletypewriter message will be dumped.

The plug-in unit is deactivated and no further dumps will be made unless requested by the maintenance operator. System operation will continue with no effective change of state.

5.1.10.3 *Method of Performing Off-Line Repetitive Exercise.* Repetitive exercises give the maintenance operator the ability to execute any specific program instruction(s) available in the off-line store(s) or specified via the teletypewriter while observing the results on an oscilloscope or making other operational checks. The instruction(s) are executed at a repetitive rate of approximately once every 50 milliseconds. An oscilloscope may be synchronized on the start pulse of the standby processor or any signal directly related to the start pulse.

Briefly, the sequence of performing repetitive exercises is accomplished in the following manner. The input teletypewriter message either specifies the address of the first instruction to be exercised or the program internally generates this start address. The same conditions apply for the last instruction to be executed (end address). If only one instruction is to be executed repetitively, the start address and end address are identical

The end address plus four (pseudo end address) is inserted into the match register physically located in the active processor. The matchers are set up in the directed mode in a manner which causes the standby processor to stop after completion of the instruction specified as the original end address.

When entered by the J-level interrupt control program, the off-line program starts a time-out counter in the active processor and inserts the stored starting address into the standby processor program address register. The match circuitry is activated to stop the standby processor on a match of the pseudo end address. The standby processor is then started. While the standby processor is executing the instruction(s), the active processor continuously monitors the "standby-processor-stopped" flip-flop.

As soon as the standby processor stops, the active processor switches buses so that it can write into both sets of stores in order that the standby store system can be kept up-to-date during call processing. It then returns to the main program for normal call processing. At the next scheduled entry from the J-level control program (nominally 50-millisecond intervals), the entire procedure is repeated.

In order to defend against the standby processor's inability to reach the pseudo end address and stop, the active processor counts the number of cycles elapsed since the program was entered by the J-level control program. Each time the standby processor does not stop within a fixed period of time (approximately one millisecond), a time-out counter is incremented. The standby is forced to stop, and the normal off-line routine described above is repeated at the next J-level entry. The time-out counter is interrogated at each J-level interrupt entry into the off-line control program. If the number of time-outs exceeds a programmed limit of 100, the off-line-control program informs the maintenance operator of the time-outs, clears all off-line indicators, and restores the system to normal. A request is made to diagnose and update the off-line store frame, and the standby processor is left out of service.

5.1.10.4 *Limiting Write Access to a Single Off-Line Store Frame.* The off-line processor is restricted to sending and receiving on a specified off-line bus. There is also a further restriction on write operations by the off-line processor. Write instructions are completely inhibited by the processor unless specifically requested by the maintenance operator. When requesting an off-line system configuration, the operator must request write access to a specified off-line store frame.

The store name code is placed into the off-line processor down-store name match register. The off-line write access flip-flop is then set. Normal down-store name matching in the active processor is not affected. However, if the off-line processor attempts to write into a store, the store name code of the store being accessed is automatically compared with the store name code in the down-store name register. If the store name codes are not identical, an E-level interrupt is generated and the off-line testing is terminated. This defensive action is necessary to protect against the mutilation of processing information.

The above restrictions have no effect on normal processing, since the active processor sends on both buses during all time intervals between actual off-line program activity.

5.2 *Input-Output Control Programs*

The SPC system provides a number of basic input-output control functions for the use of application systems. There are four major input-output control functions provided by the SPC system. These are: program tape unit read and write control; teletypewriter message translation and control; control and display panel input control; and POB execution for buffered control of various units. The design of these programs embodies the flexibility needed to provide service for a wide variety of application devices. This flexibility is achieved through the use of structured tables and provision for application sub-programs in appropriate areas.

The teletypewriter control program provides control and structuring of all system output messages. It provides the necessary features for a wide variety of output message formats. It also provides a variety of input message translations and numerous mechanisms for passing control and data to programs responsible for reacting to the various input messages. The program is designed with provisions for variable size application message translation tables in addition to the SPC translation tables. This feature allows an SPC application system to rely on teletypewriter control program for all teletypewriter control and message construction.

The program tape unit control program provides three major functions, viz., bootstrap recovery, system loading, and system copying for SPC and application systems. A bootstrap program is provided to reload the system programs when operation has been curtailed by severe mutilation of protected memory. There are two copies of this program on each store bus to insure availability for system recovery. This enables the bootstrap to function after almost any system failure. The

bootstrap program can be initiated from the control and display panel when the system is not operational by inserting a special card into the processor to cause transfer of control to the bootstrap on a manually induced A-level interrupt.

When the bootstrap program receives control, it determines a workable combination of master scanner, central pulse distributor and peripheral address bus for communication between the processor and the tape unit. It then begins the system reload by unlocking all frames on both store buses and writing the data obtained from tape into duplicated memory. The store unload logic is written in such a way as to minimize the total size of the program. Also, the program must be self-sufficient so that it does not depend on any parameter data to obtain store or program tape unit enables. In this way, it can reload any SPC system regardless of the degree of mutilation in other programs. To some degree this requires a blind approach to the memory writing operation. For example, all store frames are unlocked before each 40-bit word is written. However, since the program must wait for the slow-speed tape unit (5 inches/second), this inefficiency is acceptable and permits an office independent design.

The program tape unit control program provides two major functions during normal system operation: system reloading for installation of generic program changes; and system copying to provide permanent tape records of protected memory. The program to provide these facilities operates in J-level and base level. The J-level entry provides accurate timing for communication with the program tape unit. The base level program assembles data for the J-level program. In this way, a minimum of J-level occupancy is required.

The craftsman can use the program tape unit program in a variety of ways. Copies can be made of any desired sections of memory. Also, there are automatic features to provide dumping or loading of all office data areas and all program areas. These areas are defined by application tables and may be easily redefined for any application.

The A-level interrupt control program provides filtering of emergency requests made by the craftsman using the control and display. When a system reconfiguration request is initiated from the control and display, the control program insures that proper software status is established for the new configuration. The program also responds to manual requests for call processing recovery phases and passes control to the system initialization and recovery programs. Additionally, there are several minor functions provided. The primary one is reinitialization of the teletype software to allow the craftsman to gain control of the

teletypewriter when it is not functioning properly. In addition, for the use of application systems, several spare keys are also provided. Modification of a table used by the A-level control program enables these keys.

The last input-output control program residing in the SPC generic system is the peripheral-order-buffer execution program. For the SPC, this program provides queuing of data destined for the signal distributor. In application systems, it can provide similar services for any peripheral unit with a low data rate. In addition, the program also provides specialized services for fast units. In this way, specified order sequences can be executed by the program and rigidly defined timing between events will be automatically provided by the input-output program. For the basic SPC system, orders for the scanner and the central pulse distributor may be initiated by the peripheral-order execution program in any combination with signal distributor orders and delay orders.

The SPC implementation of the peripheral-order execution program provides a great deal of flexibility for adapting to application needs. All order sequences in the buffer consist of a control code followed by one or more data items. Each control code defines a unique subroutine designed to provide a particular service. These subroutines are accessed through a transfer vector table thus allowing any application system to tailor the program to its needs by merely providing a special subroutine for each new device to be serviced.

5.3 Maintenance Administration

5.3.1 Maintenance Control

The base-level executive control program enters the maintenance control program through the lowest priority base-level class (Class E). At this time, maintenance control searches for requested work according to a priority hierarchy. Deferred fault recognition work is higher than diagnostics, which in turn are higher than exercises. In honoring new requests, maintenance control insures that the highest priority request is performed first. Within the broad priority categories, the search for work requests is made by hardware unit type. These unit types are arranged in the status tables such that maintenance for the more critical units is performed first.

When maintenance control finds a job request, the requested client program is started and the job is carried out on a segmented basis. Segmenting of the maintenance program is necessary to prevent inter-

ference with the processing of normal system tasks. Each client segment is approximately ten milliseconds long after which maintenance control returns control to the base level executive control. When maintenance control is reentered in Class E, it initiates the client's next segment.

Maintenance control administers, coordinates, and schedules maintenance client programs through the maintenance control register, a common temporary scratch pad and control block. The scratch pad area is used by the client to record test data. Since the scratch pad is used by almost every client, maintenance control performs housekeeping functions to avoid conflicts. Maintenance control also provides periodic status reports on the teletypewriter regarding all units in the system.

The maintenance alarm scan is a subprogram of maintenance control that is entered in Class C of the base level every two seconds. It provides periodic scheduling for maintenance control clients such as audits and automatic exercises. It also provides general purpose timing for maintenance control as well as the maintenance control clients and performs administrative functions such as controlling lamps on the equipment frames.

5.3.2 *Maintenance Interrupt Return*

The Maintenance Restart and Restore Program provides a common return mechanism for maintenance interrupts. It runs any necessary clean-up jobs and determines where the system should be restarted.

The restart program provides several entry points for programs returning from maintenance interrupts. Each entry provides special functions appropriate to the returns it receives and then enters a common clean-up thread. The clean-up thread is a general purpose task dispenser. It executes a number of subprograms that check the integrity of data structures after an interrupt. Programs that cannot be reentered after maintenance interrupts provide abort routines to be run by the clean-up task dispenser. For example, the peripheral order buffer execution program provides a subprogram to insure that it will not be reentered if a maintenance interrupt occurs while it has control. Similarly, all programs administered by the maintenance control program are aborted when a maintenance interrupt occurs. Finally, one of the subprograms tests the hardware and software interrupt counts to determine if they are excessive.

The final action performed is program restart. The point of return is established, the registers are restored if appropriate, and control is returned to a lower level. Selection of the point and level to which the return is made involves several factors. Certain maintenance interrupts

or combinations of maintenance interrupts must never return to the interrupted point. Instead, a reference point return is made. Reference points are provided in the executive control program, and they will restart processing in a natural fashion. When a reference point return is not required, execution will be reinitiated at the point and level where the current interrupt took place. In some situations this may also be a maintenance interrupt level. One other factor is considered in determining the restart point. If a software recovery phase has been requested or was in progress at the time of the interrupt, control will be passed to the beginning of the software recovery control program in the emergency action program. A new software recovery phase will then be initiated. After the return point has been established, the registers are restored if a direct return is to be made, and control is passed to the appropriate program.

The restart program provided with the SPC is easily modified for use by any application program. The subprograms mentioned above can be augmented to provide data structure validation and interrupt sensitivity testing for all programs unique to a given application.

5.3.3 *Emergency Action*

The emergency action program provides monitoring of system normalcy and control of recovery from abnormal operation. When the system is incapable of normal operation due to a major hardware failure, the B-level portion of the emergency action program is entered. This program operates in conjunction with the hardware emergency action sequencer to provide an operational hardware configuration. The hardware provides a series of initial processor and store bus configurations, and the program tests these configurations. The processor is tested by the processor sanity test program under control of the emergency action program. All processor functions except peripheral communications are tested. A failure in one of these tests activates the hardware sequencer which will then select a new configuration. When a working processor has been found, the central pulse distributors are tested. A series of tests are run to verify the basic addressing and functional mechanisms of the central pulse distributor. For each central pulse distributor pair, the best central pulse distributor is selected as the active unit. The final action in hardware emergency action recovery is selection of an operational store system. This is accomplished through the store bootstrap program. This program makes a series of read and write tests to select an acceptable store system. If a complete system is not available, the hardware emergency action sequencer is started to select another configuration.

The second major function of the emergency action program is to insure the sanity of the software system. This is done by a number of tests which originate in a high-priority J-level program which is entered every 100 milliseconds. One of the functions of this program is to administer a hardware long timer. Failure to perform these actions will cause an emergency action (B-level) interrupt. In addition, the program makes several tests for software insanity of a less obvious nature, and it initiates system overload actions. Five basic tests are performed. The systems' interject response is monitored to detect loops in base level programs. The time between Class E base level visits is monitored to determine if the system is in an overload state. J-level activity is monitored to insure that no J-level program is in a loop. The base level job visitation schedule is monitored to guard against various forms of base-level processing insanity.

Hardware and software caused maintenance interrupts are also monitored. Software-caused maintenance interrupts are considered to be a very strong indication of mutilated data or residual program errors. Hardware interrupts are not considered to be a conclusive indication that the system has generated and used invalid data. Therefore, a small number of software-caused interrupts are considered to indicate a serious loss of normal processing, whereas the count for hardware interrupts must be relatively high to reach the same conclusion. For all of the above tests, certain failure thresholds are set. If these thresholds are exceeded, a call processing recovery phase will be initiated.

The emergency action recovery program provides control of call processing recovery phases. The control program initializes hardware and makes requests for all appropriate software routines to audit and correct the variable areas of memory. It also monitors the sanity of the programs operating during the phase. If a phase fails to recover the system, the control program will restart the action and run the audit routines appropriate to the next higher phase. The recovery control program is table-driven and has the flexibility necessary for use in any application system. With appropriate audit routines, a complete call processing recovery phase system can easily be developed for any application.

VI. SUMMARY

The SPC No. 1A has been designed as an economical and reliable central processor which can be easily applied to varied real-time control and logical processing functions. This article has attempted to give an

overview of the SPC hardware and software design while placing emphasis on the more novel features of the system. Thus, particular attention has been given to certain aspects of the order structure, the development of standard interfaces for the application systems, duplication and switching arrangements, and special program structures. It has been assumed that the reader has some familiarity with the stored program switching systems of the Bell Telephone System such as the ESS No. 1 system.[1,14] Thus, areas of similarity which have already been well documented in connection with these systems were described only briefly herein for completeness.

The common system design of the SPC proved to be of great value in the adaptation of the second application system, the electronic translator (ETS). A number of installations of both the TSPS and ETS systems are now in service and the SPC has proven to be dependable and easy to maintain.

REFERENCES

1. Keister, W., Ketchledge, R. W., and Vaughan, H. E., "No. 1 ESS: System Organization and Objectives," B.S.T.J., *43*, No. 5, Part 1 (September 1964), pp. 1831–1845.
2. Jaeger, R. J. Jr., and Joel, A. E. Jr., "TSPS No. 1: System Organization and Objectives," B.S.T.J., this issue, pp. 2417–2443.
3. Baker, W. A., Kinder, G. W., Myers, F. H., and Culp, C. A., "TSPS No. 1: Stored Program Control No. 1A Store," B.S.T.J., this issue, pp. 2509–2560.
4. Freimanis, L., Guercio, A. M., and May, H. F., "No. 1 ESS Scanner, Signal Distributor and Central Pulse Distributor," B.S.T.J., *43*, No. 5, Part 2 (September 1964), pp. 2205–2283.
5. Comella, W. K., Day, C. M., Jr., and Hackett, J. A., "TSPS No. 1: Peripheral Circuits," B.S.T.J., this issue, pp. 2561–2623.
6. Baldinger, W. R., and Clark, G. T., "TSPS No. 1: Physical Design," B.S.T.J., this issue, pp. 2685–2709.
7. Harr, J. A., Hoover, Mrs. E. S., and Smith, R. B., "Organization of the No. 1 ESS Stored Program," B.S.T.J., *43*, No. 5, Part 1 (September 1964), pp. 1923–1960.
8. Cagle, W. B., Menne, R. S., Skinner, R. S., Staehler, R. E., and Underwood, M. D., "No. 1 ESS Logic Circuits and Their Application to the Design of the Central Control," B.S.T.J., *43*, No. 5, Part 1 (September 1964), pp. 2055–2097.
9. Harr, J. A., Taylor, F. F., and Ulrich, W., "Organization of No. 1 ESS Central Processor," B.S.T.J., *43*, No. 5, Part 1 (September 1964), pp. 1845–1922.
10. "No. 1 ESS," B.S.T.J., *43*, No. 5 (September 1964), pp. 1831–2609.
11. Dougherty, H. J., Raag, H., Ridinger, P. G., and Stockert, A. A., "No. 1 ESS Master Control Center," B.S.T.J., *43*, No. 5, Part 2 (September 1964), pp. 2283–2319.
12. Downing, R. W., Nowak, J. S., and Tuomenoksa, L. S., "No. 1 ESS Maintenance Plan," B.S.T.J., *43*, No. 5, Part 1 (September 1964), pp. 1961–2020.
13. Beuscher, H. J., Fessler, G. E., Huffman, D. W., Kennedy, P. J., and Nussbaum, E., "Administration and Maintenance Plan, "B.S.T.J., *48*, No. 8 (October 1969), pp. 2765–2816.
14. Quinn, T. M., Toy, W. N., and Yates, J. E., "Control Unit System," B.S.T.J., *48*, No. 8 (October 1969), pp. 2619–2668.

TSPS No. 1:

Stored Program Control No. 1A Store

By W. A. BAKER, G. A. CULP, G. W. KINDER and
F. H. MYERS

The single memory system provided in the Stored Program Control (SPC) No. 1A is built around the piggyback twistor (PBT). Each SPC 1A store provides 770,048 bits of random-access nondestructive readout (NDRO), electronically alterable information organized into 16,384 words. The store has a read-cycle time of 6.3 μs.

The paper describes in detail the memory medium, memory circuits, and equipment design. A general description is given of store operations and store diagnostic and maintenance procedures.

I. INTRODUCTION

Digital storage in the Stored Program Control (SPC) 1A is provided by the piggyback twistor (PBT) memory store. The heart of this system is the PBT module which is composed of pairs of wires wrapped with two magnetic tapes, one piggyback on top of the other. The SPC 1A store has been designed around the PBT memory module. This store meets the SPC 1A requirements of high reliability and low cost with an operating speed that is adequate for the present SPC 1A applications. High reliability is provided by the nondestructive readout (NDRO) characteristic of the PBT memory. The memory element needs no regeneration after reading and is not disturbed by power interruptions. It is alterable at electronic speeds with this feature being protected from accidental use by several independent checks on the write process. Low cost has been achieved by high-volume continuous fabrication of the PBT memory element.

The SPC 1A store[1] is used for all the memory needs in the SPC 1A. This is in contrast to No. 1 Electronic Switching System (No. 1 ESS) in which the memory needs were split into two groups, temporary memory for the storage of call-related data and semipermanent

memory for storing programs and translation data. The temporary memory, known as the call store[2] uses the ferrite sheet as the basic memory medium. The semipermanent memory is called the program store[3] and uses the permanent magnet twistor (PMT).

Figure 1 shows a photograph of a SPC 1A store frame and lists some of its important characteristics. The 47-bit word consists of 40
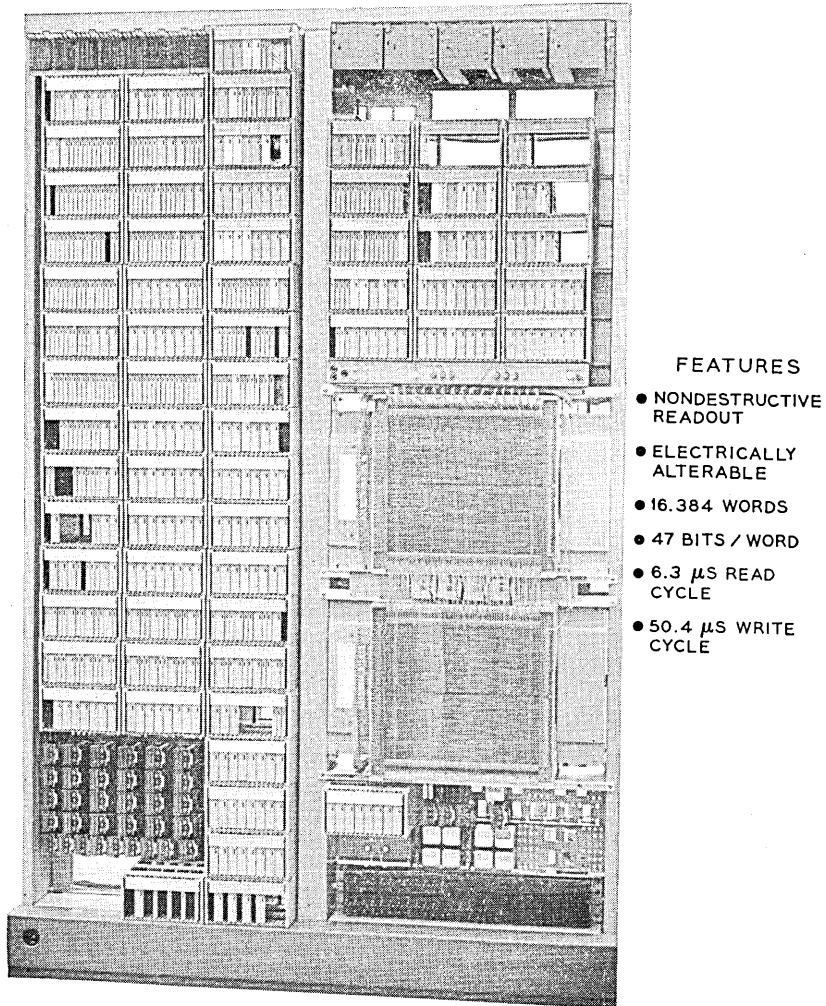

FEATURES

● NONDESTRUCTIVE READOUT

● ELECTRICALLY ALTERABLE

● 16.384 WORDS

● 47 BITS / WORD

● 6.3 μS READ CYCLE

● 50.4 μS WRITE CYCLE

Fig. 1—Photograph of SPC 1A Store and Table of important characteristics.

bits of information, six Hamming check bits, and one parity bit. This provides single- and double-error detection over the word and its address as well as single-error correction on the information. The capacity of 16,384 words allows SPC No. 1A storage to be provided economically for applications that require moderate amounts of program and data.

The store read-access time is under 3.25 μs with a 6.3 μs read-cycle time. The write time for a single store is 50.4 μs, but multiple stores in a single system permit the use of an overlap write mode in which one store can begin a read operation before another store has completed a write. With this overlap feature, a write instruction requires between 18.9 μs and 56.7 μs.

In addition to the PBT memory modules, the store contains all the circuitry required to operate the memory. Access circuits select the proper memory word and apply word-line drive currents for reading and writing. Bit drivers provide information-dependent bit current for writing. Sense circuits amplify the memory output and make the one or zero determination. Circuits are included to provide a communication link with the processor for normal operations and maintenance.

Maintenance is an important consideration in the store design. Both software and hardware facilities have been provided for detecting and isolating store faults.

II. MEMORY

2.1 *General*

Of the few known approaches to a memory device that offered the features of both electronic writability and nondestructive readout, only an adaptation of the permanent magnet twistor memory—called the Piggyback Twistor (PBT) memory[4]—offered the economics that were consistent with large-capacity stores for electronic switching systems.

2.2 *Organization*

2.2.1 *Capacity*

The 15B PBT memory module has a capacity of 4096 words and 47 bits per word for a total of 192,512 bits. An SPC 1A store uses four of these modules to obtain a total of 16,384 words. Since the

PBT module is an integrated approach to magnetic memory design; that is, 4096 memory sites on a continuous pair of wires, its manufacturing yield is very sensitive to isolated defects. One defective memory site renders the entire 4096 sites on a bit line unusable. Manufacturing yields close to 100 percent are maintained for the PBT by assembling the module with additional capacity so that seven spare bit lines are available to replace any bit line having one or more sites that fail to pass the final electrical tests.

2.2.2 Physical Arrangement

Figure 2 is a photograph showing a 15B PBT memory module as a background for a few unfolded memory planes. The inserts are enlargements of sections of the planes. The entire memory consists of 64 of these memory planes. Each plane contains 64 words at 0.175" spacings and a plane-to-plane spacing of 0.20 inch making the 64 × 64 array of words nearly square. The bits are spaced on 0.05" centers along the word line. The active memory bit density is 570 bits per cubic inch. The overall module dimensions are 15" × 15.25" × 5.75" and the unit weighs about 27 pounds. A toroidal ferrite core per word is included within the memory structure to perform part of the address decoding function (insert of Fig. 2).

Each bit line consists of a pair of piggyback twistor wires which are fabricated by wrapping two thin, flat tapes of magnetic material spirally on a 40-gauge copper wire. Figure 3 shows a photograph of a short section of twistor wire taken on a scanning electron microscope at 500X magnification. The second magnetic tape, which is wrapped underneath the first tape, can just barely be seen. The double wrap suggested the name piggyback twistor.

Each wire of the bit line is continuous throughout the memory. The 4096 bits per line require only four connections, one on each end of the pair of wires. The actual length is about 76 feet. The 108 wires that make up the 54 bit lines (47 active and 7 spare) in the PBT module are encapsulated in plastic before module assembly. In this form, the bit lines can be handled easily and economically. The word lines, which are mounted in groups of 64, are assembled as single turn solenoids complete with their access and shuttle suppression cores. Sixty-four word line subassemblies or "planes" are bonded to the bit cable. Then the bit line cable is folded so that the planes lie on top of one another. The resulting stack of 64 planes is mounted in a lightweight metal frame.
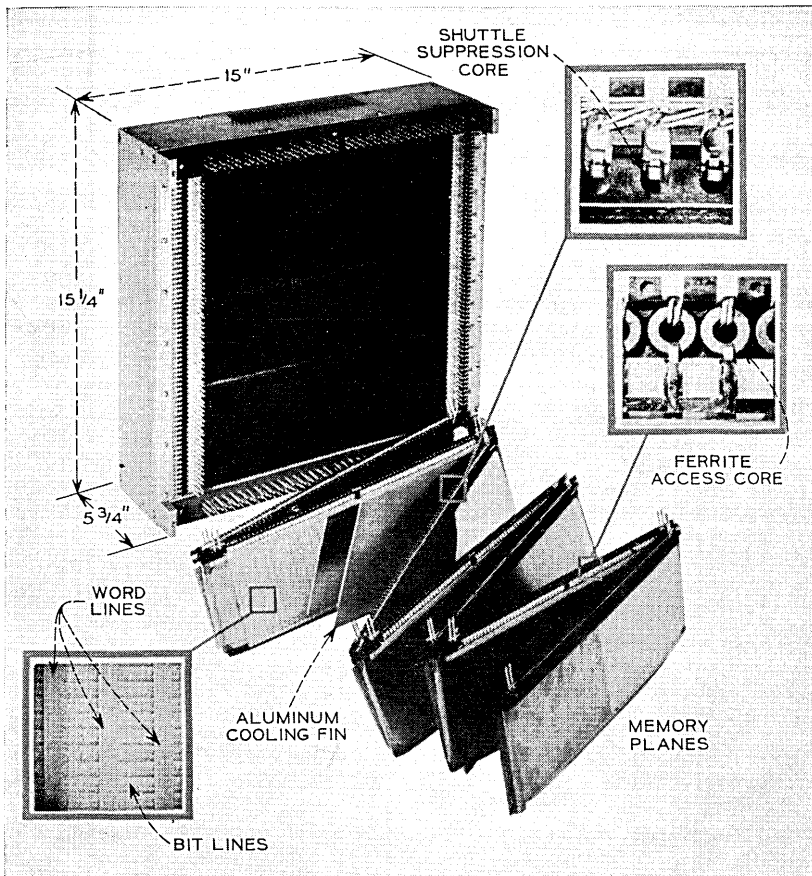
Fig. 2—Photograph of a 15B Memory module in the background and an un-folded section of a module in the foreground showing the 64-word plane assembly and the continuous bit line cable. Inserts are expanded views of the twistor wire, access core, and shuttle suppression core.

2.2.3 *Electrical Configuration*

The PBT memory, like any other memory, requires drive currents to read and write information, and contains sense lines along which it transmits the information to the external readout circuits. Figure 4 shows a block diagram arrangement of the PBT.

The access drive currents are generated external to the memory. They supply the excitation to an access core matrix which is part of
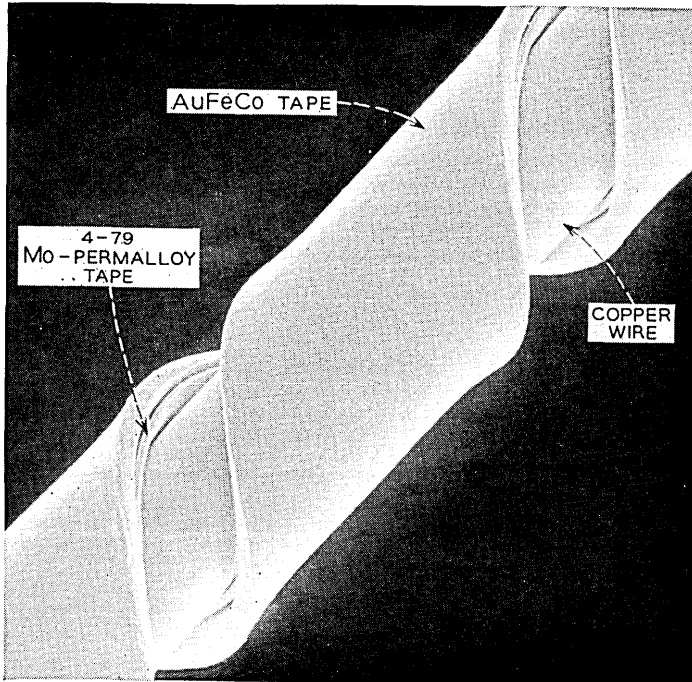
Fig. 3—A photomicrograph of a piggyback twistor wire. The memory, tape (AuFeCo) is on top wrapped around a #40 AWG copper core wire. The edge of the sense tape (4-79 Mo-Permalloy) can just barely be seen under the memory tape.

the memory and which converts the access drive into a current on a single memory word line. Between the access core matrix and the word lines is a second array of cores that compensates for the fact that the access cores are not perfect threshold devices. If the access cores were perfect, a single X-access and a single Y-access current would be transformed into a single word-line current. Actually, in addition to the desired word-line current, 126 other word lines have small currents induced in them. The shuttle suppression core array keeps these unwanted currents small enough to be tolerable.

The 4096 word lines shown in Fig. 4 intersect the 47 active bit lines. A current flowing in a single word line will drive all bit lines in parallel. The bit lines are shown to consist of pairs of twistor wires, terminated at one end by a nonreflective termination and 47 bit driver circuits; and at the other end by a diode array and 47 sense channels.

For writing information into a memory word, an X and Y-access

line is energized, thereby selecting a single word line in which useful
current flows. At the same time, each bit driver generates current
around the loop formed by the two wires of the bit line and the
terminating diodes. The information written into each bit of the
selected word is a function of the associated bit current polarity. The
coincidence of the 47 bit currents and a single word line current writes
the 47 bit word.

In order to read a PBT word, the desired word is again selected by
a single X and Y-access current. This time, however, no bit current
flows. The word current switches magnetic material on one wire of
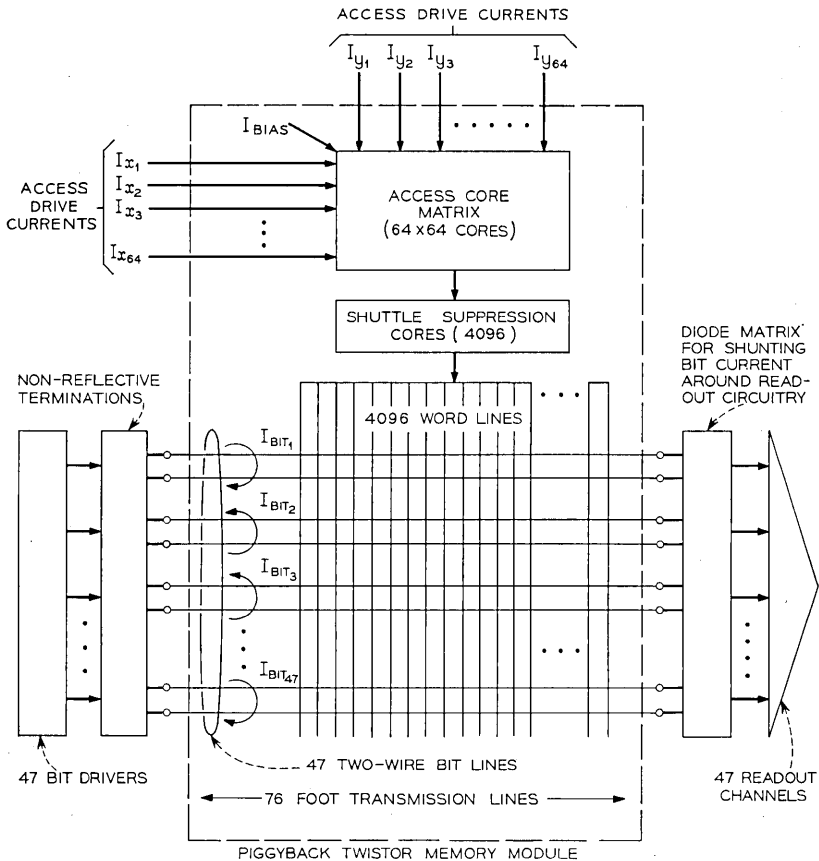
Fig. 4—Block diagram of a PBT memory showing connections to the external
drive and readout circuits.

each of the 47 bit line pairs thus inducing a voltage on the bit line which now acts as a sense line. This information—either a positive or negative voltage—is transmitted down the sense/bit line to the external readout circuits.

The details of how each of the major components of the PBT contributes to the overall performance of the memory is given in the following sections.

2.3 *Theory of Operation*

2.3.1 *Two Material System*

The memory element of the PBT is made up of two short sections of twistor wire, a sample of which is shown in Fig. 3. The two tapes are mechanically wrapped on the copper wire, one to perform the information-storage function and one to sense the information. Both are made from square loop magnetic materials; that is, their characteristic hysteresis loops have vertical sides and very little slope on the top and bottom. Such loops are typical of magnetic memory materials, but the hysteresis loop of a PBT wire is not typical in two important aspects. First, it is the composite of the loops of two magnetic materials having different values of coercivity and magnetization. This produces steps in the hysteresis loop. The second difference is that there is a significant demagnetizing field caused by the fact that only a short length of wire is magnetized in operation. This demagnetizing field has the effect of shearing the otherwise vertical sides of the hysteresis loop. Figure 5 is an idealized sketch of the hysteresis loop of a PBT wire.

Of the two tapes on the PBT wires, the outside tape stores the information and the inside tape senses it. The information in the storage tape is determined by the direction of magnetization. The magnetization produces a magnetic field external to the tape that is related to the demagnetizing field. This external field can be approximated by:

$$H_{\text{ext}} = \pm \frac{2\varphi_r}{\pi \text{l}^2} : . \tag{1}$$

The sign of the external field (H_{ext}) is determined by the direction of magnetization of the remanent flux (φ_r) and the amplitude is directly proportional to the flux level and inversely proportional to the square of the length (l) of the magnetized section.
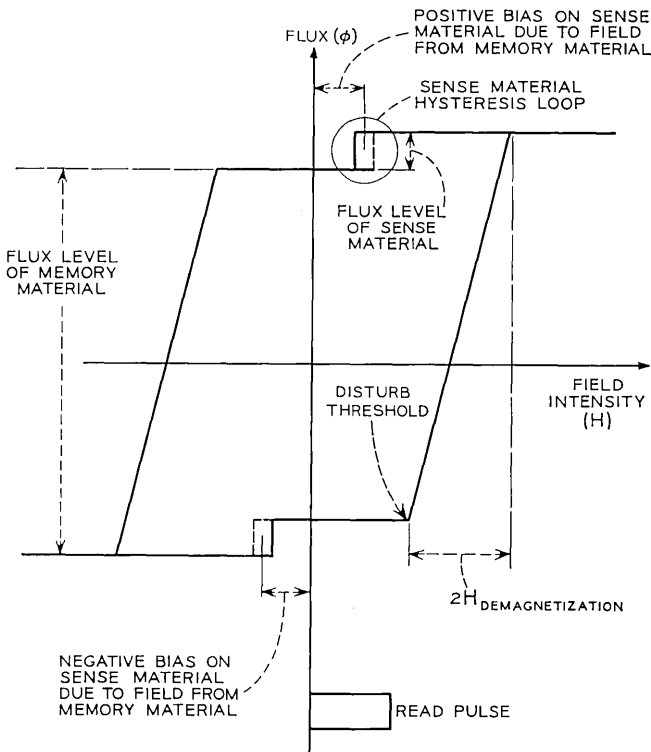
Fig. 5—The composite hysteresis loop of a PBT memory element.

2.3.2 Writing a PBT Bit

In order to write information into the PBT, a short length of twistor wire must be definable as a bit location, and the magnetization of the memory material along that short length must be driven to positive or negative saturation.

The bit location is defined by the intersection between a word line and a wire pair that makes up a bit line. The magnetization of the memory material is driven to saturation in this region by the coincidence of a word-line current and bit-line current. (Since the magnetic material is wrapped at about a 45° angle on the copper core wire as shown in Fig. 2, it is coupled by both bit- and word-line currents even though these currents are orthogonal to each other.) Figure 6 shows
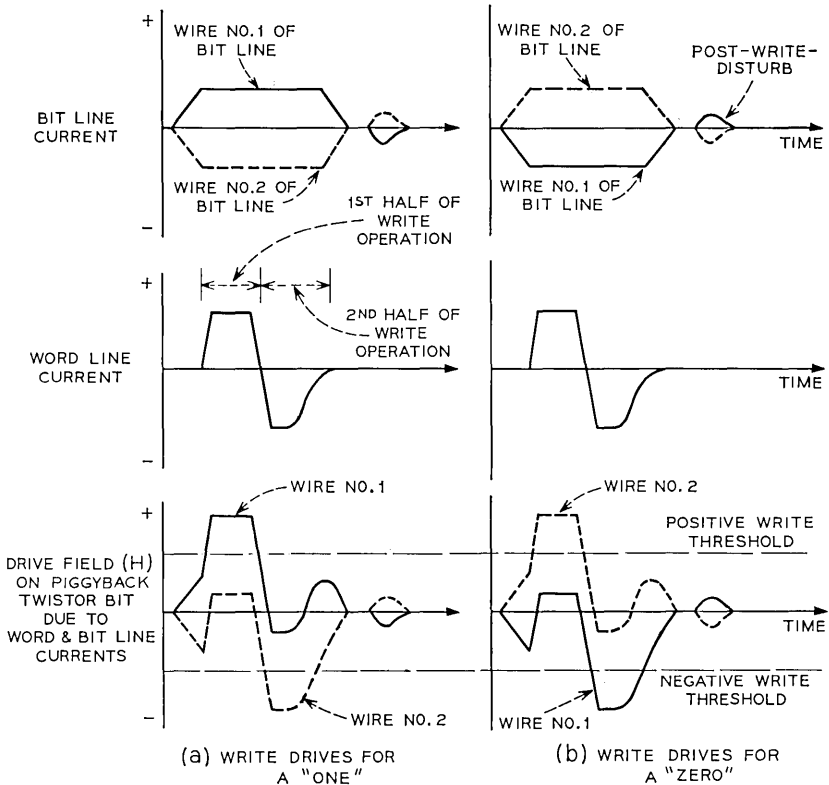
Fig. 6—Write fields acting on each wire of the bit line for writing: (a) a ONE, and (b) a ZERO.

the currents and resulting fields for writing a one or a zero. The bit current flows in opposite directions in each wire of the pair. The word-line current for writing flows in one direction for half of the write time and the other direction for the other half. For the first half of the write operation, the fields due to the bit and word current will add at one wire and subtract at the other. During the second half of the operation, the two fields will add at the other wire of the pair but with the opposite polarity than the previous summation. The result is that the magnetization of the memory material will be driven to saturation on both wires of the pair, but one will be driven positively and the other negatively. For writing information of the opposite polarity, only the bit current polarity is changed, but this

interchanges the effects on the two wires of the bit pair, reversing the magnetization in each.

For proper writing, the currents involved must be held within certain limits. The derivation of these limits will be discussed in Section 2.5. Another requirement is that the timing of the currents must be such that the bit current is maintained at its proper level during the application of the word current. When the bit current is removed, the bit line voltage is reversed momentarily resulting in a post-write-disturb (PWD) current that completes the writing operation. The purpose of this pulse is to counteract the disturbing effects of the bit current.

To insure that the bit length is the same for both polarities of magnetization (i.e., for both ones and zeros), the memory material between adjacent bits must be close to zero magnetization. If it were not, it would tend to make a bit location, of the same magnetization direction as the interbit material, appear longer than a bit location of the opposite magnetization. This would result in an asymmetrical external field [equation (1)] and an asymmetrical composite loop (Fig. 5). To avoid problems due to lack of symmetry, all bit lines are demagnetized during memory manufacture. During memory use, no drive fields are applied to the bit lines with amplitudes sufficient to disturb the zero magnetization level between bit locations. Only the individual bit locations are driven to saturation.

2.3.3 *Reading a PBT Bit*

The reading of stored information depends on the sense material magnetization state being a function of the memory material state. This is accomplished by the action of the external field [equation (1)]. The amplitude of this field is sufficient to bias the sense material beyond its switching threshold. The polarity of this field, which is determined by the magnetization direction of the memory material, either permits or prevents sense material switching as a result of a word-line current. For example, in Fig. 5, for a read pulse as shown, the sense tape would be switched if it were under the influence of a positive bias, and it would not switch under the influence of a negative bias. The read pulses must not exceed the disturb threshold or else the memory material flux will be switched, thus altering the stored information.

For the two-wire bit, information is written so that there is one and only one wire biased to switch. The information—whether a one or

a zero—is determined by which of the two wires is biased to switch and, consequently, whether a positive or negative readout voltage is induced on the bit line.

Reading, then, is accomplished by impressing a current on a word line that had been used in the past to write the desired information into the 47 associated bits. This current results in switching sense material on one of the wires of the bit pair and 47 voltages—some negative, some positive, depending on the stored information—are induced on the bit lines and transmitted to the input of the sense circuits. After the read current is removed from the word line, the external field from the memory material resets the sense material making the read operation a nondestructive one. An unlimited number of read operations are possible on any memory word without any intervening write operations.

2.3.4 Access Core

The access core switch is similar in operation to the access cores used in permanent magnet twistor memories (PMT). It performs part of the memory address decoding function by virtue of its threshold characteristic. Each word line in the memory is coupled to an individual access core. The cores are arranged in a 64 × 64 array. Each column of 64 cores has a two-turn winding threading the cores, and each row of 64 cores has another set of two-turn windings. A single-turn bias winding threads all 4096 cores in the array. The bias winding is energized with dc to bias all the cores to saturation. In order to select a word line for reading or writing, the row and column windings, which thread the core associated with the proper word line, are energized with a current pulse. The combination of these pulses is enough to overcome the bias on that particular core, switching it and inducing a current on the word line. The other cores on the selected row and column are not switched because they receive only a single pulse, which is not sufficient to overcome the core bias. This results in a linear select mode of operation for the PBT. Using a ferrite core as the access device, instead of the more commonly used semiconductor diode, decreases the number of access switches from that required for diode access, provides a current step-up to the word line by transformer action, and also provides a bipolar word-line current by the resetting action, thereby eliminating the need for a bipolar current driver.

There are two major differences in the use of the access core in the

PBT compared to its use in the PMT. The first is that the core is used to write information into the PBT. This means that significantly higher currents must be induced in the word line and, because of the two-wire bit, these currents are bipolar as shown in Fig. 6. The second difference is that the read output is sensitive to the current induced in the word line when the access core is being reset by its bias. Unless this current is limited properly, the timing of the output signal can change, the read-cycle time can be affected, and the NDRO feature can be destroyed.

In order to perform the write operation, the access core must contain sufficient magnetic flux to induce currents in the range of 3.8 amperes in the word line for a duration of about 6 microseconds. As long as the X- and Y-access pulses are of sufficient amplitude and duration, the write pulse will be bipolar—the bias current resetting the core will induce the opposite polarity word current. The duration of the reset current is determined by the amount of access core flux that had been switched by the coincidence of the X and Y pulses. If these pulses are too short, the core flux available on reset may be insufficient to write the second wire of the bit pair. For optimum writing the bipolar currents on the word line should be symmetrical.

During the read operation, the opposite problem occurs. The read is a linear select operation; i.e., all 47 bits of a word are read out in parallel by a single current on the word line. This current is induced by switching the access core in the same way that the write current was induced. In this case, however, the appropriate word line current is in the range of 1.7 amperes. Too large a current during the read operation will approximate a write-access current. This will not write information because of the absence of bit current but it will destroy stored information. This effect can be prevented on the initial read pulses by limiting the amplitude of the X and Y pulses. In order to limit the reset current, however, the amount of core flux switched must be limited. This is accomplished by a maximum limit on the duration of the X and Y read pulse.

2.3.5 *Transmission Line Effects*

Since each bit line has a capacity of 4096 bits, its physical length is relatively long—approximately 76 feet. This length and the frequency spectrum of the output signals from the bit locations make the transmission line characteristics of the bit line important memory parameters. Of particular interest are the velocity of propagation, the

attentuation, and the phase shift. The effect of these characteristics on the memory output can be seen in Fig. 7. This figure shows an oscillogram of two output-voltage wave forms from a PBT memory—one from an address adjacent to the readout terminals (near end) and the other from an address 4095 bits away from the readout terminals (far end). The oscillogram shows the far-end signal on the right to be lower in amplitude, longer in duration, and delayed when compared with the near-end signal. The variability of the output signal due to transmission line characteristics increases the difficulty of signal discrimination. However, because the transmission delay is a function of address, discrimination circuits can be designed to compensate for the variable delay of the memory output.

2.4 Design Details

2.4.1 Magnetic Materials

The choice of magnetic materials for the two twistor tapes that are wrapped on the twistor wire is based on a compromise among design parameters. The sense tape requires a relatively low switching threshold so that it can be biased easily by the memory tape and also so that it can be switched with relatively low drive currents. The total flux of the sense tape determines the amplitude and duration of the output signal and should therefore be as large as possible. However, total flux is restricted by the limitation on tape cross-sectional area necessary to
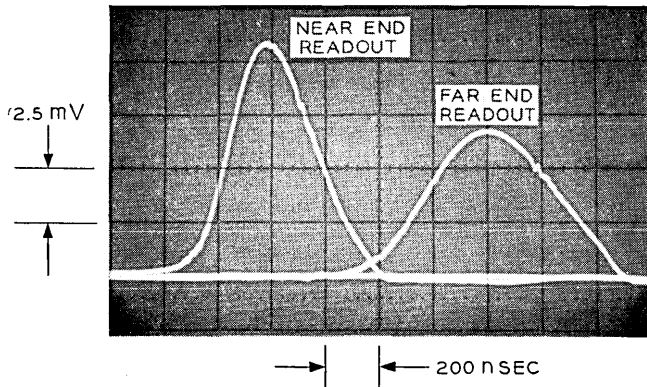
Fig. 7—Output waveforms showing the effect of bit position on the bit line. The earlier, higher amplitude signal is at the near end and the later, lower amplitude signal is at the far end with reference to the read amplifier.

minimize eddy currents, which tend to impede flux switching. In addition, the sense tape flux must be considerably less than the memory tape total flux so that a sufficient bias can be developed by the memory tape. The coercive forces of the two materials should be in the ratio of about 15 to 1 with the memory tape being higher so that it will not be disturbed by read currents. Minimum read cycle time and acceptable drive current variations also require trade-offs in material characteristics. The switching time produced by the read current is expressed by:

$$\tau_{read} = \frac{S_W}{H_{read} - (H_0 + H_{ext})}$$

where:

τ_{read} = time required to switch 90 percent of the tape flux
S_W = switching coefficient of the tape
H_{read} = read current field
H_0 = switching threshold of the tape
H_{ext} = external field from memory tape [see equation (1)].

The time required for the external field from the memory tape to reset the sense tape is:

$$\tau_{reset} = \frac{S_W}{(H_{ext} + H_{read}) - H_0}. \tag{3}$$

Because H_{read} is a function of time, its polarity and amplitude in equation (3) differ from those in equation (2). Since the total memory read cycle is related to the sum of τ_{read} and τ_{reset}, there is an advantage to minimizing this sum. This requires a compromise value for H_{ext}. Equation (1) shows that the value of H_{ext} is determined by the memory tape properties. A limitation on H_{ext} that is not expressed in equation (1) comes from the fact that H_{ext} affects the memory material as a demagnetizing field; that is, it tends to reduce its magnetization to zero. In order to assure a stable state of magnetization in the memory material and thereby insure against a loss of information, the switching threshold of the memory tape (H_{0m}) must be considerably larger than H_{ext}. That is:

$$H_{0m} > H_{ext}. \tag{4}$$

The value of H_{0m} cannot be increased without increasing the write current amplitudes.

The relationships that determine the field intensities required for proper writing are:

$$H_{\text{word}} + H_{\text{digit}} \geqq H_{0m} + H_{\text{ext}} \tag{5}$$

$$H_{\text{word}} - H_{\text{digit}} \leqq H_{0m} - H_{\text{ext}} \tag{6}$$

$$H_{\text{bit}} \leqq H_{0m} - H_{\text{ext}} \tag{7}$$

where

H_{word} = field intensity of the word line write current

H_{bit} = field intensity of the bit current.

All these inequalities must be satisfied concurrently in order to have proper memory writing. In order to minimize write currents on both the word line and bit line, H_{0m} must be minimized.

A compromise among the requirements expressed in equations (1) through (7) coupled with a desire to maximize the amount of signal induced by switching the sense tape, led to the selection of material properties. A 4-79 molydenum permalloy was selected for the sense material. It is processed to give a coercive force of 0.75 oersted and a total flux of approximately 1 mV-μs. The memory material is a 4Au-12Fe-84Co alloy[5] with a 13 oersted coercive force and a total flux of about 5 mV-μs. Besides the fact that these two materials can be processed to produce compatible properties, a second outstanding characteristic of these materials is the lack of strain sensitivity. Thus, inexpensive assembly techniques can be used in memory production.

2.4.2 Ferrite Cores

Each word line in the PBT memory threads two ferrite cores—an access core and a shuttle suppression core. (See the inserts of Fig. 2.) The access core uses a square-loop material so that it acts as an address decoder. The core has low loss so that it is an efficient current transformer, and has sufficient flux to sustain the high write currents for a sufficient duration. The core that was selected was a Cadmium ferrite with a nominal cross section of 0.023 cm^2 and a mean diameter of 0.314 cm. This produces a loss of 0.3 ampere-turn and a total flux greater than 900 mV-μs.

By itself, the access core is acceptable for driving the selected word address—its flux is sufficient and its loss is low enough. However, it is not an adequate decoder because it produces a shuttle current when it is one of the 63 cores on a selected row or column that receives a single pulse. Outputs induced by 126 of these currents can obscure the output from the selected core. To remedy this situation, the additional shuttle suppression core is used. This core adds a large induc-

tance in series with the word line impedance, thereby reducing the word line current induced by a half-selected core. (See Fig. 8a.) In order that the fully-selected word line current is not reduced by the inductance of the shuttle suppression core, this core is designed to saturate whenever the word line current approaches a full select current. (See Fig. 8b.) The added nonlinear inductance minimizes noise currents without seriously affecting drive currents. To do this, the core is made from a linear ferrite with a total flux of 100 mV-μs and an ID of 0.038 inch.

2.4.3 *Plastics and Adhesives*

The low assembly cost of the PBT memory is due in large measure to the generous use of plastics and adhesives. The 108 twistor wires
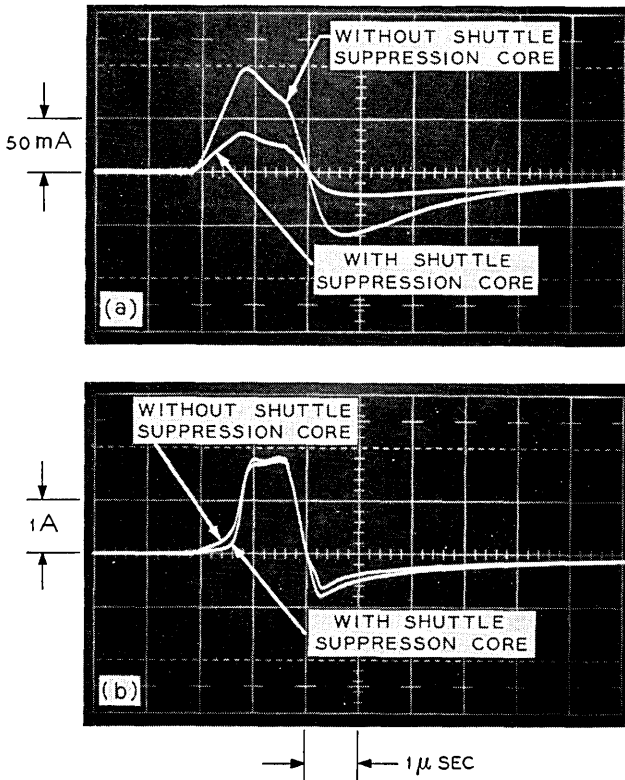
Fig. 8—Word line current waveforms showing: (a) Current on a half-selected word line with and without a shuttle suppression core. (b) Current on a fully-selected word line with and without a shuttle suppression core.

which form the bit line cable are economically assembled into the memory by first encapsulating them in a polyester sandwich. Each side of the sandwich is made up of a 1-1/2 mil polyester film coated on one side with 2-1/2 mils of polyester adhesive. The adhesive sides of the two films are brought together with the wires between them and bonded under heat.

The materials and process for this assembly had to be designed to produce a structure that has long-term stability under operating conditions that could include short-term temperatures as high as 180°F within the memory. The use of preshrunk film and high temperature adhesive, rigid control of encapsulating temperatures, and film tensions offer the solution. The success of this assembly technique is greatly enhanced by the use of strain insensitive magnetic tapes on the twistor wire.

A similar encapsulating process is used to assemble word lines in groups of 32. A single film is used in this case. One side of the flat copper word lines is embedded in the adhesive. Long lengths of word line cable are produced and cut into short lengths to make the word line loop. Two of these 32 word line subassemblies are required per memory plane.

A specially developed pressure sensitive adhesive is used to bond the plastic encapsulated bit line cable to the word line subassemblies. Experience showed that using only a pressure-sensitive adhesive for this bond would not result in a uniform and stable spacing between the word line and bit line and this dimension critically affects the transmission line properties of the bit line. In order to stabilize this dimension, a 1/2 mil plastic film coated on both sides with pressure-sensitive adhesive is used. The film adequately stabilizes the spacing between the word lines and bit lines. A cross section of the word line and bit line cable assembly is shown in Fig. 9.
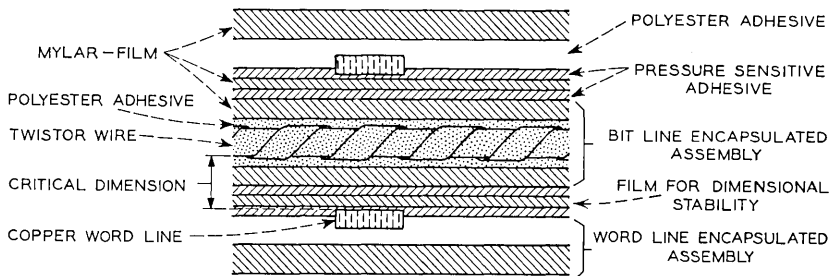

Fig. 9—Cross section of a bit line and word line assembly.

2.5 *Operating Characteristics*

2.5.1 *Drive Variations*

Since selection of the magnetic material properties determines the operating currents, any practical design must assume operation over some variation in drive currents. The PBT is designed for access drive current variations (*X*-axis, *Y*-axis and bias) of ±2 percent. These currents determine word line current according to the equation:

$$I_{word} = 2(I_x + I_y) - I_{bias} - I_{loss} \tag{8}$$

where I_{loss} is a loss term due to the energy required to switch the access core. A study of equation (8) will show that the 2 percent variation in access currents amounts to a considerably larger variation in word line current. Table I summarizes the drive variations throughout which the PBT memory cell must operate.

2.5.2 *Output Signal*

The output signal from the PBT is bipolar. The polarity is related to the polarity of bit current that was used to write the information. The output amplitude ranges from a minimum peak of 3.5 mV to a maximum of over 10 mV depending on the drive levels, information stored, operating history, etc. The width of the output signal at the 0.5 mV level can have a duration of about 900 ns but this duration is decreased and its timing is varied by the operating conditions. Figure 10 shows the effects of some of the operating variables on the output signals. Figure 10a is a photograph of 64 ONE outputs from a single

TABLE I—DRIVE VARIATIONS

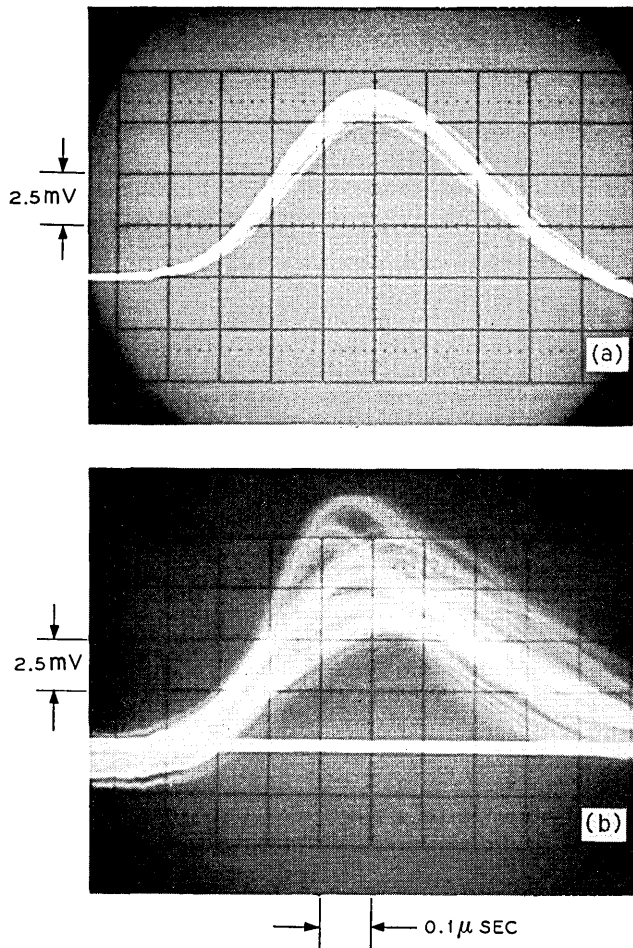|	Maximum (amps)	Minimum (amps)	Variation (%)
Bias I_{bias}	3.98	3.82	2.0
Read Access I_x I_y I_{word}	1.48 1.48 1.8	1.42 1.42 1.4	2.0 2.0 12.5
Write Access I_x I_y I_{word} I_{digit}	2.04 2.04 4.04 0.15	1.96 1.96 3.56 0.21	2.0 2.0 7.7 16.6

Fig. 10—ONE outputs from a PBT: (a) 64 outputs from a middle plane with noise canceling pattern and no read pulse amplitude variation. (b) 128 outputs from near end and far end planes with maximum noise pattern and extreme read pulse amplitude variations.

memory plane. Since the outputs are from the same plane, there is very little variation due to transmission line effects. All the bits are read with a constant amplitude read pulse and a minimum noise information pattern has been stored in the rest of the bit line. By contrast, Figure 10b shows 128 ONE outputs—64 from a near end plane and 64 from a far end plane. The bits are read with both

extremes of read pulse amplitude and the information pattern ends
to maximize the negatively going noise output. A comparison of the
two photographs shows how the area between the bundle of output
traces and the zero line decreases as the number of operating variables
is increased.

2.5.3 *Transmission Delay*

The average transmission delay of an output signal at room tem-
perature is about 10.6 ns per memory plane. This varies about ±6
percent with information content. A bit line having all ONES or all
ZEROS stored will have a total delay of about 640 ns. That same bit
line with alternating ONES and ZEROS on adjacent bits will have a
total delay of about 720 ns. This change in delay is due to the fact
that bit line inductance is related to the information content of the
bit line.

2.5.4 *Temperature Effects*

Magnetic metals with Curie temperatures of 700°C to 800°C, well
in excess of any practical operating temperature, from the basic
memory element in the PBT memory and, therefore, the character-
istics of the memory cell show no temperature effects. However, both
access and shuttle suppression cores do show temperature effects. The
transmission line characteristics of the bit lines change slightly due
to temperature because of the conductivity changes in the copper
core wire and the dimensional variations in the plastics. All these
changes are small. The only temperature effect that is important
enough to require compensation is the output signal delay. As the
temperature increases, the output is generated earlier in time because
of the changing characteristics of the ferrite access core. This change
is nominally 1.0 ns per °C.

2.6 *Worst-Case Testing*

In a memory that has as many operating variables as the PBT, an
extensive test sequence is needed to guarantee all the requirements
implied by random access, electronic writability and nondestructive
readout performance. This is accomplished by a sophisticated test
machine that uses a small digital computer for address and function
control, and pulse measuring and recording equipment for data dis-
play. The test sequence is a combination of worst-case information
patterns, including history effects, and extreme values of drive cur-

rents. Testing at temperature extremes is performed on a sample basis. Outputs under all these variations are tested for proper polarity, amplitude, and duration. The output noise level prior to the signal interval is also tested as an additional check on memory quality. Figure 11 shows a block diagram of the PBT test facility.

III. ACCESS

3.1 *General*

The access system must provide the regulated read or write current drive pulses to the PBT modules. It also provides the regulated DC bias for the biased core access matrix. The access method and many of the circuit designs are similar to those used in the permanent magnet twistor[3] program store of ESS No. 1. Circuits were modified wherever necessary to accommodate the higher current levels and the different timing of the PBT module.

3.2 *Operation*

Each PBT module has 64 X- and 64 Y-access lines. Four PBT modules are arranged in a two-by-two matrix with X or Y lines from adjacent modules connected in series as shown in Fig. 12. The result is a 128 by 128 array of access wires. To select a desired word, a current is applied to the proper X and Y wire.

Figure 13 shows the general arrangement of the access system for
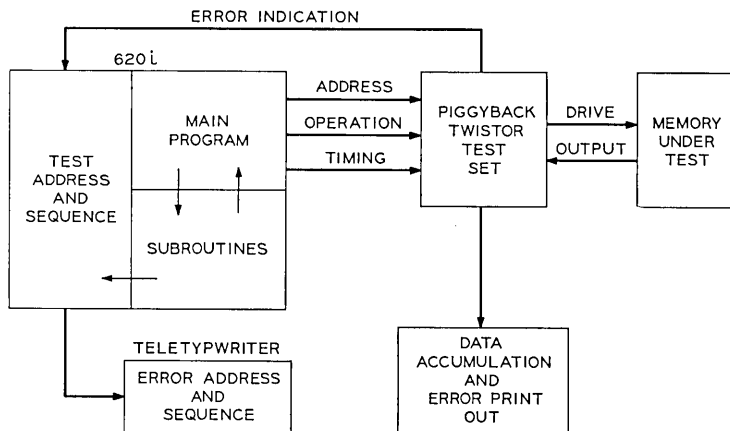
Fig. 11—PBT test facility block diagram.
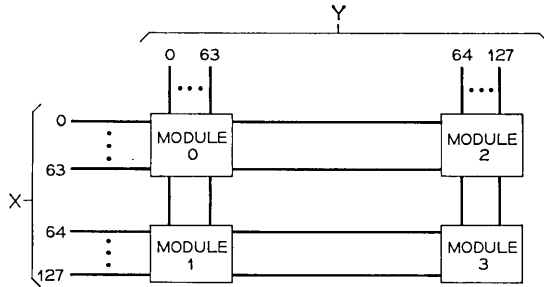
Fig. 12—Module access arrangement.

one axis (X or Y). This arrangement is duplicated for the other axis. One bias current regulator, which is not shown, is provided for the whole store. The 128 wires are arranged in an 8 by 16 array. The particular access wire is then chosen by selecting one out of eight lower access switches and one out of sixteen upper access switches.
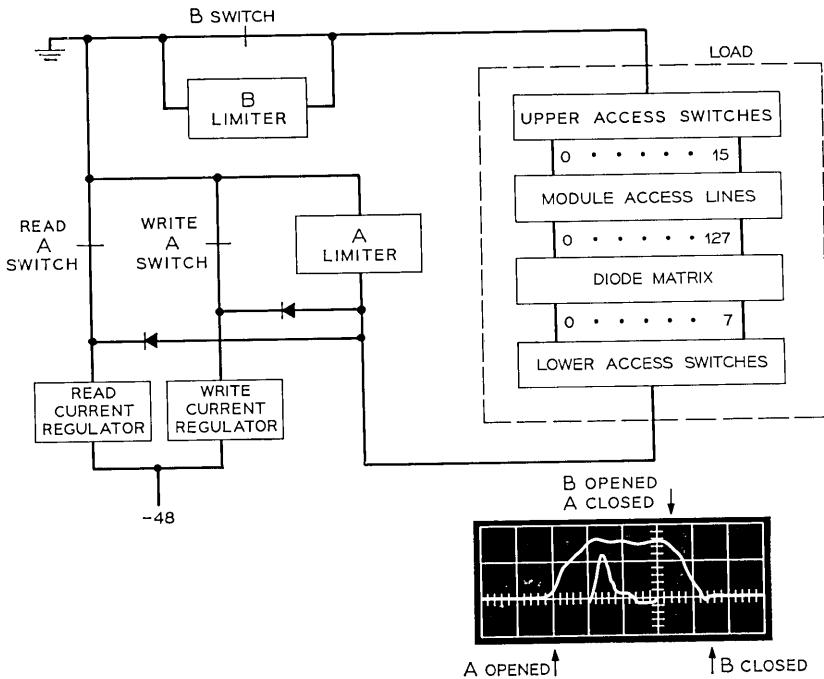

Fig. 13—Access system for one axis.

Three bits of the address are used on each axis for lower switch selection and four bits for upper switch selection.

The regulated read and write currents normally flow to ground through their associated A switch. When the module is to be accessed, the address information is used to first turn on the access matrix switches associated with the selected address. After allowing sufficient time for the matrix switches to operate, the desired (read or write) A switch is pulsed open, forcing the current to flow through the associated steering diode, the access matrix switches, and the B switch to ground.

The A limiter clamps a constant voltage across the access system during access rise time. This forces a current ramp through the primarily inductive access matrix. The access current waveform is shown in Fig. 13. When the current reaches the flat top level, the A limiter ceases conduction.

One microsecond later, the A switch is closed and the B switch is opened. This initiates the fall time interval. The B limiter clamps the matrix voltage to set the fall time. After the matrix current reaches zero, the B switch is closed and the access matrix switches are opened.

3.3 *Current Regulators*

The active series current regulators represent a major departure from the switching mode regulators used for the PMT. Figure 14 shows a partial schematic of the regulator.

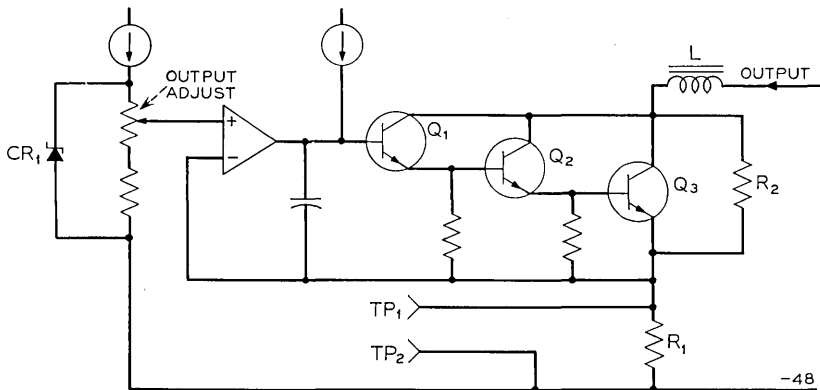The circuit functions by comparing the drop across R1 to a regu-

Fig. 14—Access current regulator.

lated voltage and adjusting the drive to the Darlington configuration of transistors Q_1, Q_2, and Q_3 to maintain the voltage equal to the reference. The reference is an aged zener diode (CR1) driven by a constant current source. The "output adjust" potentiometer provides some adjustment range to compensate for initial tolerance on CR1 and R1, and offset in the differential amplifier. The reference resistor (R1) is a precision power resistor chosen for good initial tolerance and tight load-life tolerance.

The differential amplifier is a matched transistor pair, with a constant current load. The series regulator uses a triple Darlington circuit to minimize base drive effect on output current. Both Q_2 and Q_3 are mounted on heat sinks to accommodate the high power dissipation. Resistor R2 shunting the series transistor provides most of the output current, thus lowering dissipation in the series transistor.

The regulator loop is stabilized by a single capacitor across the differential amplifier. Test points TP1 and TP2 are provided to allow a field check of the regulating voltage (nominally 6.00 volts). The inductor (L) absorbs the module back voltage during access current pulsing, or, on the bias regulator, provides the high bias source impedance needed by the access matrix.

This same design is used for the read, write, and bias regulators, with only the values of R1 and R2 changed to provide the currents shown in Table I.

The flat top regulation is determined primarily by the regulation of X, Y, and bias regulators. End-of-life regulation is ± 2 percent in each case. Regulation is primarily controlled by the current sensing resistor in the regulator (R1 in Fig. 14). Measured short term regulation with power supply (42.7 to 52.5 volts) and ambient temperature (2 to 50°C) variation is less than ± 0.35 percent. The ± 2 percent tolerance of the individual regulators is magnified by the access matrix, resulting in a ± 12.5 percent solenoid drive variation which meets the PBT memory module requirement.

IV. READ

4.1 *General Organization*

In the module arrangement shown in Fig. 12, whenever a given module is accessed, the half-access current of either X or Y access alone flows through two other modules. Although shuttle suppression cores are used, the shuttle noise during readout is still significant.

The module which is diagonal to the selected module however, receives no half-access currents, and so has a quiet bit line. Therefore, the bit lines of modules 0 and 3 are connected in parallel, and those of modules 1 and 2 are connected in parallel. This always places the outputs of a quiet module and a fully-accessed module in parallel. The paralleling has the undesired effect of reducing the readout signal amplitude by a factor of two, but it does require only one-half as many readout preamplifiers, as well as simplifying preamplifier selection.
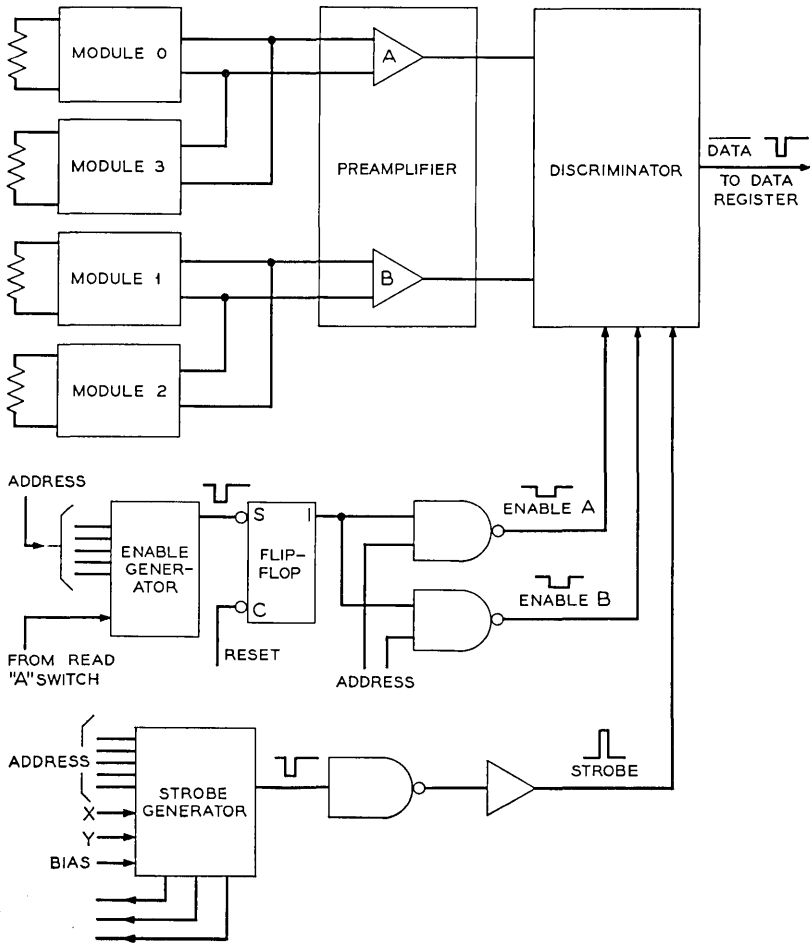
Fig. 15—Readout system.

Fig. 16—Relative readout system timing.

Figure 15 shows the block diagram of the readout system. The outputs of the two preamplifiers, A and B, are fed to the zero-crossing discriminator, where selection is made by using two separate enable signals, controlled by X and Y address bits. The enable signal is started when current through the access A switch is interrupted (at about 10 percent of access drive). The enable generator then introduces both a fixed and an address-dependent delay before generating its enable pulse to set the read enable flip-flop. This flip-flop is reset after strobing of the information into the data register.

The signal from the zero-crossing detector in the discriminator is gated to the data register by the strobe pulse. Since the strobe timing is more critical to proper operation of the readout system, it is generated with greater relative timing accuracy. This is accomplished by starting the strobe reference timing from a fixed point on the read access overdrive. The proper timing is derived by using X, Y, and bias currents from the access system and triggering the delay start with the optimum 1.0 ampere-turn overdrive point. The strobe generator then produces its fixed and address-dependent delays and generates the 150 ns strobe pulse. This is then amplified and fed to the discriminators. The relative timing of these signals is shown in Fig. 16.

4.2 *Preamplifier*

The readout amplifier raises the output of the PBT modules to a level of 500 mv sufficient for gating and polarity detection in the discriminator. It is a two-stage direct coupled amplifier with a minimum gain of 2000 and 6 dB response points at approximately 150 KHz and 1.5 MHz. The low frequency cut off is necessary to insure fast amplifier recovery after a write current pulse. In addition, a resistor diode network is used at the input to limit the overload the amplifier sees during the write process.

4.3 *Discriminator*

The discriminator circuit provides selection of either the A or the B preamplifier and polarity detection of the selected signal. The output for a ONE signal is a low-going logic pulse used to set the associated data register flip-flop.

Figure 17 shows a partial schematic of the discriminator. The inputs from both preamplifiers are AC coupled to diode bridge clamp circuits. The clamp circuits provide a low impedance ground for the pre-amplifier signals until one side is unclamped by the enable pulse, which back biases the clamping diodes in that particular bridge. Resistors R1, 2, 3, and 4 form a resistive adder to couple the enabled signal into the differential detector $Q1$ and $Q2$. The output of $Q2$ is coupled through zener diode CR1 into the output gate, and ANDED with the strobe to produce the data output.
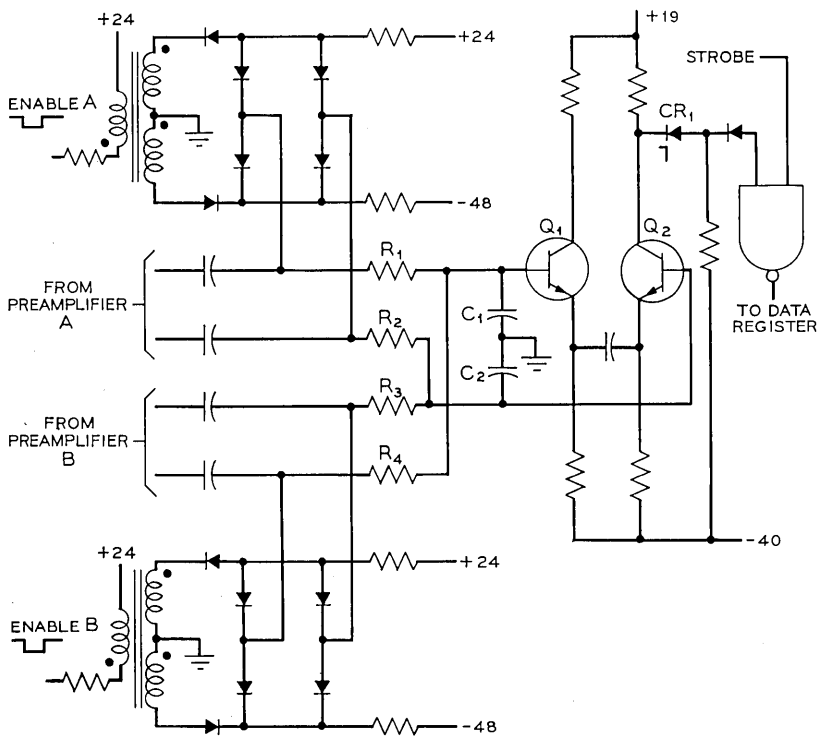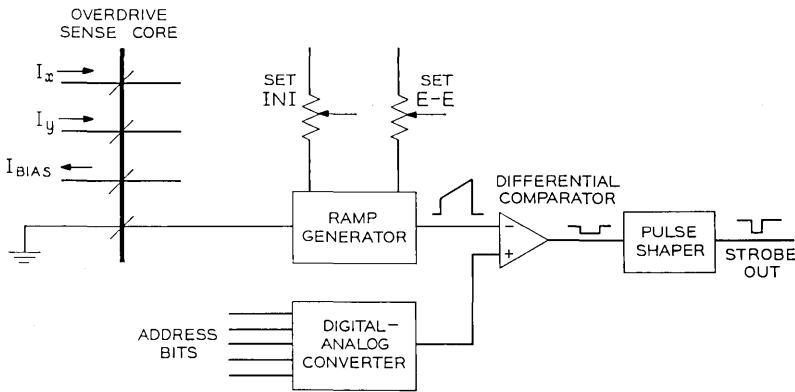
Fig. 17—Readout discriminator.

Fig. 18—Strobe generator.

The differential amplifier is greatly overdriven by the amplified output to provide zero-crossing detection. Capacitors C1 and C2 provide low-pass filter action before the differential detector. This effectively lengthens the readout signal, giving a larger operating window. Two discriminators are provided on a single circuit board, with the enable bridges being shared by both circuits.

4.4 *Strobe Generation*

The strobe generator and associated fanout amplifier chain must generate the address and temperature dependent strobe pulse and deliver this pulse to the 47 readout discriminators. Figure 18 shows a block diagram of the strobe generator circuit pack.

The overdrive sensing core is selected from PBT access matrix cores and threaded with the X, Y, and bias current leads. The turns are arranged so that it produces an output pulse at the 1.0 ampere-turn overdrive point. Using an access matrix core provides first-order temperature compensation for the output signal delay temperature effect described in Section 2.5.4.

The five address bits, which partially determine the plane to be accessed, are used as inputs to the digital-analog converter. The analog output then is proportional to the distance along the twistor tape from the readout amplifier, and thus proportional to the expected time the output will occur.

The pulse from the overdrive sensing core is used to start a ramp generator. The ramp generator output is then compared to the output

of the digital-analog converter, and the comparator output used to initiate the strobe pulse.

Adjustments are provided on the strobe generator board to set both the initial (address independent) and the end-to-end (address dependent) values of the delay. These are factory adjusted to compensate for component variations on the strobe board itself.

4.5 *Enable Generation*

The enable pulses for the discriminators are generated in much the same way as the strobe pulses. The accuracy of positioning is less critical, however, so the parameters are not as tightly controlled. The address tracking is accomplished in the same manner as for the strobe. The pulse must be considerably earlier than the strobe, however, so the start of the ramp is initiated by a pulse from the "*A*" access switch when it turns off. Starting it from overdrive would not allow it to be early enough for optimum positioning. The enable overlaps the strobe, so the pulse from the enable generator is used to set a flip-flop which is reset after the strobe interval. The flip-flop output is then used to drive high power logic gates for fanout to the discriminators.

4.6 *Performance*

Readout system performance has been excellent. The store operating margins are sufficiently large that once the strobe is adjusted the error rate is zero over all operating conditions. No failures have been encountered due to drift.

V. WRITE

5.1 *General Organization*

Figure 19 shows the circuits for writing one bit line in each module. Forty-seven of these circuits are provided in each store. The four modules are effectively connected in parallel during a write current pulse, and are isolated by the reverse biased diodes at other times. The writing of four modules in parallel is dictated by economics, and is the limiting factor on bit current regulation. The readout preamplifiers are connected across the near ends of two modules in parallel, and the far ends of all modules have resistive terminators (RT). The write current driver provides the bidirectional current switching and the current regulation.
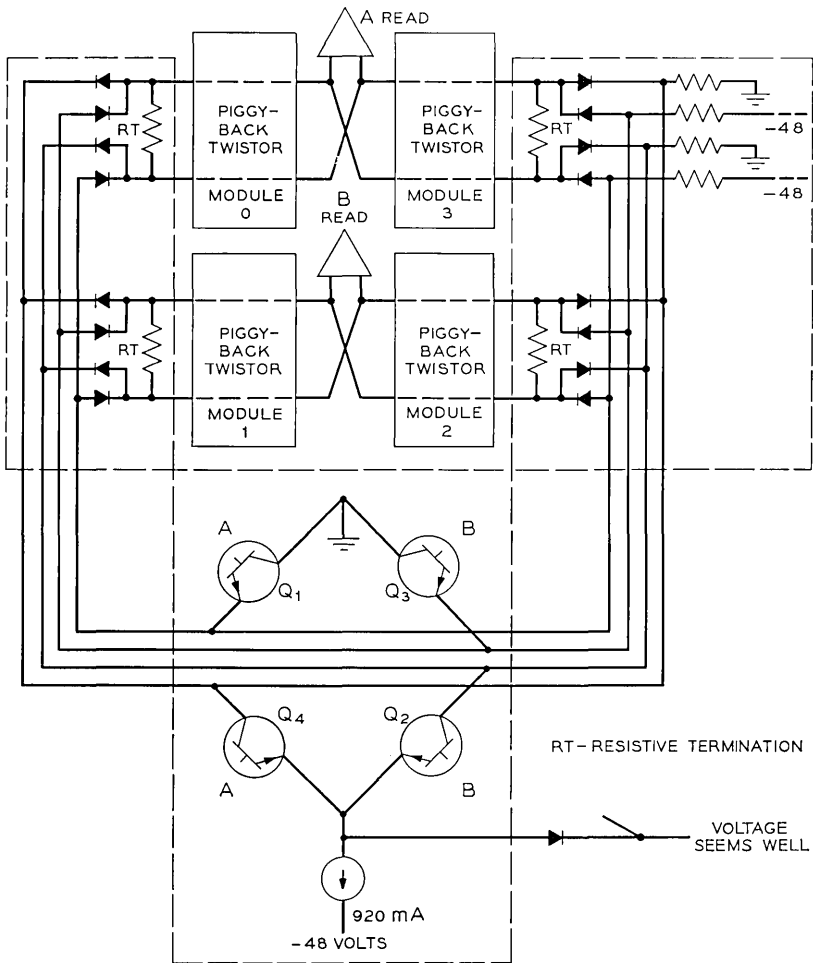
Fig. 19—Module bit interconnections.

5.2 *Bit Current Driver*

Figure 20 shows a simplified schematic of the bit current driver. To write a ONE, a positive write current must be followed by a negative post-write-disturb. For a ZERO, both currents must be reversed. The logic to accomplish this is done by the AND and NOR gates shown. The WP (write positive) and WN (write negative) pulses correspond to the "write" and "post-write-disturb" pulses for modules
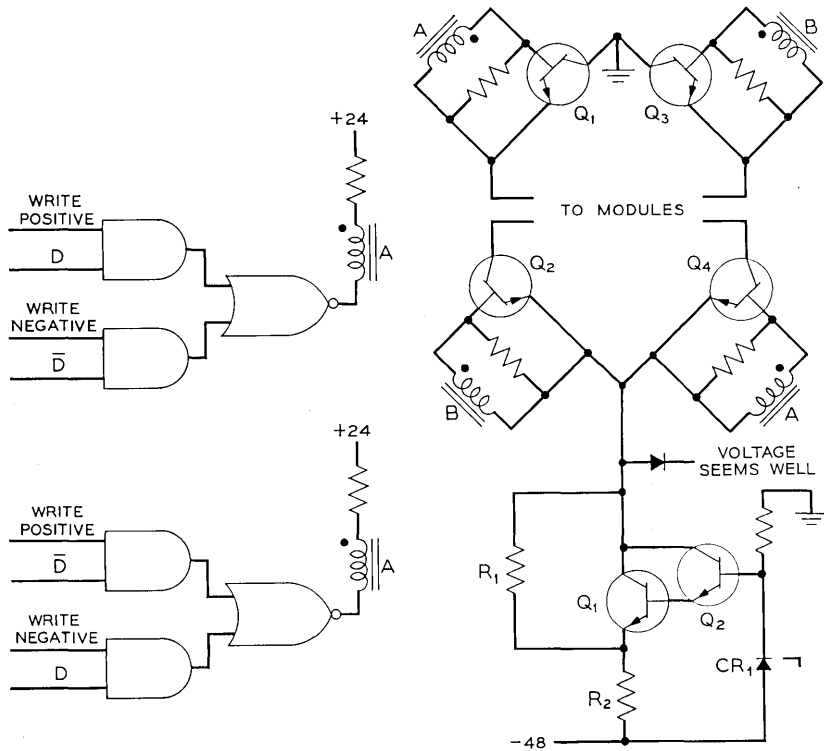
Fig. 20—Bit current driver.

0 and 1. Since modules 2 and 3 are connected in reverse to the readout amplifiers, the WP and WN functions are interchanged by logic preceding the bit current drivers during writing in these two modules.

The NOR gate outputs drive either the A or B transformers to generate drive for the proper two bridge transistors.

The current regulator is composed of the Darlington pair of $Q1$ and $Q2$, CR1, R1, and R2. The dissipation in $Q1$ is kept low by bypassing most of the current around $Q1$ by R1. One bit current driver is provided on a single circuit pack.

The voltage seems well (VSW) test point is monitored during the bit current pulse to check for shorted bridge transistors. If a bridge transistor is shorted, the VSW point will rise almost to ground, causing an ASW failure to be sent to the processor.

During the post-write-disturb interval, the current limiter is satu-

rated due to the high load inductance. At this time modules are driven by the −48 volt supply. To maintain the necessary volt-microsecond control, the length of the post-write-disturb is varied according to the potential of the −48 volt supply. The pulse generator circuit is designed to keep the volt microsecond area constant with supply voltage changes from −42 to −53 volts.

VI. READ-WRITE TESTS

The object of the read-write tests is to establish all the worst case operating conditions in the frame and then evaluate the read-write margins.

The variables that must be controlled to establish worst case read-write operating conditions are those which affect the module operation and are listed in Section II of this article. To carry out worst case testing required a facility utilizing a computer-controlled test set.[6] This facility is shown in Fig. 21. The computer uses a complicated algorithm to establish the testing sequence which is then applied to the store. The variables under computer control are listed in Table II. Since this facility is very flexible many testing sequences have been checked that were suspected of being worst case thus evolving a very comprehensive test.

To measure the read-write margins, the readout strobe timing is varied while store outputs are checked against expected outputs. There are two timing adjustments on the strobe board; initial (INI) and end-to-end (E-E) delay. The range of these two adjustments for correct outputs is a measure of the store frame operating margins.

For read-write tests, both strobe adjustments are put under computer control and each can be set to one of 32 possible values. The CCSTEVA (computer-controlled store evaluation) program determines the strobe settings at which the store will pass all tests at all addresses. It then produces a plot of failing and passing settings with the E-E

TABLE II—COMPUTER CONTROLLED STORE VARIABLES

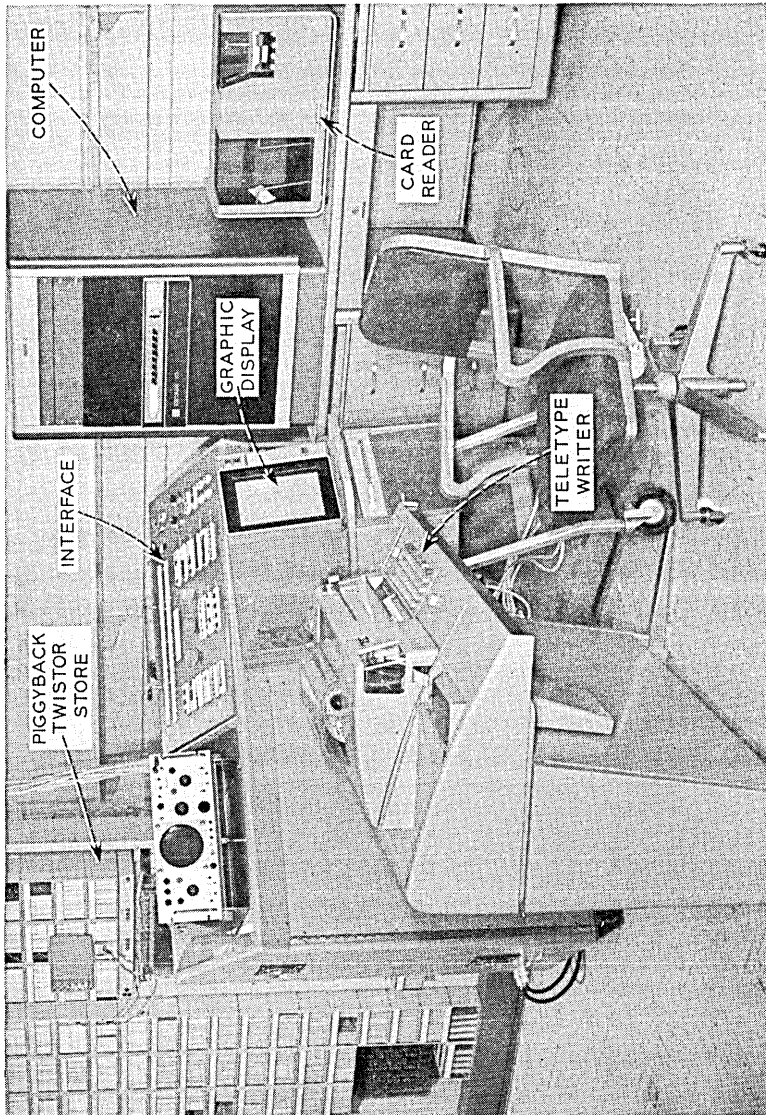| Variable |
| --- |
| Read-Write Program |
| Strobe Timing |
| Enable Timing |
| Access Drives |
| Bit Drives |
| Frame Supply Voltage |
| Post-Write-Disturb |

Fig. 21—Computer-controlled test facility.

```
FILE OUT 0

FILE 0     DESCR= 01    PROG=05      03/14/69      PROT=0
INH= 00     RUN= 1    MOD= ALL    Q.C.=0541

CCSTEVA

      31  +++++++++++++++++++++++++++++++++++
      30  +++++++++++++++++++++++++++++++++++
      29  +++++++++++++++++++++++++++++++++++
      28  +++++++++++++++++++++++++++++++++++
      27  000000000000+++++++++++++++++++++++
      26  0000000000000000+++++++++++++++++++
      25  000000000000000000+++++++++++++++++
   I  24  00000000000000000000+++++++++++++++
   N  23  000 00000000000000 0000+++++++++++++
   I  22  000 0000000000000 00000+++++++++++++
   T  21  0000 000000000000 00000+++++++++++++
   I  20  +000 00000000000000 0000++++++++++++
   A  19  +0000 000000000000 00000++++++++++++
   L  18  ++0000 0000000000000 0000++++++++++
      17  ++00000 00000000000000 0000+++++++
      16  +++0000 0000000000000 0000+++++++
   D  15  ++++0000 0000000000000 00000+++++ +++      DURING PRODUCTION
   E  14  +++++0000 0000000000000 0000++++++       THE STORE IS TESTED
   L  13  +++++0000 0000000000000 00000+++++       FOR PROPER OPERATION
   A  12  ++++++0000 0000000000000 00000+++++       AT ALL POINTS WITHIN
   Y  11  +++++++0000 0000000000000 0000++++       THIS BOUNDARY
      10  +++++++0000 00000000000000 0000+++
      09  ++++++++0000 000000000000 0000+++
      08  ++++++++00000 00000000000000 0000++
      07  +++++++++0000 000000000000 00000+
      06  ++++++++++0000 0000000000000 00000
      05  ++++++++++00000 00000000000000 0000
      04  ++++++++++++0000 00000000000000 0000
      03  ++++++++++++00000000000000000000
      02  +++++++++++++00000000000000000000
      01  +++++++++++++++++++++++++++++++++++
      00  +++++++++++++++++++++++++++++++++++
           ^           ^           ^          ^
           0          10          20         30
                      E-E   DELAY
```
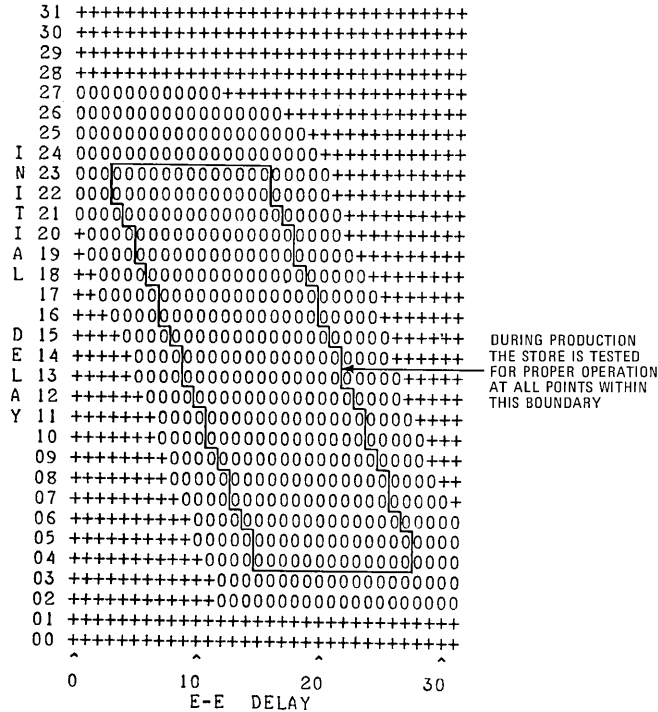
Fig. 22—Strobe window.

values as the abcissa and the INI values as the ordinate. Such plots are known as "strobe windows" and one is shown in Fig. 22. It was taken on a store at room temperature with all drive currents held at nominal. The read-write program was worst case and took about 120 minutes to run. Also shown is the production test requirement.

The computer-controlled test set has been used extensively in the development of the PBT store. It is also being used for production testing. Field experience has shown that a store whose strobe board is adjusted optimally with respect to the strobe window will operate without errors as long as there are no circuit faults.

TABLE III—PBT STORE COMMUNICATION LINKS

Unit	Type Signal	Wire Pairs	Description
Input From Processor	Unipolar 0.5 μs Pulses	71 Data 5 SYNC 1 Write Go	Duplicated address bus from the processor. Each bus is shared by half of the stores in the system.
Output to Processor	Unipolar 0.5 μs Pulses	47 Data 2 ASW 1 SYNC	Duplicated answer bus to the processors. Each bus is shared by half of the stores in the system.
Input From Central Pulse Distributor	Bipolar 0.5 μs Pulses	2	Dedicated leads to control the trouble and protected area flip-flops.
Input From Master Control Center	DC	2	Dedicated leads to control the trouble flip-flops.
Output to Master Control Center	DC	2	Dedicated leads to indicate the state of the trouble flip-flop and the power relay.
Input From Signal Distributor	DC	8	Dedicated leads to control the several scan relays and the out-of-service lamp.
Output to Scanner	DC	56	48 scan leads shared by all stores and 8 dedicated scan leads.

VII. COMMUNICATIONS

The major communication links between the processor and the PBT store are the address and answer buses. The signals on these buses are 0.5 μs pulses for binary ONES and no pulses for ZEROS. The address bus carries the address, control, and data information from the processor to the stores and the answer bus carries data back to the processor. The bus systems and stores are completely duplicated with each store having access to only one bus system but with each processor having access to both bus systems. All store inputs and outputs are listed in Table III.

VIII. STORE OPERATIONS

Store operations are divided into normal and maintenance. There are three normal operations, normal read, normal write, and inactive. When the store is faulty or suspected of being faulty, various main-

tenance operations are possible. Three of these, the bus-register test (BRT), control read, and control write operations are powerful tools that use the full address and answer bus facilities of the store. If necessary, all bus communications with a given store can be stopped by setting a flip-flop in that store. This can be used to silence a store which is signaling on the answer bus when it should be quiet. When this flip-flop, which is called TBL (trouble) is set, DC measurements can be made on the store by the scanner. The TBL state can be modified by setting the PORT (partial override of trouble) flip-flop to allow the store to receive address bus inputs and execute instructions but not respond. This state is used in the store update operations after a store has been out of service.

Figures 23, 24, and 25 illustrate the major circuits that are used for store operations. On all bus operations, the 24 bits specifying the store name, operation, and address are gated to the proper registers in all stores by the SYNC pulses. If the transmitted name matches the name that is wired in the decoder, the store goes active. If not, the store executes an inactive operation.

During each bus-controlled operation the ASW (all-seems-well) circuit gathers data on the operation of critical store circuits. If they all give indication of functioning properly, it then sends an ASW signal to the processor. If the signal is not returned, the processor retries the command. If it fails again, the processor forces a transfer to the maintenance programs.

8.1 Normal Operations

Figure 23 shows the major information flow for the normal read, normal write, and inactive operations.

8.1.1 Normal Read

On a normal read, the addressed location is interrogated and the resulting 47 bits are passed to the data register and then to the processor. The data is gated out sometime between 3.1 microseconds and 3.8 microseconds after receipt of the read command from the processor.

8.1.2 Normal Write

The normal write operation consists of a store read (called preread) and store write. The preread operation is a normal read with the exception that the name, operation, and address registers are not reset at the end of the cycle. This serves two functions: Often, only part of
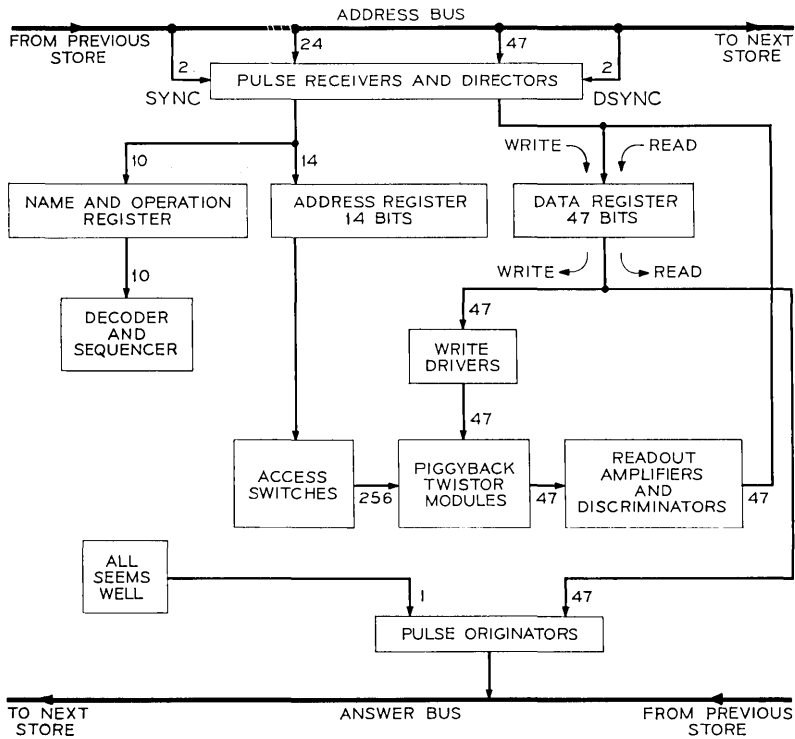
Fig. 23—Information flow for normal operations.

a word is to be modified and the preread is necessary to obtain the other bits that will not be changed since all bits must be written simultaneously.

The other preread function is that it provides a fail safe protection of the processor to store address path. The processor makes a Hamming check of the data and address of the preread cycle. If this check is passed, then the proper address has been locked into the store. Now the processor sends the 47 bits of data for the write operation which are received by the store and are held in the data register. The store writes this data at the location just read. Thus a check has been made on the whole address path from the processor to the store assuring that the correct location is written. In addition, a "write go" pulse from the processor is required during the write cycle by the store. This pulse initiates write access. The absence of this pulse causes the

cycle to abort and fail to send an ASW signal. The store is ready for another cycle 50.4 μs after the start of the preread cycle.

The PBT store has a protected area feature that prevents programs from writing into certain preselected areas. The protected area is specified by permanent wiring in the decoder. There are seventeen possible protection options; no protection, $\frac{1}{16}$ protected, $\frac{1}{8}$ protected, etc., to all protected. All areas protected are continuous and start at address 2. Addresses 0 and 1 are always unprotected for maintenance purposes. If a write is attemped into protected area, it is recognized by the decoder on the preread operation and the store fails to go active. Since an inactive store does not respond, the processor detects an ASW failure and calls maintenance programs.

All program and translation data are located in protected areas of the store so that they will not be destroyed by programs that attempt to write in the improper area. Since maintenance programs are called by these attempts, many program bugs can be uncovered by analyzing attempts to write in protected area.

8.1.3 *Inactive*

All stores receive and register the full information for each operation. Only the store that recognizes its name will go active. The other stores perform a simple sequence. The most important part of this sequence is resetting all registers and verifying the reset with the all-seems-well circuits. Thus, if during a period of inactivity, a transient sets a register in a store, the register will be reset on the next operation in any store.

As was noted above, a write cycle takes 50.4 μs. Much of this time is spent applying bit current and then waiting for the read circuits to recover. The processor has completed its write cycle after two cycles (12.6 μs) and is ready to read again. Therefore, all inactive cycles are 6.3 μs long and if the next operation after a write is in a store that is different from that just written, it will commence two cycles after the write began. This write overlap feature allows a write instruction to consume between three and nine system cycles (one cycle is added to read the write command).

8.2 *Maintenance Operations*

Store maintenance operations are used both for store fault recognition programs and store diagnostics. The store fault recognition programs, using these operations, run a very brief set of tests on a suspect

store and determine whether the store should be diagnosed or remain
in service. The store diagnostics, using these operations, run a lengthy
sequence of tests which will resolve a majority of the faults to a small
number of circuit packs.

The communication bus is used to execute the bus-register test
(BRT), control read and write operations, and to set up the trouble
mode. These are all 6.3 μs operations. The signal distributor is used
to set up the scan mode. Scanning proceeds at relay speeds.

8.2.1 BRT (Bus-Register Test)

The BRT operation is used for verifying the proper operation of
registers, sequencers, and bus communications. The information flow
for this operation is shown in Fig. 24. In this operation the six-bit
name and the 14-bit address are formed into a 20-bit data word,
double-gated into the data register and outpulsed to the processor. The
Hamming bits cannot be checked by this method, but if this test
passes, the diagnostic program can proceed. The Hamming bits will be
checked in a later test.
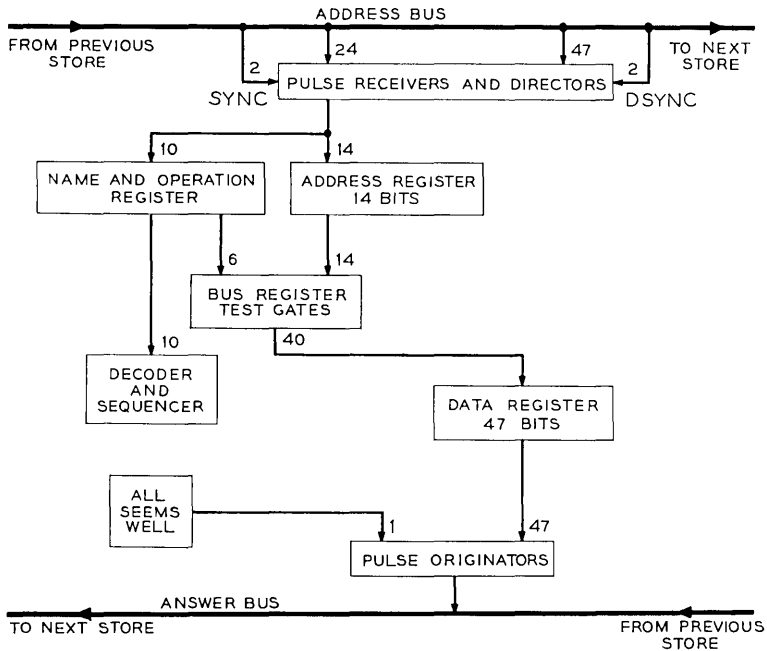
Fig. 24—Information flow for BRT operation.
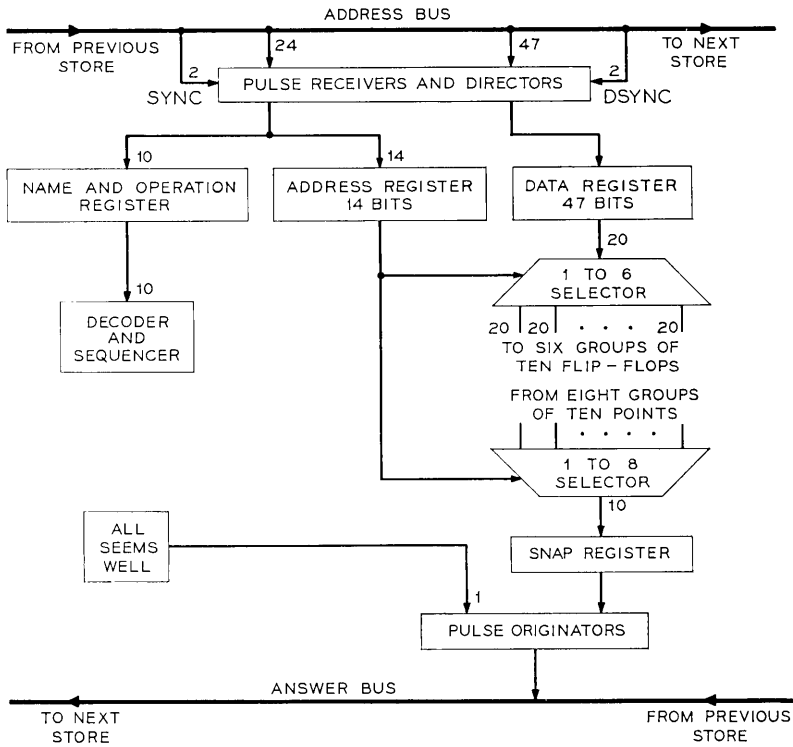
Fig. 25—Information flow for control operations.

8.2.2 *Control Read*

The control read operation is used by the maintenance programs to determine the state of a group of store circuits. Figure 25 shows the information paths for this operation. At one of three times in the cycle the states of a group of ten flip-flops are gated into the SNAP register. This register holds the data until it is time to send it on to the processor. Any one of eight groups of ten flip-flops may be specified. These 80 flip-flops comprise most of the control and maintenance flip-flops in the store.

8.2.3 *Control Write*

The control write operation is used to set the state of important store circuits. Figure 25 shows the information paths for this operation. One of six groups of twenty flip-flop gates can be pulsed in a

control write operation. After these points have been written, their states are gated into the snap register and can be read on a subsequent control read to verify the control write operation.

8.2.4 *Delayed Snap*

The delayed snap operation makes it possible to get a control reading on a normal operation by using both a control write and a control read. This is important for maintenance since many store functions are inhibited during maintenance operations.

To make a delayed snap requires a control write of the snap flip-flops to establish on which of the next seven operations the control reading is to be taken. On the specified operation, at the specified point in the cycle, the states of the selected group of ten flip-flops is gated into the SNAP register. A subsequent control read brings this information to the processor.

8.2.5 *Scan Mode*

The scan operation provides DC information about various circuits in the store. In this mode relays are operated to connect a 1.3 KΩ ferrod scan point in the scanner into the desired store circuit. The scanner is interrogated by the processor under program control and returns a ZERO if more than 3.9 mA flows and a ONE if less than 1.8 mA flows. This mode is used to test the output of the voltage regulators, the fanout diode in the timing circuits, the states of many flip-flops, and the state of the store power relays.

IX. MAINTENANCE

Every store fault must be repaired as rapidly as possible. Figure 26 shows the process used to repair stores. The next sections describe the three techniques used to identify the fault.

9.1 *Diagnostic Program*

All attempts to repair stores begin by running the diagnostic program which usually generates a trouble number. The diagnostic program usually is called automatically because of either massive operation failures, the failure of a scheduled exercise, or a fuse alarm.

Occasionally a fault will occur that causes a very low number of store operation failures (typically less than 30 in any 30 second interval). If these persist, the maintenance man will begin his diagnosis by calling the diagnostic program.
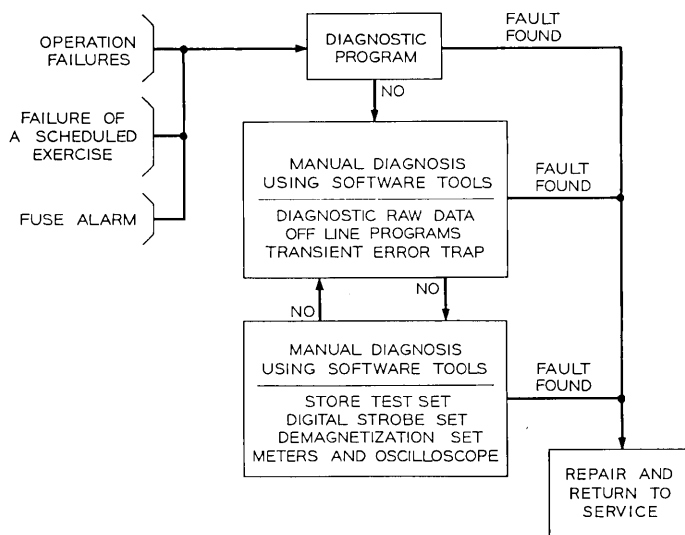
Fig. 26—PBT store maintenance process.

If the diagnostic program produces a trouble number, the mainte-
nance man looks up the trouble number in the trouble locating manual
(TLM). He usually finds the number listed and along with it is a list
of circuit packs whose failure result in that number. He replaces these
packs, one at a time, until the store operates properly.

Studies have shown that about two-thirds of the store faults will
result in a trouble number that can be found in the TLM. Studies
of the TLM show that 65 percent of these matches will identify three
or less circuit packs. However, 10 percent identify more than nine
circuit packs and may require manual methods to resolve the fault.

The remaining third will result in no trouble number or a random
trouble number due to marginal fault conditions. In these situations
manual diagnosis must be used. To date the manual diagnostic tools
have proved satisfactory. As a matter of fact, several very subtle faults
have been found readily and rapidly using these tools.

9.2 *Manual Maintenance Using Software Tools*

If a trouble number is not listed in the TLM, the maintenance man
can request the diagnostic program to print out all the raw data on
which the trouble number is based. This data shows exactly which

store tests failed. By analyzing the raw data he can determine areas of the store which should be checked for failure.

It may be necessary for him to cause the store to sequence to isolate the problem. To this end a set of "off-line" programs are provided that will allow him to sequence the store repetitively through as many as twenty operations. He can then use an oscilloscope to dynamically view the suspected store areas.

If the fault is a transient operational failure and the store passes the diagnostic program, it is necessary to recreate the conditions that exist when this failure occurs. To do this, programs and hardware are provided that will enable the maintenance man to pinpoint what program steps were being executed at the time of failure. He can then use the off-line program to set up this condition and to trace the fault.

9.3 *Manual Maintenance Using Hardware Tools*

When software approaches have failed to locate a store fault and the problem appears to be in the read-write circuits, it is necessary to resort to manual testing. Several pieces of equipment have been provided in each TSPS office for manual maintenance: the SPC store test set, the digital strobe set and the PBT demagnetizer set. When this equipment is used, the store is taken completely off-line and is manually disabled from communicating with the rest of the system.

The SPC store test set connects to the store via a special test connector. It gains direct access to the sequencer, decoder, read, write, access, and ASW circuits. It has facilities for sequencing several severe but not worst case read-write programs through every address in the store. On every read operation it checks the data read out and will indicate errors. Most marginal read-write or ASW faults are easily located with this test set.

The digital strobe set is used with the store test set for marginal checks on the store. By varying the strobe settings while the test set is running test patterns, a check may be made on the strobe window (see Section VI and Fig. 22). Figure 27 shows the test set and strobe set connected to a store.

If a portion or all of a bit wire is bad, there is a possibility that the memory material in the interbit regions has become magnetized due to circuit pack malfunction. To remedy this, the PBT demagnetization set is connected to the bit wire and operated. A decaying 60 Hz sine wave is applied to the bit wire which leaves it completely demagnetized.
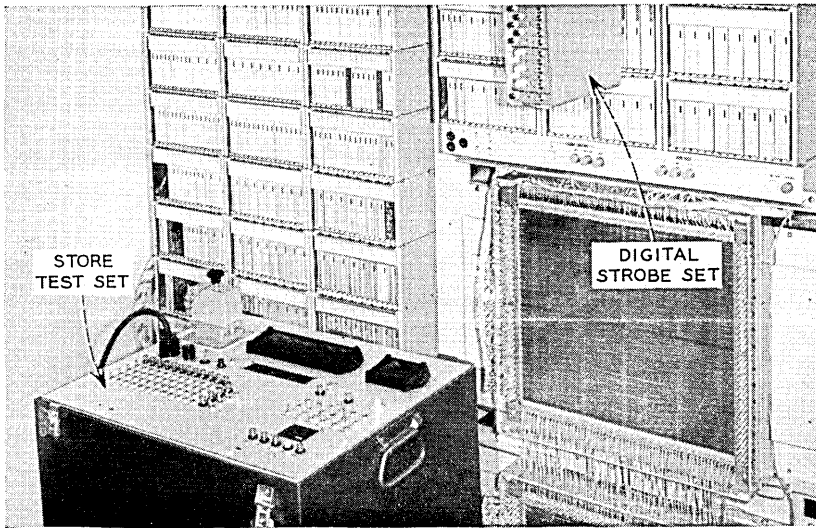
Fig. 27—PBT store test set and digital strobe set connected to a store.

X. PHYSICAL DESIGN

10.1 *Frame Description*

The store physical design uses No. 1 ESS style frameworks and packaging techniques[7] wherever possible. The store physical design problem can be stated by four general requirements:

(*i*) Package the store circuitry within a single 7-foot high double-bay framework.

(*ii*) Separate circuit functions to eliminate noise interference that would cause erroneous circuit operations.

(*iii*) Locate circuit functions to minimize interconnecting lead lengths.

(*iv*) Provide adequate thermal dissipation for high wattage components and isolate them from temperature sensitive circuitry.

The store, shown in Fig. 1, consists of one double-bay framework that contains four piggyback twistor (PBT) modules; access, read, and write circuits; and other related circuitry. The location of the major circuits is shown in Fig. 28. ESS No. 1 type circuit packs are used to package the majority of the store circuitry. Over 600 of these boards, representing 58 different codes, are required.
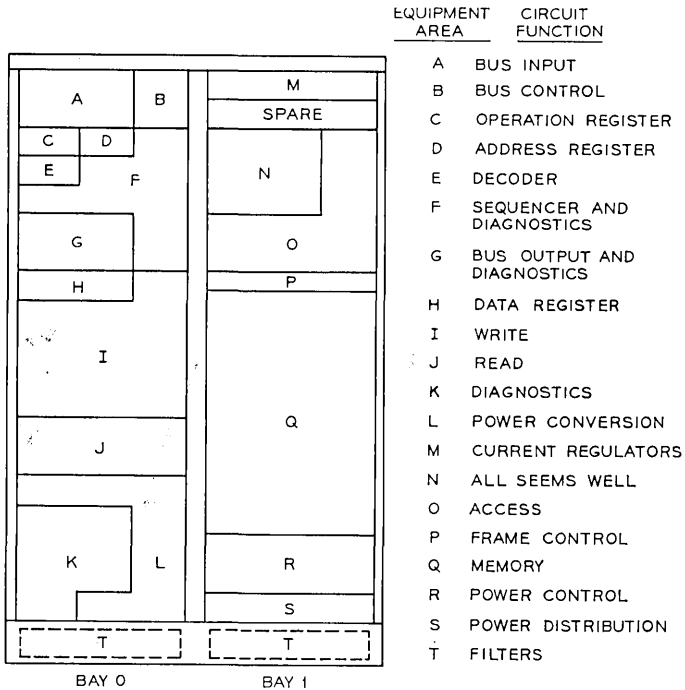
Fig. 28—SPC No. 1A store showing the location of the major circuits.

The requirement to package the store circuits within a single double-bay framework requires that the PBT modules be mounted within 34 inches of bay space. This objective is accomplished with a back-to-back mounting of the four modules, which also shortens interconnecting cable lengths. Of particular importance was the need to separate the access circuit, which produces 2.0 ampere pulses, from the logic, register and read circuitry. To this end, bay 1 contains the high level access and power distribution circuits along with the four back-to-back PBT modules. All other store functions are contained within bay 0.

Five current regulators that supply access and bias current for the modules contain components operating at temperatures up to 250°F. The regulators, in accordance with requirement (iv), are mounted at the top of bay 1 on a single mounting plate. A 4-inch wide unblocked area is retained below the regulators to allow convective cooling of

regulator components. A detailed description of these regulators is given in Section 10.3.1.

10.2 *Power Dissipation*

Each SPC No. 1A store frame in an idle or read state dissipates 1240 watts. An additional 585 watts may be introduced by writing at the rate of one write operation every 157 μs in a particular store. This represents the highest average writing frequency of present SPC No. 1A applications. Presently a maximum of 40 stores, dependent on memory requirements, may be provided with the SPC No. 1A. Thus, the store has considerable effect on office ventilation[8] and power plant requirements.

Approximately 62 percent of the total power dissipation is concentrated in three areas; the five current regulators, the four PBT modules and the write circuit packs. Each of two read current regulators dissipates approximately 75 watts, each of two write regulators 95 watts, and the bias regulator 135 watts. For the highest expected write rate, each of the four PBT modules dissipates 40 watts, including 20 watts of bias. The write circuit consumes 505 watts. Power levels of these magnitudes require special design treatment to avoid high component operating temperatures which increase failure rates. The following section deals with the thermal design considerations in some detail.

10.3 *Thermal Characteristics*

The thermal characterization of the store was achieved by four laboratory experiments involving the 15B modules, the current regulators, and the circuit packs within bay 0 of the frame.

10.3.1 *Current Regulator*

Five current regulators are required to supply bias and read and write access current for the PBT modules. The regulator circuit is an active series type regulator utilizing a power transistor and resistors as the series elements. A plug-in unit design was adopted because it offers the advantage of a fully tested and adjusted functional unit. Also, it is readily accessible for maintenance or replacement in a working store with minimal downtime. Of primary importance for a final overall unit design is the high power dissipation that is a characteristic of a series type regulator circuit. The three design objectives needed to accommodate this high power dissipation are:

(*i*) Provide efficient heat-transfer mechanisms to limit component operating temperatures to safe values.

(*ii*) Isolate power dissipating components from thermally sensitive regulator components.

(*iii*) Limit the exposed surface temperature of the package to 140°F to prevent personnel injury when extracting the regulator.

A laboratory evaluation of models indicated an economical approach to produce a plug-in unit (see Fig. 29) that meets these three objectives. A conventional heat sink was found adequate to limit the transistor temperature within its rating. One quarter inch thick black lacquered aluminum parts were used to construct a rectangular heat sink mounting for the resistors. To achieve requirements (*ii*) and (*iii*) a package design utilizing six thermally isolated areas within the regulator was conceived and evaluated. The principle form of heat transfer between the six areas is conduction, whereas the principle form of heat transfer from the unit is convection.

Laboratory evaluation of regulator models of the compartmented design indicated that only semi-isolation between the six areas was essential. Therefore, only some degree of restriction of conductivity was
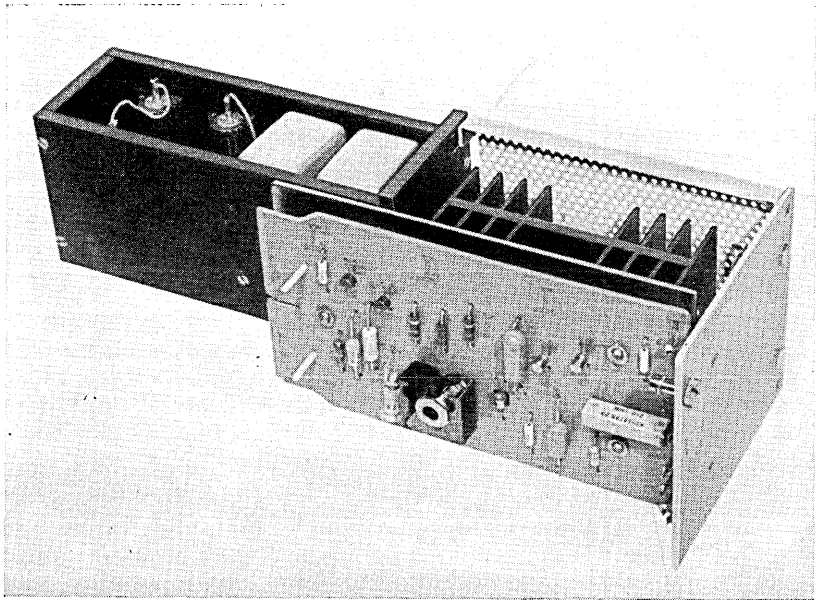
Fig. 29—Current regulator.

necessary to accomplish the objectives. This thermal restriction was accomplished by reducing the total area of contact between sections to four small connecting areas. This reduced cross-sectional area between sections established a thermal gradient large enough to greatly reduce the thermal effects of the resistors and transistor on each other and on the more sensitive components in the regulator. Perforated steel, 50 percent open, further increased the temperature gradient along the right side plate to the front panel.

Power resistors and transistors, which normally operate at elevated surface temperatures, were located internally to minimize danger to personnel. Heat from these components is dissipated by conduction to heat sinks to insure that temperature limits are not exceeded. Somewhat in opposition to this arrangement is the requirement that outside surfaces of the unit must not be a hazard to maintenance personnel. Accordingly, the design provides thermal isolation for the front panel and right side plate which limits these surfaces to 135°F even under maximum operating conditions. This is within the 140°F design limit specified by objective (iii).

10.3.2 *15B PBT Module*

The 15B module (see Fig. 2) contains two heat sources, the digit lines, and the bias winding. The principle form of heat transfer from the module is by convection from the front surface of the memory planes and from vertical fins extending outward from the rear of the module. Each fin covers the area of a memory plane and extends past the edge of the plane by 0.65 inch. Thus a 0.65 by 12 inch area of each fin extends into a chimney formed by the back-to-back mounting of the modules.

Since the maximum internal temperature of the module must not exceed 180°F, a study was conducted to establish the relationship between module power dissipation and its internal temperature. Write rate was used as the variable parameter instead of power to provide an easily understood expression that can be directly applied by system programmers who determine store usage.

The write rate (t_p) is defined as the percent of the time required to perform a write operation in a particular store (50.4 μs) with respect to the average time between the start of consecutive write operations (τ). The definition may be represented by the following formula:

$$t_p = \frac{50.4 \times 10^{-4}}{\tau}.$$

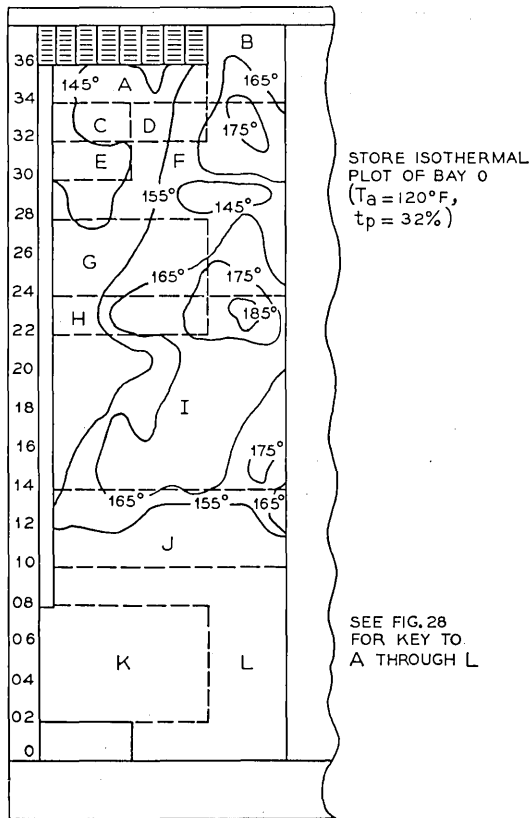Module thermal properties were established through a series of tests

Fig. 30—Frame isotherm plot.

on a store frame maintained at a constant ambient temperature in an environmental chamber. These tests established the relationship between write rate and module hot spot temperature as:

$$T_{max} = 0.78t_p + T_a + 13.$$

where T_a is the ambient temperature.

The internal temperature limit of 180°F restricts continuous store operation to a 60.2 percent write rate at the ambient temperature limit of 120°F. Knowledge of this write rate limitation is an important application parameter. It represents a relationship between the method of use and the component physical characteristics which always exists but is often unknown. Considerable temperature safety margins exist

since present SPC No. 1A applications will not exceed a 32 percent write rate.

10.3.3 *Bay 0 Circuit Packs*

The circuit packs mounted within bay 0 of the store frame dissipate approximately 770 watts. To determine component operating temperatures the store was placed in an environment whose temperature was near the maximum ambient and with air movement by natural convection only. The store was then exercised at the expected worst case write rate of 32 percent ($\tau = 157\mu$s). After the store reached thermal equilibrium, temperatures were measured to produce an isothermal plot. This plot, corrected to the maximum ambient of 120°F, is shown in Fig. 30.

The resulting maximum component environmental temperature was found to be 185°F for an ambient temperature of 120°F. No component was found to operate above its maximum temperature ratings. Since the thermal characterization was produced without forced air movement, the air velocity about the frame was due only to natural convection and represents a worse case condition. With some forced air motion, as now recommended for these equipments, additional temperature margins are provided.

XI. SUMMARY

The requirements of the SPC 1A store motivated the development of a new and highly sophisticated memory device. The complexity of this device, the PBT memory, led to an intensive test program in order to understand the theory and to characterize its performance. The results of this effort have produced a memory that is very well characterized and stringently tested.

The resulting store combines the nondestructive readout features of the No. 1 ESS program store and the electronic alterability of the call store. The preread write sequence along with the protected area feature has made accidental program destruction practically impossible. The electronic alterability makes office data and program changes an easy, quick procedure. Field experience has demonstrated data integrity over wide ranges of store adjustments.

REFERENCES

1. James, D. A., Myers, F. H., "An Electrically Alterable NDRO Memory System. Using a Twistor," IEEE Trans. Communication Technology, *COM-17* (April 1969), pp. 93–99.

2. Gerke, P. M., Harding, P. A., and Staehler, R. E., "No. 1 ESS Call Store—A 0.2-Megabit Ferrite Sheet Memory," B.S.T.J., *43* No. 5 (September 1964), pp. 2147–2191.
3. Ault, C. F., Gallaher, L. E., Greenwood, T. S., and Koehler, D. C., "No. 1 ESS Program Store," B.S.T.J. *43* No. 5 (September 1964), pp. 2097–2146.
4. Baker, W. A., "The Piggyback Twistor—An Electrically Alterable Nondestructive Readout Twistor Memory," IEEE Trans. on Comm. and Elec., *83* (November 1964), pp. 829–833.
5. Nesbitt, E. A., Chin, G. Y., and Jaffe, D., "New Low-Magnetostrictive Permanent-Magnet Alloys," J. Appl. Phys., *39* (February 1968), pp. 1268–1269.
6. Myers, F. H., "Using a Small Computer to Test a Complex Memory System," Proceedings of the National Electronics Conf., Chicago, Illinois, *25* (December 1969), pp. 711–715.
7. Ferguson, J. G., Grutzner, W. E., Koehler, D. C., Skinner, R. S., Skubiak, M. T., and Wetherell, D. H., "No. 1 ESS Apparatus and Equipment," B.S.T.J., *43* No. 5 (September 1964), p. 2355.
8. Baldinger, W. R., and Clark, G. T., "TSPS No. 1: Physical Design," B.S.T.J., this issue, pp. 2685–2709.

TSPS No. 1:

Peripheral Circuits

By W. K. COMELLA, C. M. DAY, JR., and J. A. HACKETT

(Manuscript received August 18, 1970)

The TSPS No. 1 peripheral circuits permit the data processing ability of the Stored Program Control No. 1A to be applied to TSPS functions. These circuits include scanners, signal distributors, networks, trunk and service circuits, operator positions and the position subsystem. A description of each circuit is given, followed by a description of the features provided by the associated maintenance program.

I. INTRODUCTION

Some of the TSPS peripheral circuits are new or have novel features worthy of description while others are very similar to their No. 1 ESS counterparts and need no further description. Thus scanners, signal distributors, the AMA frame, and TTY buffers which have been described previously[1] will not be treated in detail here. The communication bus, networks, trunk and service circuits, local and remote position subsystems, and novel features of various maintenance programs are the essential subjects of this article.

The communication bus joining all units to the Stored Program Control (SPC) is described in Section II.

A network consisting of trunk link network (TLN) and position link network (PLN) frames is used to establish connections between the trunk circuits and the operator positions. The network fabric is new and described later in Section III, although the implementation and control circuitry is similar to that used in the No. 1 ESS network.

In addition to operator positions and trunks, numerous service circuits such as MF receivers and outpulsers also appear on the network. These circuits and their associated maintenance programs are described in Section IV.

Displays for all TSPS equipment and test facilities for all trunks appearing on the network are provided in the Control Display and Test (CDT) frame as described in Section V.

The group of circuits which links the communication bus (data) and position link network (voice) with the operator's console is called the "position subsystem." This subsystem is arranged to work with operators in the same building as the SPC (local) or many miles distant (remote). These circuits, which do not have ESS counterparts, are arranged in a series chain which leads to a maintenance program strategy different than that for other peripheral units. The position subsystem and the associated maintenance programs are described in Sections VI and VII.

The maintenance programs for the TSPS peripheral units operate in the same interrupt structure and under the same executive and maintenance control program as do the SPC generic maintenance programs, which are described in the article, "Stored Program Control No. 1A" of this issue.[2] Thus much of the material of that article is prerequisite to an understanding of the organization, strategy and methods used by the TSPS maintenance programs.

II. TSPS COMMUNICATION BUSES

Communications with units in the TSPS No. 1 is accomplished through groups of paired wires called buses (Fig. 1).

Some of the translation necessary to decode orders from the SPC is implemented on a common basis for a number of peripheral units in the Communications Bus Translator (CBT). The CBT receives 20 bits of data, plus parity and various control signals from the SPC. The CBT translates the information to a code consisting of several groups of one out of N and broadcasts the translated data and the original binary data to the TSPS peripheral units. The use of a one out of N bus and a common translator eliminates the need for a translator in each peripheral unit and allows the peripheral unit to make a simple error check of the data received.

Answer information is transmitted from the peripheral circuits to the SPC via the scanner answer bus (SAB). Enable information transmitted from the SPC is decoded by the Central Pulse Distributor and results in a pulse over a private pair which selects the particular unit which should receive a particular order from the common address bus, and reply over the common answer bus.

The maximum cable length for the CBT address bus is 450 feet. The maximum number of series connected cable pick-off transformers is 50. The bus circuitry and apparatus are similar to that of ESS No. 1[3]; however, both a one out of N and a binary bus are provided, and fanout circuitry is not used as these buses may run in two directions.
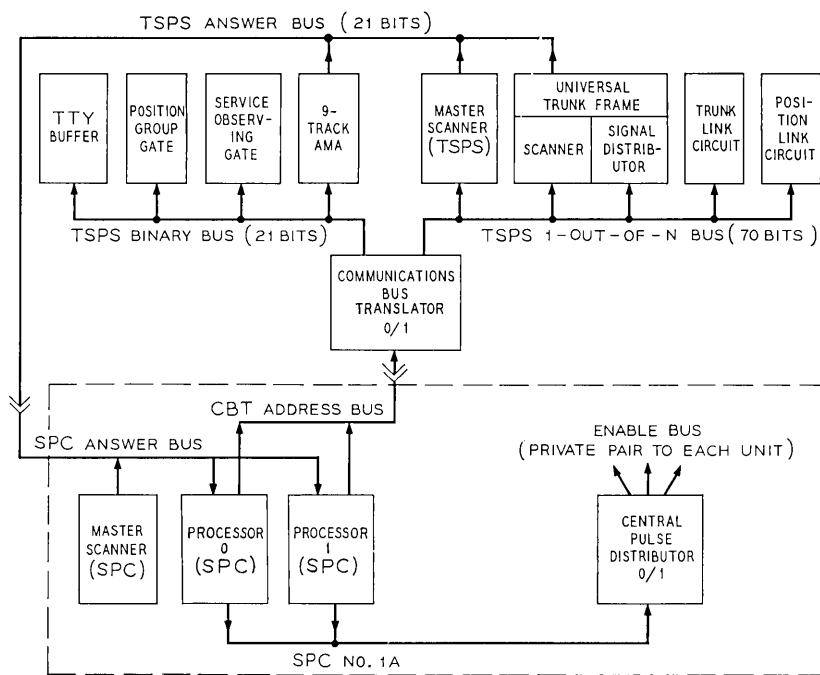
Fig. 1—TSPS No. 1 communications buses.

2.1 *Connecting Units*

The TSPS Master Scanners (MS), Universal Trunk Frame Scanners (UTSC), Universal Trunk Frame Signal Distributors (UTSD) and Networks receive 1 out of N information from the CBT. The AMAs, Position Group Gates, TTY Buffers and Service Observing Gate (SOG) receive binary information. The CBT 1/N translation lead assignments and functions for the connecting units are shown in Figs. 2 and 3.

2.1.2 *Use of SPC Maintenance Programs*

The SPC generic maintenance programs are extended by means of TSPS application programs to serve some of these TSPS peripheral units. For example, the network controllers function in a manner practically identical to signal distributors, from the viewpoint of the SPC. Therefore the SPC peripheral unit fault recognition program, PUFR, which serves the SPC signal distributor, also serves the TSPS signal distributors and the TSPS network controllers. The SPC scanner

BINARY DATA INPUT BIT POSITION				TRANSLATED DATA OUTPUT 1/N DATA		CONNECTING UNIT			
19 18 17 16 15	14 13 12 11 10	9 8 7 6 5	4 3 2 1 0	LEAD	1/N	TLN	PLN	MSC OR UTFSC	UTFSD
			0 0 0	00					
			0 0 1	01					
			0 1 0	02					
			0 1 1	03	$\frac{1}{8}$	LEVEL		LS	
			1 0 0	04					
			1 0 1	05					
			1 1 0	06					
			1 1 1	07					
		0 0		08					
		0 1		09	$\frac{1}{4}$	TSBO	LEVEL		
		1 0		10					
		1 1		11					
		0 0		12					
		0 1		13	$\frac{1}{4}$	SWITCH	SWITCH		
		1 0		14					
		1 1		15					
		0		16	$\frac{1}{2}$			MT	O/R
		1		17					
	0 0			18					
	0 1			19	$\frac{1}{4}$	GRID	GRID		RELAY POINT
	1 0			20					
	1 1			21					
	0 0 0			22					
	0 0 1			23					
	0 1 0			24					
	0 1 1			25	$\frac{1}{8}$	JUNCTOR	JUNCTOR		
	1 0 0			26					
	1 0 1			27					
	1 1 0			28					
	1 1 1			29					
	0 0			30					
	0 1			31	$\frac{1}{4}$	CHANNEL	CHANNEL		
	1 0			32					
	1 1			33					
0				34	$\frac{1}{2}$				
1				35					
0				36	$\frac{1}{2}$	JSBO	JSBO		
1				37					
0 0				38					
0 1				39	$\frac{1}{4}$	ORDER	ORDER	BAY	
1 0				40					
1 1				41					
0				42	$\frac{1}{2}$	MATE	MATE		TEST
1				43					
	0 0 0 0			48					
	0 0 1 0			49					
	0 1 0 0			50					
	0 1 1 0			51	$\frac{1}{8}$			CIRCUIT 0	Y
	1 0 0 0			52					
	1 0 1 0			53					
	1 1 0 0			54					
	1 1 1 0			55					
0 0	0			56					
0 0	0			57					
0 1	0			58					
0 1	0			59	$\frac{1}{8}$			CIRCUIT 0	X
1 0	0			60					
1 0	0			61					
1 1	0			62					
1 1	0			63					
	0 0 0 0			56					
	0 0 1 1			57					
	0 1 0 0			58					
	0 1 1 1			59	$\frac{1}{8}$			CIRCUIT 1	X
	1 0 0 0			60					
	1 0 1 1			61					
	1 1 0 0			62					
	1 1 1 1			63					
0 0	1			48					
0 0	1			49					
0 1	1			50					
0 1	1			51	$\frac{1}{8}$			CIRCUIT 1	Y
1 0	1			52					
1 0	1			53					
1 1	1			54					
1 1	1			55					
	0			70	$\frac{1}{2}$				CIRCUIT
	1			71					
		0 0	0 0	72					
		0 0	0 1	73					
		0 0	1 0	74					
		0 0	1 1	75	$\frac{1}{8}$			MS	
		1 1	0 0	44					
		1 1	0 1	45					
		1 1	1 0	46					
		1 1	1 1	47					

Fig. 2—CBT translation, lead assignments and functions for connecting units.
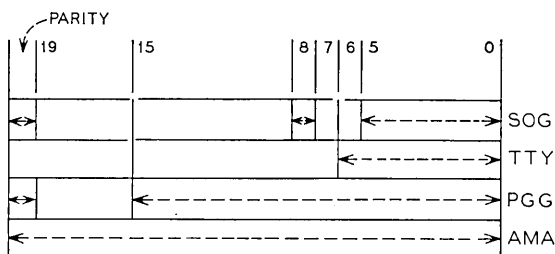
Fig. 3—CBT binary bus lead assignments.

fault recognition program, SCFR, also serves the TSPS scanners. The SPC scanner and signal distributor diagnostics are extended by means of short TSPS application programs to also serve the TSPS units. Thus a description for some of the maintenance program for the TSPS peripheral units is covered in the article "SPC No. 1A" of this issue.

2.2 *False Code Facility*

Invalid 1 out of N or bad parity may be broadcast to the TSPS peripheral units by maintenance programs. To achieve this, a false code register within the CBT is addressed by the program and loaded with a code indicating the modifications desired in the output data from the CBT. The first order distributed through the CBT subsequent to this will be modified accordingly. The false code register then automatically resets and the CBT reverts to normal operation. A We Really Mean It (WRMI) pulse is required to set the false code register to insure that a false code is not inadvertently set up by noise conditions.

2.3 *Interrupts Caused by the CBT*

When data is distributed through the CBT, a parity check is performed by the CBT circuit. A successful parity check in conjunction with proper receipt of timing signals from the SPC causes an All Seems Well (ASW) pulse to be returned to the SPC. Failure to receive an ASW from the CBT will result in an F-level interrupt, and a CBT diagnostic request by the F-level program.

2.4 *CBT Diagnostic Program*

The CBT diagnostic checks the ability of the CBT to receive data from the SPC by using two test ferrods, which report an all ones condition in the buffer register and false code register respectively.

Diagnostics of the connecting units have been structured in such a way that the tests which check the ability of these units to receive data from the CBT are run as subroutines by the CBT diagnostic. A connecting unit which is indicated as being good is selected by the CBT diagnostic and the subroutine for that unit is called to test the ability of the CBT to transmit data on the leads which connect to that unit. This process is repeated with successive unit types until all output leads have been tested. Referring to Figs. 2 and 3, it will be seen that the AMA serves to check all leads of the binary bus, and the PLN, UTSC, and UTSD serve to check all leads of the 1/N bus.

2.5 CBT Faults Detected by Connecting Units

The performance of the CBT translation and output circuitry is monitored through responses of the connecting units. If the fault recognition program for a connecting unit detects possible address data problems, a diagnostic is requested both on the particular unit involved and on the CBT. Both the peripheral unit and the CBT are marked in trouble.

As the CBT is assigned a lower unit type number than all connecting units, the CBT diagnostic is run first and updates the CBT trouble flag according to the result. The connecting unit diagnostic then runs avoiding the use of the CBT marked in trouble. Thus of the two diagnostic requests one will pass and one will fail, resolving the trouble either to the CBT or to the peripheral unit.

This process takes some time, however, and in the case of a CBT trouble, a number of peripheral units may be affected by the fault. This condition is guarded against by the fault recognition programs for the connecting units. The typical strategy is to retry the order over the original bus. If this retry also fails, the order is sent to the same unit over the other bus. If that order succeeds, bus trouble is assumed and the enables of all units are updated to change the bus choice. The assumption of bus trouble is not necessarily correct as the fault may not lie in the bus, but in the connecting circuitry of the peripheral unit. However, the bus switch (enable update) is requested to avoid propagation of possible bus troubles.

In the case of scanners, failing orders may be tried with a second scanner unit to positively isolate bus trouble. This strategy cannot be used with other unit types as it would result in an unwanted relay or network operation. A short CBT fault recognition routine is provided by the TSPS application program and is run immediately prior

to the generic scanner fault recognition program to isolate bus troubles affecting scanners.

The TTY is not monitored by a fault recognition program but by the craftsman who observes improper printing and manually requests a diagnostic. The TTYs are routinely checked by an exercise which requests a daily diagnostic on each TTY.

The authors acknowledge the contributions of W. R. Serence, and P. J. Cuffaro to the AMA maintenance programming, P. J. Brendel to the AMA data transfer and bus fault recognition programs, and R. J. Greylock and R. Riley whose efforts permitted the extension of SPC programs to TSPS.

III. NETWORK

3.1 *Network Fabric*

The basic function of the network is to connect trunk circuits to position circuits, receivers, outpulsers and other service circuits. Ferreed switches are used as network elements in a fabric designed to meet the particular needs of TSPS, namely: ability to handle large or small installations, serve trunks carrying light or heavy traffic, and permit addition of equipment to grow with a minimum of installer effort.

The network uses 4 stages of switching to provide a 3 link connection from input (trunks) to output (positions and service circuits). Eight by eight ferreed switches are used in the first three stages; 8×4 switches are used in the final stage to provide a $2:1$ concentration. The switches are arranged in grids, the first two stages being mounted on the trunk link frame, the last two on the position link frame. The ferreed switches and wiring for one grid are shown in Fig. 5 which,
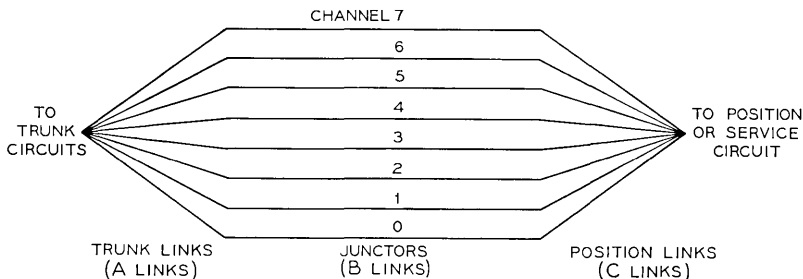
Fig. 4—Eight-channel pattern.
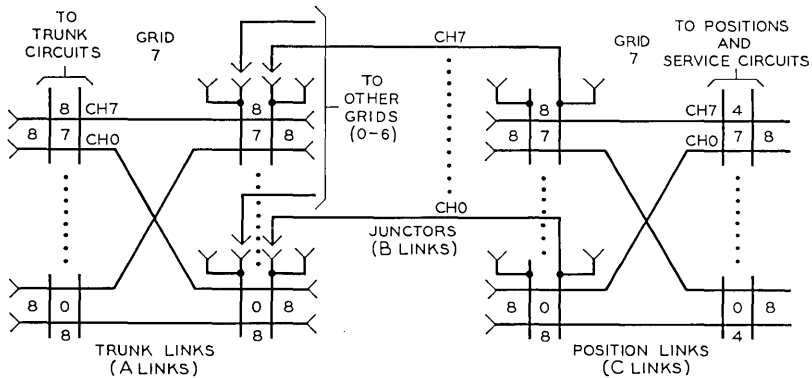
Fig. 5—One grid (7) illustrating the eight-channel pattern.

along with Fig. 4, illustrates the eight possible channels which could be used to connect a given input to a given output. A connection is made up of an A link which interconnects the switches of a grid on the trunk link frame, a B link or junctor which connects a trunk link grid to a position link grid, and a C link which interconnects the switches of a grid on the position link frame. Eight grids are mounted on a frame and are wired as shown in Fig. 6. Thus, one network provides 512 input appearances with full access to 256 outputs.

Positions are engineered for a maximum occupancy of 91.5 percent as determined by requirements on operator work load, and service circuits for 70 percent as determined by the group size. Positions have two appearances on the network; service circuits have a single appearance. If an 8 × 4 switch has two positions and two service circuits on the outputs, the above occupancies result in a link occupancy of 29 percent;

$$\frac{91.5 + 2(70)}{8} = 29\%$$

or a traffic intensity of 148 erlangs (0.29 × 512) at the input of a network. At this link occupancy in order to meet the network blocking objective of 0.001 it is necessary to provide at least two retrials on connections to positions (both appearances are examined on each trial) and at least 3 retrials on connections to service circuits. The call processing program provides the necessary retrials and queuing. Retrials are accomplished by choosing another idle service circuit or position. If the call is blocked after the above retrials, it is queued
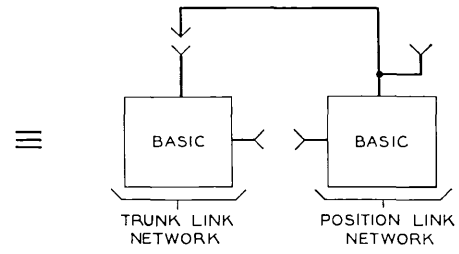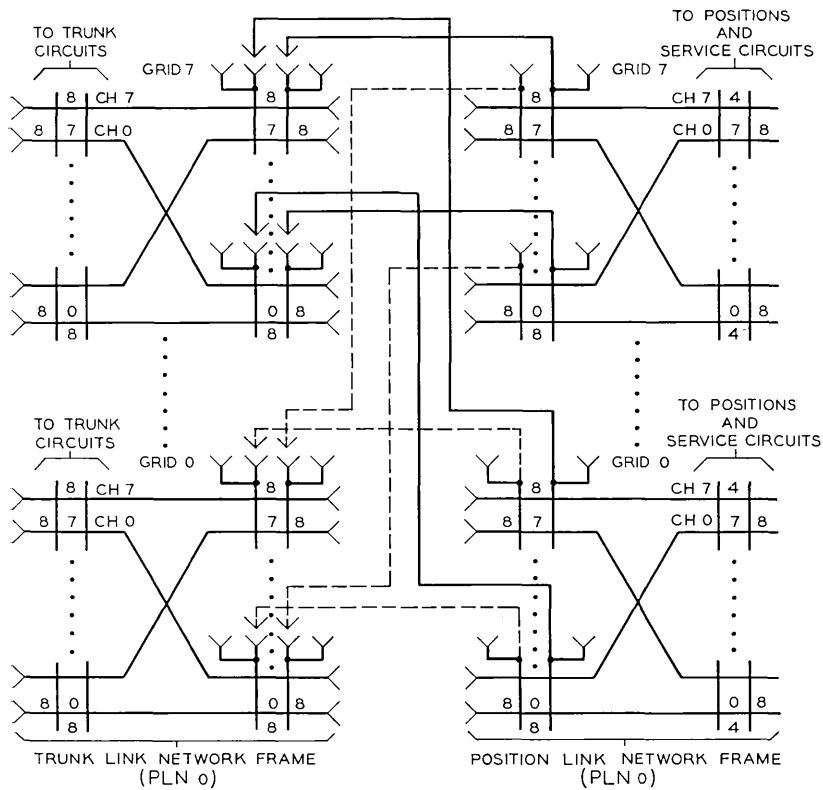
Fig. 6—One network, comprised of eight grids, illustrating junctor wiring pattern and symbolic representation.

and a new attempt is made every half second. On each attempt retrials are made as described above.

If a traffic intensity of 148 erlangs is not reached with 512 inputs to a network, trunk switch buildouts may be connected into the trunk side of the network to increase the linkage load by increasing the number of trunks which have access to 8 A links, as shown in Fig. 7. Thus, additional trunk groups carrying lighter densities of traffic may be accommodated. Concentration at the input to the network may be varied from 1:1 to 4:1 in unit steps by appropriate usage of trunk switch buildouts.

Connectors are provided to minimize installer effort and provide for future growth as shown in Fig. 6. The B links or junctors are hard wired to the position link at the factory, and are distributed across
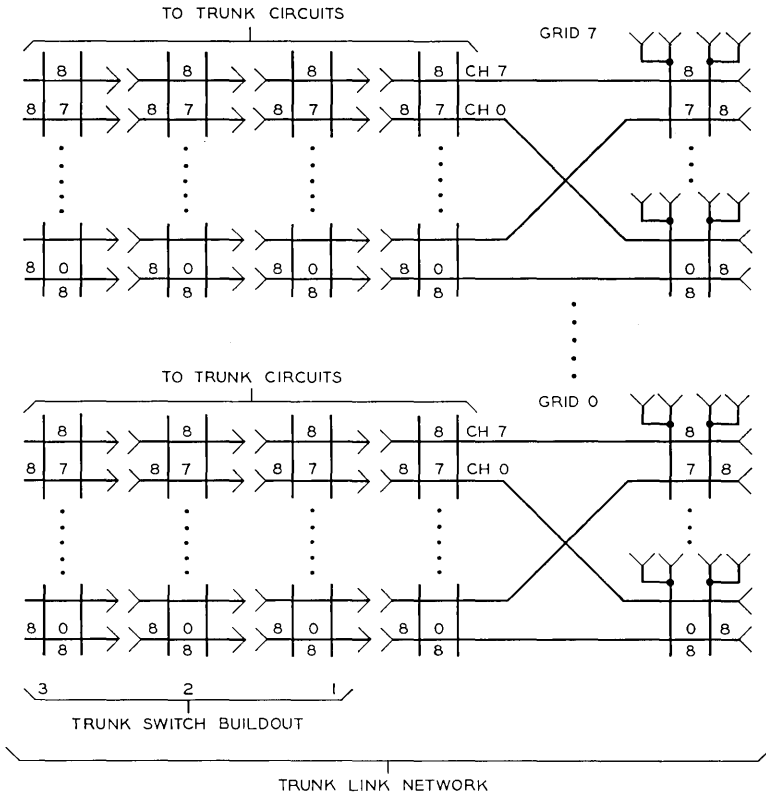
Fig. 7—Addition of trunk switch build-outs to trunk link network.
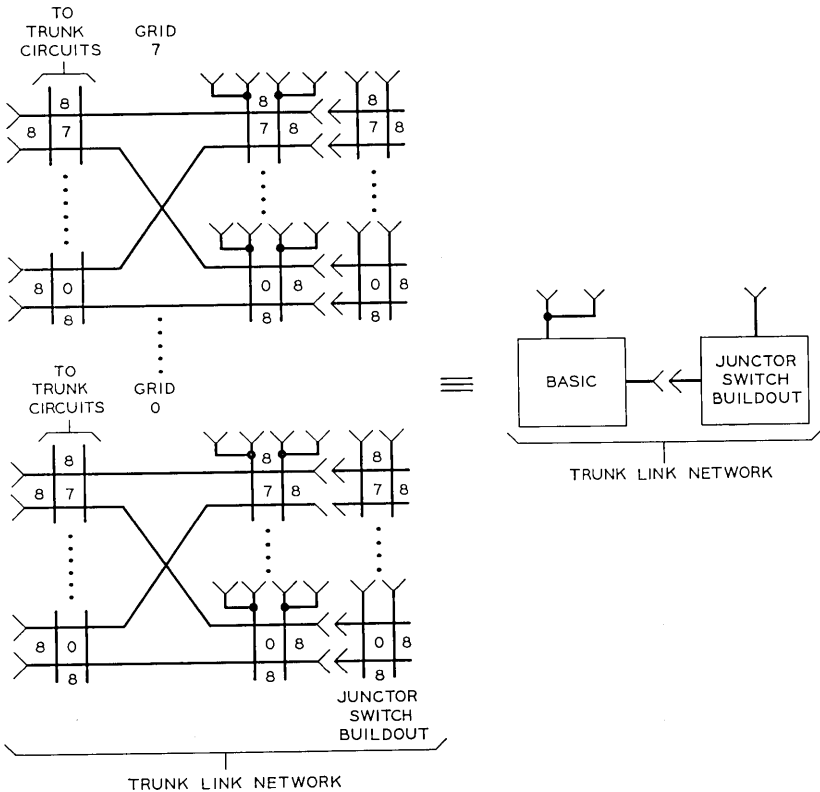
Fig. 8—Addition of junctor switch build-out to trunk link network and symbolic representation.

all grids in the position link to provide full access. At the job site the connector ended cables are simply plugged into the appropriate trunk link grids. Connectors are provided for plugging additional junctor cables as the network grows.

A junctor switch buildout is connected into the basic trunk link frame to provide access to additional position link networks as indicated in Fig. 8. The two network configuration is shown in Fig. 9, and is the smallest size recommended. It uses the junctor switch buildout on the trunk link frame, as previously described, and multiplying of junctors via connectors at the trunk link to provide full access.

A three network configuration requires a junctor switch buildout on the position link frame as does the four network configuration illustrated in Fig. 10. As the network configuration grows, existing
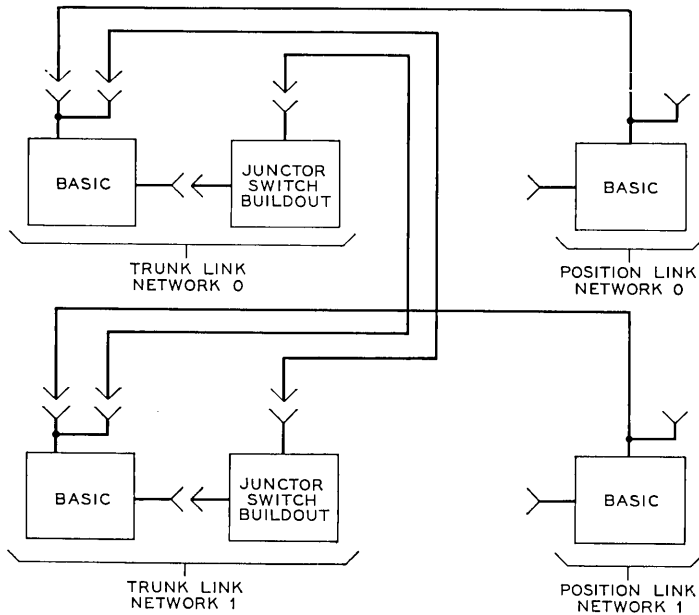
Fig. 9—Two-network configuration.

junctors need not be rearranged. The added buildouts and junctor cables are connectorized for ease of installation.

3.1.1 *Path Hunt Program*

This network organization facilitates a simple path hunt program. As networks are added to the system, an area of memory known as a link map is also added. As shown in Fig. 11, this table is organized in such a manner that when it is desired to connect a trunk to a position, portions of the trunk terminal number (TTN) and position terminal number (PTN) are used by the path hunt program as indices into the link map table to obtain the addresses of three words containing busy-idle bits. These busy-idle bits correspond to the eight sets of A, B, and C links which may be used to connect a given trunk to a given position. The first word contains eight bits corresponding to the eight A links, channel 0 to channel 7. The bits are set to 1 to indicate an idle condition on the corresponding link, and to 0 to indicate a busy condition. Thus, to determine an available path it is merely necessary to form the logical product of these three words. Idle channels will be indicated by 1's in the result. In the example
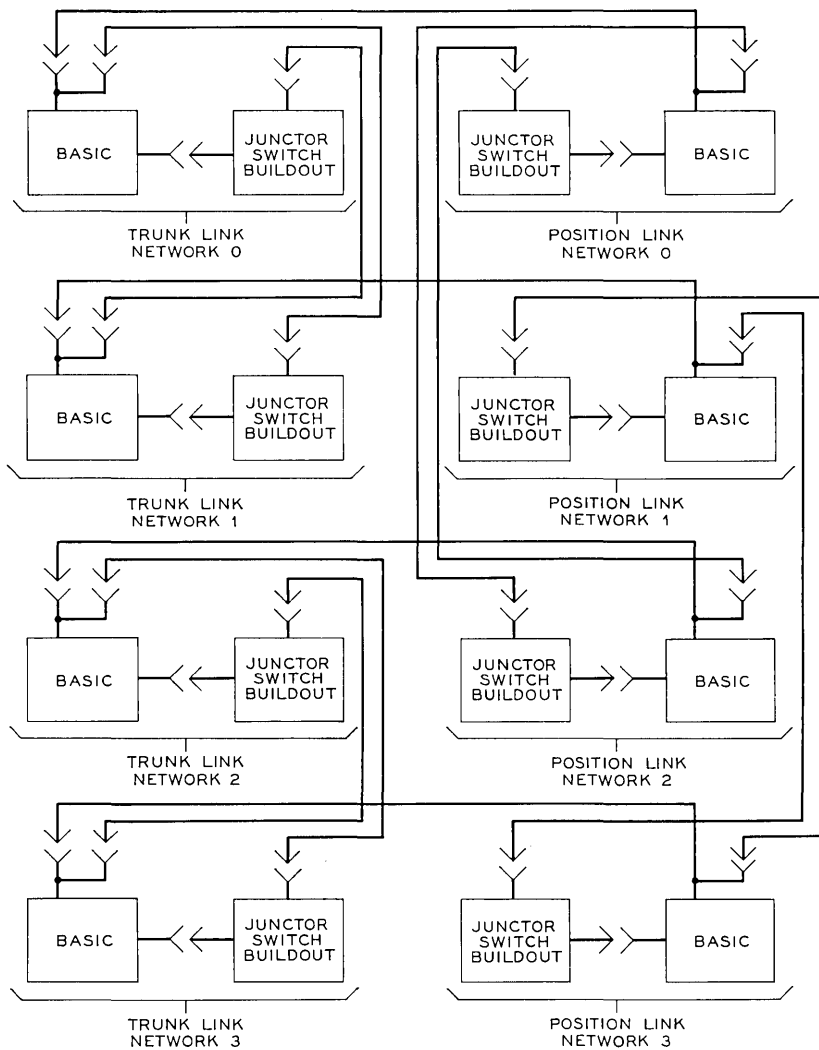
Fig. 10—Four-network configuration illustrating junctor switch build-outs (JSBO) on both trunk and position link networks.

below this procedure indicates that channels 1 and 5 are idle.

Channel:	7	6	5	4	3	2	1	0
A links	1	0	1	1	0	0	1	0
B links	0	1	1	1	0	1	1	1
C links	1	0	1	0	0	1	1	0
Product	0	0	1	0	0	0	1	0
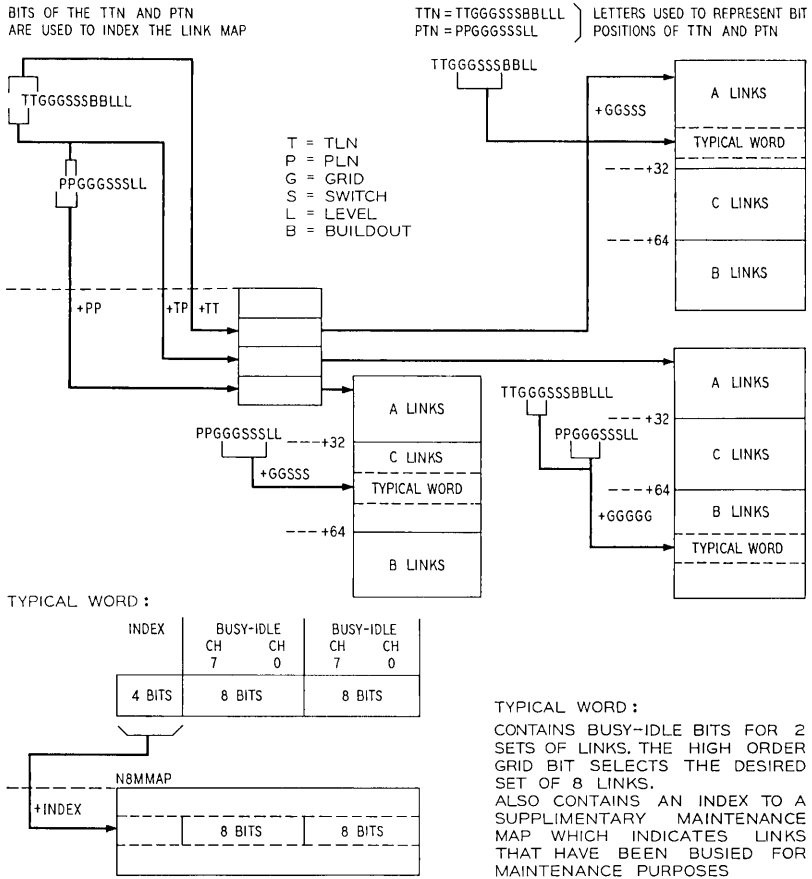
Fig. 11—Link map.

The channel which is finally used is selected from the available idle channels. The low order bits of the system clock are used to change the first choice channel for each selection to spread usage over the equipment.

3.1.2 Fabric Maintenance

The integrity of the fabric is monitored by continuity checks made via ferrods in most service circuits and by operator trouble reports. Continuity check failures or operator trouble reports keyed at the positions result in TTY printouts of the links used on the failing con-

nections. The TTY information may be analyzed to isolate possible network troubles, or troubles external to the TSPS office, either in the trunks or in connecting offices. In the case of continuity check failures involving service circuits, the service circuit is automatically diagnosed to check one possible cause of the failure.

Portions of the network fabric may be made busy for maintenance purposes by a teletype request which initiates a program that sets zeroes in the busy-idle bits for the desired links. As two 8-bit link groups are packed within one 20-bit word of the link map table, the remaining 4 bits may be used to set an index to a supplementary table which indicates to the audit programs exactly which of the links are maintenance busy as shown in Fig. 11.

After calls have been blocked from the links by making them maintenance busy, test connections may be established on these links by means of a TTY request. Suspected hardware faults may then be verified and cleared. A TTY request is available to repeatedly connect and disconnect a test path at 200-millisecond intervals for dynamic testing. If an active switch must be replaced, it may be made busy for the replacement interval by a similar method. It is also possible to busy out a single faulty link for a longer period of time to prevent call failures if the repair cannot be made immediately.

3.2 Network Controllers

The ferreed switches are operated by controllers which are similar to those used in No. 1 ESS* described in the September, 1964, B.S.T.J.; therefore, only a general description and new features unique to the TSPS controller will be given here.

The TSPS controller differs from the ESS controller in the maintenance features provided, and in that a separate disconnect order rather than the destructive mark of a subsequent connect order is used to release the ferreed switches.

3.2.1 General Description

The controllers, shown in Fig. 12, receive orders in 1/N form from the CBT and operate appropriate path selection relays to set up a pulsing path in the ferreed switches. When the pulse path is complete, as indicated by continuity over that path, a high current pulser fires to effect a connect or disconnect of the ferreed switches, and causes

---

* B.S.T.J., September 1964, *43*, p. 2221, No. 1 ESS Switching Network Frames and Circuits, D. Danielson, K. S. Dunlap and H. R. Hofmann.
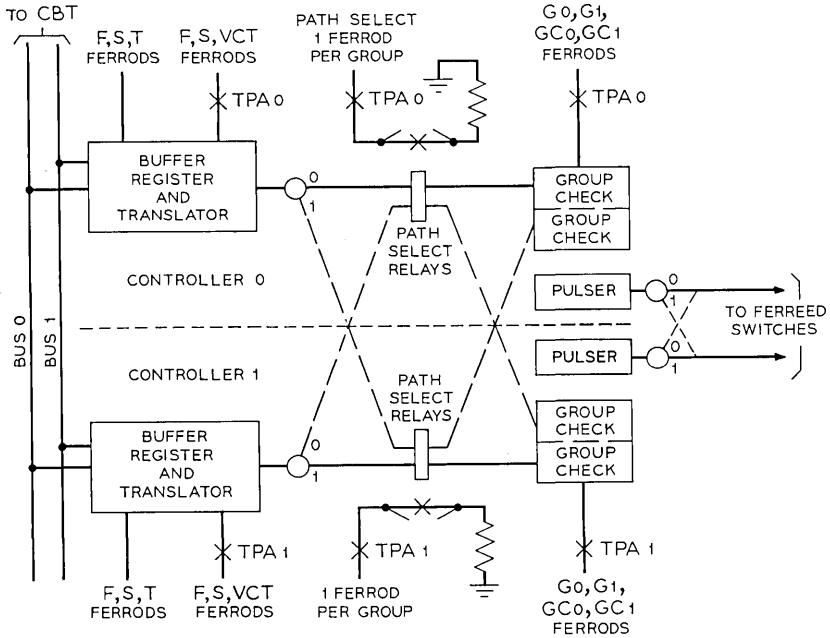
Fig. 12—Network controllers.

a successful completion of these operations to be indicated to the SPC via ferrods designated F, S, and T. Two controllers are used per frame. Controller 0 normally controls wire spring relays which set up pulse paths in grids 0–3; controller 1 normally serves similar relays in grids 4–7. These wire spring relays are a double wound type with one coil connected to each controller. Thus, it is possible for either controller to operate in any of the eight grids should its mate fail.

The path select relays are arranged in various groups. To effect a network connection, one and only one relay in each group is operated. A group check circuit senses the initial current drain within each of these groups to determine if the 1/N condition is met and, if it is not, causes battery to be removed from the path selection relays preventing their operation. It is necessary to prevent the relays from operating under this condition to avoid firing the pulser over an improper path, as the pulser is triggered when continuity is presented to its output leads.

Additional controller points may be cut through via the TPA relay
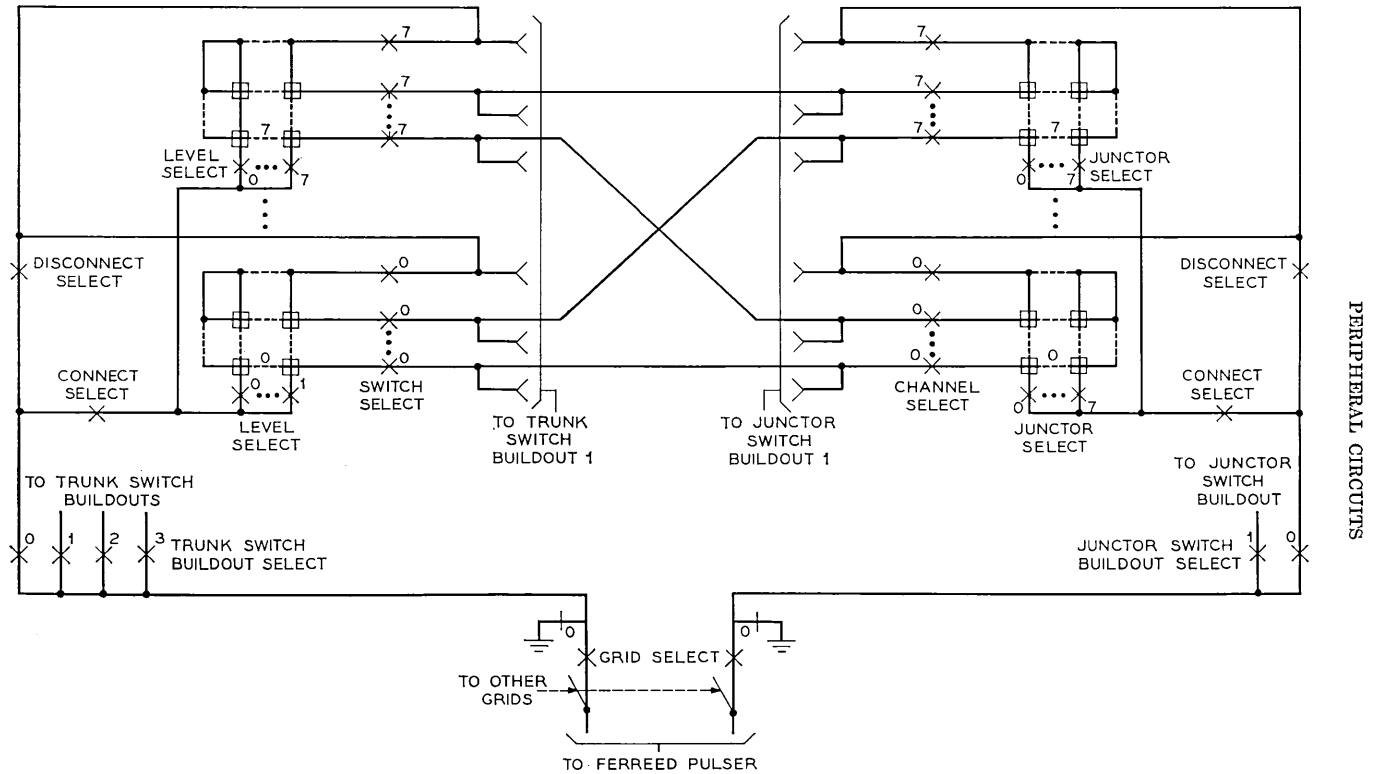
Fig. 13—Pulse path for one grid of trunk link network.

to a diagnostic bus consisting of a group of ferrods shared by all controllers of the same type within the office.

The pulse path selection for one grid is shown in Fig. 13. Path selection relays are operated to select the grid, trunk switch buildout, connect/disconnect, level, switch, channel, junctor and junctor switch buildout. Connectors are provided to plug in buildouts. The additional pulse path selection circuitry for one grid of a trunk switch buildout which would be plugged is as shown in Fig. 14.

As it is necessary to insure disconnection in one buildout before establishing a connection in a second to avoid bridging the talking paths, separate orders for connect and disconnect are required. Thus, the advantages of buildout frames introduces the penalty of a separate disconnect order.

3.2.2 *New Maintenance Features*

The integrity of the pulse path is monitored by the ferreed pulser which detects opens or short-to-ground conditions. An innovation
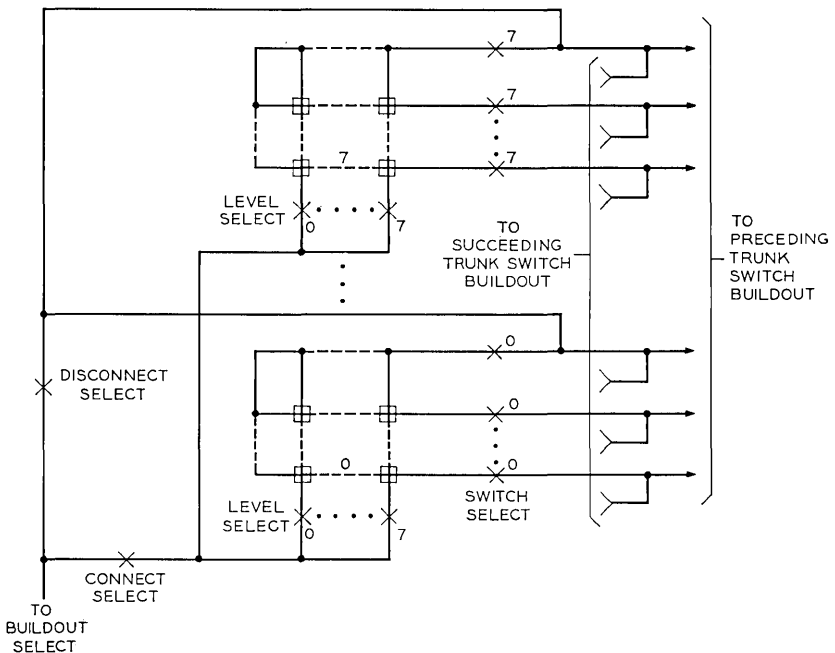
Fig. 14—Trunk switch build-out for one grid.

found in the TSPS controller is the use of transfer contacts on the grid select relays to hold ground on pulse paths in grids other than that grid in which a connection is being established. With this feature, shorts between pulsing paths can be picked up by the short-to-ground detector.

Also new is the indication of short-to-ground conditions via a ferrod to notify the fault recognition program of the short condition while the network order is still available. Network orders resulting in short-to-ground or open pulse path indications are printed on the TTY and may be analyzed to determine the probable location of the fault.

The TSPS network controllers also provide additional outputs from the group check circuits to increase diagnostic resolution.

One path selection ferrod is provided per group and is wired to a make contact from each relay within the group to provide an indication that at least one relay has operated. The steering circuit at the output of the high current pulser allows the use of test orders which operate path selection relays but do not fire the high current pulser. The path selection relays lock up and may be checked via the path selection ferrods for a positive indication of armature movement.

A TTY request may be used to input a test order to be repeated to the frame at 200-ms intervals. This feature may be used to check relay operation and controller functions without pulsing into the fabric which might cause cutoffs or possibly harm the ferreed switches if certain faults are present.

As previously mentioned, under normal operation battery is removed from the path selection relays before they have operated in case of a group check failure. The TSPS controller overrides this check on diagnostic test orders, allowing the path selection relays to operate and lock up, providing specific information on relays which fail to operate.

The authors acknowledge the contributions of the late E. L. Erwin who conceived the network fabric arrangement, L. J. Murphy, who implemented the circuit, and S. Lederman who designed the software control.

IV. CIRCUITS WHICH APPEAR ON THE NETWORK

4.1 *Trunk Group Number*

Every position, service circuit, and trunk appearing on the TSPS network is a member of a group of similar circuits. Each group is

assigned a trunk group number (TGN), and each member of the
group is assigned a member number (MEMN) sequentially. Table I
lists the trunk group numbers and their functional assignment. The
TGN and MEMN of a position is identical with the chief operator
group number and member number. Hence, TGNs 0–8 are for the 9
groups of positions. TGNs 9–30 are assigned to the various service cir-
cuits by their function, with 24–30 unassigned. Thus, each network
appearance on the position link corresponds to a member of a group
with a TGN between zero and thirty.

TGNs 31–511 are assigned to groups of trunks with network ap-
pearances on the trunk link. TGNs 31–39 are unassigned; 40–55
identify groups of service trunks, CAMA trunks, delay call trunks,
and test trunks. A TGN of 57 or more identifies a universal trunk
group which carries TSPS traffic from the local office to the toll office.
Universal trunks are grouped by their origination and function in

TABLE I—TGN ASSIGNMENTS

TGN	Group Function	CRI	Register Type	Link Appear.
0–8	Chief operator groups	0	Sr	
9	Service observing	6	Jr	
10	Coin control and ringback	5	Sr	
11	Audible tone	4	Jr	
12	MF receiver	1	Jr	
13	DP receiver	2	Jr	PLN
14	MF outpulser	3	Jr	
15	Idle line termination	13	Jr	
16–21	Reorder tone and announcement	7–12	Jr	
22	Master test line (TMTL)	23	Sr	
23	Access line zero	24	Sr	
24–30	Unassigned			
31–39	Unassigned			
40	Service trunk (information)	14	Jr	
41	Service trunk (rate & route)	15	Jr	
42–43	Service trunk (unassigned)	16–17	Jr	
44	MFR test	25	Sr	
45	DPR test	26	Sr	
46	OP test	27	Sr	TLN
47	Master test line (PMTL)	28	Sr	
48	Access line one	29	Sr	
49–53	CAMA trunks	30	Sr	
54–55	Delayed call trunks	18	Sr	
56	System monitor circuit	31	Sr	
57–511	Universal trunks	31	Sr	

Note: The Circuit Register Identifier (CRI) is used by many programs in a manner
similar to the use of the TGN. The administrative registers of each circuit are of two
kinds: senior and junior.

the usual manner (e.g., Dover, N.J., to Morristown, N.J., coin) and each of these groups is then assigned a TGN.

4.2 *Trunks and Service Circuits*

The local-toll TSPS trunks can be placed in six categories since two types of pulsing (dial pulse [DP] and multifrequency [MF]), two types of signaling (loop and E&M) and two types of transmission (2-wire and 4-wire) are provided. The six types, rather than eight, obtain because all 4-wire trunks use only E&M signaling to the local office. At present all TSPS trunks use loop signaling toward the toll office and must be located near enough to the toll office to insure not exceeding a 2-dB loss. It is expected that use of carrier facilities between TSPS and the toll facility will be provided in the near future. A typical trunk is shown schematically in Fig. 15.

4.2.1 *Dial Pulse Trunks*

Handling Dial Pulse Signaling posed the classic problem of how to cope with fast dialing which occurs before a DP receiver is attached. A stored program system aggravates this problem since not only must time be taken to obtain a link connection to an idle receiver, but also the scanning interval time must be considered.
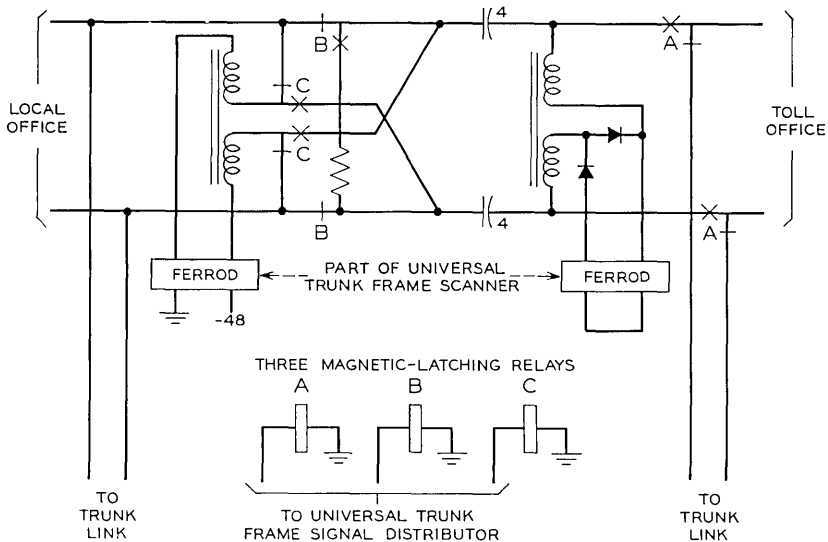
Fig. 15—Simplified schematic—TSPS MF trunk.

The unique solution employed in TSPS provides a two-pulse detector in each DP trunk circuit. That is, the first two pulses of any dialed digit can be recognized and stored by the trunk whether or not a DP receiver is attached. Thus, if a customer dialing from a step-by-step local office manages to start the next digit (after the initial 0 or 1) before a link connection is established to an idle receiver, the call is not necessarily lost. When the receiver is attached, it notes the state of the two-pulse detector (0, 1, or 2 pulses) by a voltage state sent over the ring lead and follows subsequent pulsing over the tip lead. If 0 or 1 pulse is found, then the receiver connection is considered to have been made in time and the call is handled normally. If a count of two is noted, then a ferrod output of the receiver is energized to alert the SPC that receiver connection occurred too late. The call is then aborted and connected to reorder tone. Laboratory testing, simulation, and some limited field experience indicate that the probability of 2 (or more) pulses before a receiver connection is very low, and there should be no need to expand the feature to a three or four pulse counter.

The two-pulse detector is realized by use of three dry reed relays which are part of a six-relay pack intended for two TSPS trunks. The remainder of the trunks consists of a transmission path, three magnetic latching relays, two ferrod scan points and a pulsing relay. The magnetic latching relays define the different talking and signaling states of the trunk under SPC control, while the pulsing relay follows dial pulsing from the local office and repeats the pulses for use by the DP receiver. The two scan points are in a supervisory scan field and provide supervision toward the local office and toward the toll office.

4.2.2 MF Trunks

The MF trunks resemble the DP trunks in appearance and are simpler, since a pulsing relay and two-pulse counter are not required. The three magnetic latching relays provide eight possible states, six of which (including the idle state) are used at present.

The delayed call trunk is essentially an MF trunk which has both its ends connected to the toll office. This allows an operator to originate calls from her console and reach both the calling and called parties.

Operation of the magnetic latching relays and scanning of the supervisory ferrods is handled the same as in No. 1 ESS which has previously been described.

4.2.3 *Service Circuits*

The circuits terminated on the position link frame which can be connected to trunks as needed are called service circuits. These include digit receivers, digit outpulsers, coin control circuits and announcement trunks. Maintenance of these circuits is aided by use of service circuit test circuits connected to the trunk link frame and discussed in Section 4.4.

4.2.3.1 *Dial Pulse Receivers.* The dial pulse receiver receives dial pulses over the tip lead and counts complete digits using conventional relay circuitry. Upon seizure the receiver notes the state of the two-pulse counter in the trunk via a voltage on the ring lead. Its counting circuitry is advanced by one if a "1" is noted on the ring lead. A "reorder" ferrod is energized if a "2" is noted on the ring lead since this could mean two or more pulses were received by the trunk before the receiver was connected. Output from the DP receiver is in BCD code on four ferrods and a "signal present" ferrod is energized each time an interdigital interval is recognized. Thus, scanning of all but the "signal present" ferrod can be a directed scan and the SPC need only deal with whole digits, not the individual pulses of a digit.

4.2.3.2 *MF Receiver.* The major portion of the MF receiver is similar to conventional MF receivers and delivers only whole digits to the SPC. However, in addition to the 10 decimal digits, several "start" signals can be received. Both start pulses and digits are received as frequency pairs in 2 out of 6 combinations. This 2-out-of-6 (2/6) reception is passed to the SPC ferrod scan points. The variety of "start" pulses (pairs of tones) are used as a means for the local office to describe the type of call (coin, noncoin, dial "0" or 0+) to TSPS, thus permitting several types of calls to be handled by a single group of trunks. The SPC translates the 2/6 code to BCD (in the case of a digit) for temporary storage in the software trunk register.

While the MF receiver is primarily used to receive digits from a local office, it can also be connected to the toll office side of the TSPS trunk. This is done when a call which could not be handled by TSPS (mobile radio, marine, etc.) has been passed forward through a toll office connection to a cord switchboard operator. In this case when the cord switchboard operator must signal back to the calling customer (possibly to collect or return a coin) inband tones are generated at the operator location and sent to TSPS. An alerting pulse (reversal) permits

TSPS time to attach an MF receiver which then receives the inband tones. TSPS then sends appropriate signals to the local office.

In the reverse direction there is the further problem of passing a disconnect or flash by the calling customer forward to the cord board location without actually disconnecting. The MF receiver has the ability, when connected to the toll office side of the TSPS trunk, to generate a +130 volt simplex "ring-forward" signal which is recognized by the toll office trunk and passed to the cord board operator. One or two spurts of this ring-forward signal can be used to indicate flash or disconnect while still maintaining the connection in case the switchboard operator must take some additional action.

4.2.3.3 *MF Outpulser.* The MF outpulser transmits standard 2/6 frequency combinations to the toll office under SPC control. Since several signals must be sent to it by the SPC in fairly rapid sequence, it is controlled by CPD outputs rather than signal distributor points. A pair of CPD enables will cause the outpulser, once connected to the toll office side of the TSPS trunk, to outpulse the correct digit, time the length of the pulse, and time the appropriate interdigital interval. The present timing is that specified for Bell System use and can easily be changed to a lower value if projected higher outpulsing speeds are standardized.

4.2.3.4 *Coin Control and Ringback Circuit.* The coin control circuit can generate three signals, used for coin collect, coin return, or ringback to the calling customer. When E&M signaling between the local office and TSPS is used or when the local office is an ESS office, these three signals are sent as inband tones. In this case the timing of the "quiet period" before tones are sent as well as timing of the tone interval itself are done by the coin control and ringback (CC&R) circuit. Signal distributor point input (from SPC) is used to select the particular signal to be sent.

When the local office trunk uses loop signaling (except for ESS local offices) the three signals are sent by use of "high-low" dc signaling. This requires +130 and −48 volt potentials on the T&R leads which are recognized by marginal and polar relays in the local office. The CC&R circuit applies the correct potentials to the line, again under signal distributor point control, and provides the required timing.

4.2.3.5 *Audible Tone Trunk.* Audible tone is returned to the customer whenever the connection to an operator is expected to take longer than four seconds. The customer would then hear standard 2 second

on—4 second off tone, simulating an operator being rung. This informs him that his call is recognized and that an operator connection is being accepted.

4.2.3.6 *Recorded Announcement and Tone Circuit (RATC).* A recorded announcement is used to inform the customer of unusual delays or catastrophic conditions (fire, flood, etc.), which have caused traffic to back up. Standard procedure is to allow two repetitions of the announcement and then switch to reorder tone. The RATC circuit is arranged to connect to a continuous recorded announcement so as to start at the beginning of the announcement, count two cycles of the message, and automatically switch to a source of reorder tone. The initial connection is, of course, under SPC control and provision is made to have the announcement bypassed and immediate application of reorder tone if appropriate SD point signals are received from the SPC.

Announcements may be provided for a variety of reasons such as local call intercept (LCI) or delay messages. These different announcements require separate groups of announcement trunks. However, in event of a catastrophic condition, special announcements can be recorded on two spare tracks of the announcement machine and these messages would be routed to the two groups of RATC circuits connected to those tracks. As these two groups become overloaded all other RATC circuits can be switched (at the announcement frame circuit) to these two disaster channels. Thus, all RATC circuits in the office can be used for disaster purposes, if needed, without requiring a large, separate, and mostly unused, group of RAT circuits for that eventuality.

4.3 *Maintenance of Trunks*

Automatic maintenance facilities are not provided in TSPS for universal trunks, as responsibility for maintenance of these trunks rests on the local office. The local office performs routine testing to the toll office in the usual manner, with TSPS merely forwarding the test codes which have been pulsed from the local office. If trouble is detected by testing from the local office, a craftsman at the TSPS control display and test frame may be contacted to aid in isolation of the trouble by testing back toward the local and forward to the toll office.

Routine checks made in the TSPS office during the processing of calls may result in indications of trunk trouble. When connecting

service circuits to trunks, continuity checks are made via ferrods in the service circuit. Continuity failures may indicate possible trunk trouble and result in a TTY printout identifying the trunk involved. For outpulsers, continuity and polarity are checked before outpulsing; and continuity is again checked after outpulsing to verify proper response from the toll office. If relays within the trunk fail to operate, a TTY report is also made. When a pattern of printouts indicates a particular trunk, the craftsman may remove the trunk from service and check for possible faults.

4.4 Maintenance of Service Circuits

Trunk group numbers 44, 45 and 46 are test circuits which may be connected to a service circuit in order to diagnose them; (see Fig. 16). The circuits are set in their various states and their reactions are checked via ferrods.
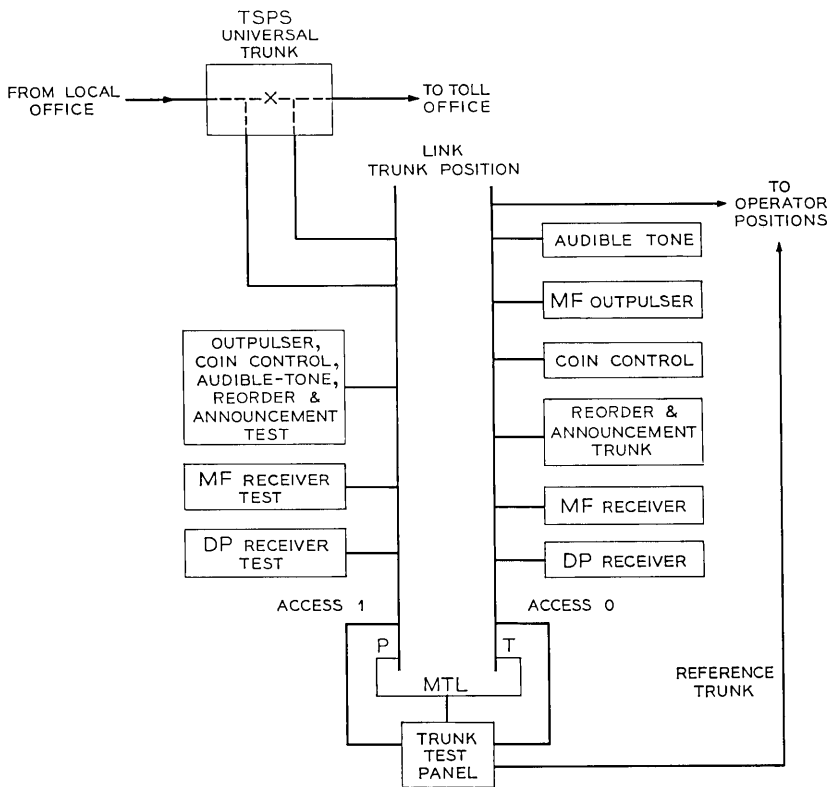
Fig. 16—Test circuits for trunk and service circuits.

Service circuit diagnostics may be initiated by manual requests from the TTY or the control display and test (CDT) frame, by the automatic progression test program, or by requests from other programs which encounter difficulties in using a particular service circuit during call processing.

Service circuits to be diagnosed are processed one at a time by a control program which prepares the circuit for diagnosis and transfers it to the appropriate diagnostic program. The diagnostic programs are broken down into a number of different phases so that a particular test may be requested from the TTY or CDT frame. The tests for each service circuit are summarized in Table II.

Diagnostic results are processed by the dictionary output control program (DOCP) using a pseudo unit type number to produce a trouble locating number in the usual manner. Thus standard dictionary lookup techniques have been extended to the service circuits.

Automatic progression testing of service circuits is initiated twice a day. All circuits are diagnosed with a TTY report only on those circuits which fail diagnosis. The circuit is diagnosed a second time before removing it from service to prevent out-of-service conditions resulting from transient errors.

The test circuits are also monitored for failures. If a test circuit is used on two consecutive failing diagnostics it is considered suspect, and a service circuit which is assumed to be good is selected at random to check the test circuit. The diagnostic for this service circuit is run in the usual manner but the results are used to produce a trouble number for the test circuit rather than the service circuit. In this manner the service circuit diagnostics perform double duty, and diagnostics for the test circuits are obtained for free.

The Circuit Maintenance List Auxiliary (CMLA) is also processed by the trunk maintenance programs. Circuits are placed on the CMLA by audit programs to return them from an unknown hardware state to the idle condition. Circuits are removed from this list, all relays are initialized and the circuits are then returned to the appropriate idle link list. Universal trunks are placed on the high and wet list where they are held in a busy condition until a seizure is no longer indicated. The trunk is then idled.

The authors acknowledge the contributions of W. Fisher who designed the trunks and R. F. Pina who wrote the trunk and service circuit maintenance programs.

V. CONTROL DISPLAY AND TEST FRAME

The Control Display and Test (CDT) frame, shown in Fig. 17,

TABLE II—TESTS FOR SERVICE CIRCUITS

Test Code	Coin-Control and Ringback	Audible Tone	MF Receiver	DP Receiver	MF Outpulser	Reorder Tone & Announcement
1	*Inband Coin Signaling* Check quiet period, level and duration of tones	*Frequency Content and Interval*	*Loss* Check reception for all frequencies under loss and tone duration variations	*Digit 1 and 2* Check proper inter-relation of signals from trunk 2 pulse counter	*KP & ST* Frequency and duration of KP and ST	*Announcement Sequences*
2	*High-Low Signaling* Check voltage and duration of collect, return, and ringback signals		*Twist* Check reception of digits when one fre-quency is appreci-ably lower than the other	*Pulsing* Check proper reception of various pulse rate, percent break and digits	*Tone Duration* Proper duration of pulsing	*Frequency Content of Reorder Tone*
3			*Simplex Rering* Check voltage and duration of signal		*Loss* Check levels of all frequencies	
4			*Saturation* Check ferrods			
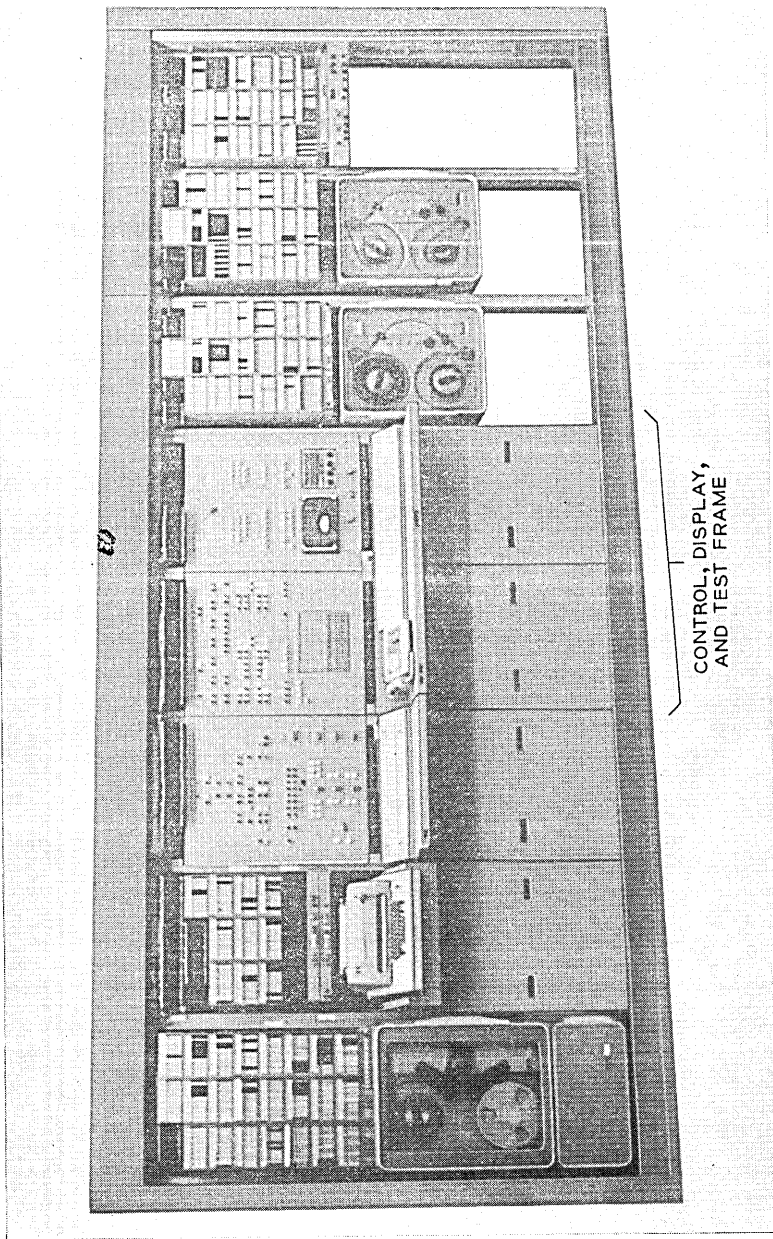
Fig. 17—Control, display, and test frame.

provides display facilities for all TSPS equipment and a Trunk Test
Panel (TKTP) which may be used to test all equipment appearing
on the network.

5.1 *Displays*

Lamp displays are of three types: A/B, primary/secondary, and
trunk group status. Refer to Fig. 18. The A/B display is provided for
units having a single duplicated pair, for example AMAs. One lamp
is provided for each half of the pair and is lit steady to indicate
trouble. Primary/secondary lamps are provided for units having more
than one duplicated pair, such as the Trunk Link Network. The
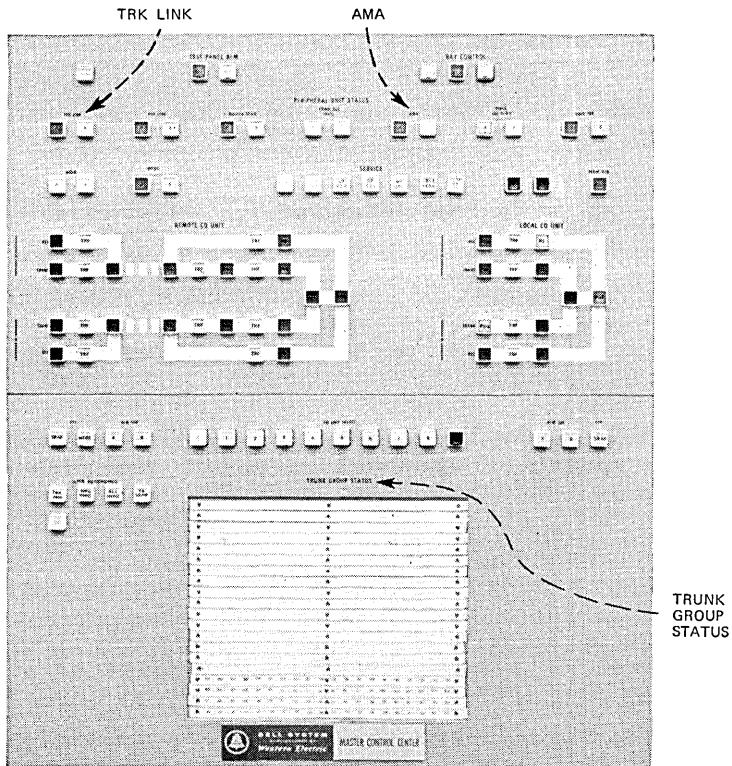
Fig. 18—Display panel.

secondary lamp is lit steady to indicate trouble in one half of a pair, and flashed to indicate such a condition in more than one pair. The primary lamp is used in a similar manner to indicate trouble in both duplicated halves of a pair. Trunk group status lamps are provided for all units which have been assigned to trunk groups; (refer to Table I). The lamp is lit steady to indicate one or more members of a trunk group out-of-service and flashed to indicate when the number placed out-of-service exceeds a service affecting threshold.

The position subsystem is comprised of a serial chain of units and is pictured on the display panel in block diagram form. An A/B lamp display is provided for these units. As the equipment configuration is different for local and remote groups, separate diagrams are provided for each. In the normal case the lamp states from all chief operator groups are ORed together. If it is desired to view the status of a single group, a key may be depressed to select a status display of that group alone.

A test of the more than 500 lamps of the CDT frame may be initiated from the TTY.

5.2 *Trunk Test Panel*

The Trunk Test Panel shown in Fig. 19, has four (4) network appearances (Access 1, Access 0, PMTL, TMTL) as previously shown in Fig. 16. The access circuits 0 and 1 are to connect to voltmeter or transmission test terminations.

The voltmeter termination may be used as a voltmeter or milliammeter, keys being used to select the function and meter scale to be used. By means of graphs contained in the circuit drawing, readings may be converted to resistance readings if desired.

The transmission test termination provides facilities to send tone to, or measure tone levels from the circuit under test. Quiet terminations or noise measurement facilities may also be connected under key control.

A reference trunk to each chief operator group (see Fig. 16) is used for transmission testing with positions, allowing one man tests to be made with any position on a loop around basis. Two man tests may also be performed with a craftsman at the position if desired.

A key telephone set provides for 4 to 15 lines which may be utilized for central office lines, tie lines or intercom lines within the building.
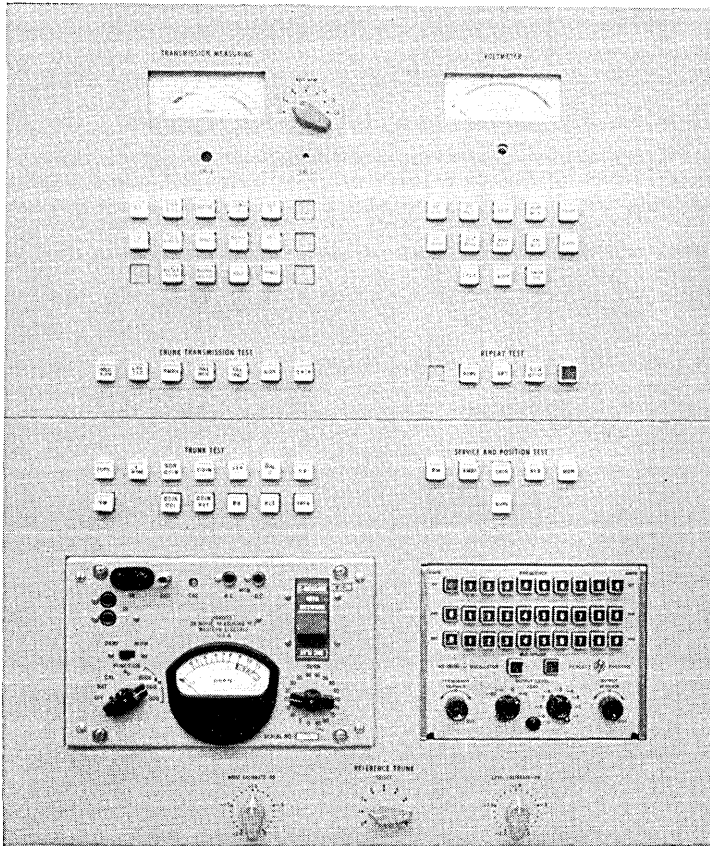
Fig. 19—Trunk test panel.

A telephone headset may be associated either by the key telephone set, or the Master Test Line. If desired, the frame belt line may also be added on a call.

5.2.1 *Test Connections*

Facilities are provided which enable the craftsman to connect any circuit with a network appearance to the test lines at the Trunk Test Panel. The craftsman establishes a connection to a MF receiver by operating the Test key and, using the MF keyset at the test panel,

keys in the Trunk Group Number (TGN) and Member Number (MEMN) of the circuit to be tested. The circuit under test is then connected to the Master Test Line at the TKTP. If a voltmeter or transmission test is to be made, the circuit must be connected to the appropriate access circuit. Under key control, trunks are connected to Access Line Zero; positions or service circuits are connected to Access Line One.

If a trunk, service circuit, or position is traffic busy when testing is requested, it will be automatically camped on. When the circuit becomes idle, it will be removed from service and a TTY message will inform the craftsman that the circuit is available for testing.

Programmed tests may be requested on circuits connected to the test panel by keying in test digits.

5.2.2 Service Circuit Testing

Service circuit diagnostics can be requested by keying in test information after the TGN and MEMN which selects the tests to be performed as shown in Table II.

5.2.3 CPD and SD Operations

The craftsman is given the capability to easily operate or release any relay in any circuit on the network including the Trunk Test Panel itself. He can also pulse any unipolar or bipolar CPD point in any circuit on the network. Once set up from the TKTP these tests can be controlled remotely at the circuit location by plugging a pushbutton control (32A Test Set) into a frame belt line.

5.2.4 Scan Display

When the Scan Display key is operated a program displays the states of the ferrods associated with the circuit that is under test by the TKTP. The ferrod states will be displayed on the Program Display lamps on the SPC Control and Display Panel and will be extinguished when the key is released.

5.2.5 Maintenance Busy, Out-of-Service, and High and Wet

Trunks, service circuits, or positions are unavailable for service (indolent) because they are "maintenance busy," "out-of-service," or "high and wet." Maintenance busy implies that the circuit is under control of a maintenance program; that is, the circuit is about to be

tested from the Trunk Test Panel or by a diagnostic, or is currently under test, or has just been tested and has not yet been restored. Out-of-service implies that the circuit, after being tested, has been found faulty and is being kept from service until it can be repaired or replaced. Incoming trunks that are maintenance busy or out-of-service appear busy at the originating office. All circuits that are out-of-service are kept on a list which can be printed out on the TTY by operating the Busy Status key or by TTY request. High and Wet implies that the circuit (trunks only) is in the state of permanent seizure.

The Trunk Test Panel program makes a circuit maintenance busy before testing and restores it to idle or out-of-service after testing. A circuit can be taken out-of-service after failing a test, or restored to service after being repaired by operating the Make-Busy key or Remove Busy key.

5.2.6 *Thru Test and Incoming Call*

Trunks incoming to TSPS from local central offices are routinely tested at the local office by sending codes over the trunks which request connections to test equipment in the toll office. The TSPS receives these codes, identifies them as test codes, and checks their validity. If found valid, they are pulsed forward to the toll office, and the trunk is placed in a state appropriate for the test. This eliminates contacting the TSPS maintenance center for most tests.

If a trouble is found, it may become necessary to sectionalize the trunk to locate the trouble. A feature is provided to permit the local office to contact the TSPS maintenance man over the suspected trunk. The craftsman at the local office dials another test code which the TSPS recognizes as an incoming call to the TKTP. The audible signal at the TKTP is rung and the MTL lamp flashed, indicating an incoming call; audible ringing is returned on the trunk. When the craftsman operates the Test key, the calling trunk is connected to the Master Test Line.

5.2.7 *Trunk Class Lamps*

Universal Trunks connected to Access Line Zero will have their class of service identified by the Trunk Class Lamps. Lamps light to indicate that the trunk carries noncoin, coin, 1+, 00, and/or 0+ traffic. If any trunk (not restricted to universal trunks) is a 4-wire trunk, a lamp indicating 4-wire lights and several relays within the

trunk test panel operate to configure the test circuit for the 4-wire trunk.

When testing universal incoming trunks, it is necessary to test the trunk toward the originating office, toward the toll office, and in the "cut-thru" state. The desired state is established automatically by the program in response to the Toll Only and Toll Bridged keys making it unnecessary for the craftsman to obtain the trunk type, study the SD and CD, and decide which relay must be operated.

Outpulsing is necessary to complete a test toward the Toll Office. This is done by requesting a connection to Access Line Zero and keying in the outpulsing information after the TGN and MEMN. Then by operating one of the "TOLL" keys and the OP key on the TKTP, the proper code will be outpulsed.

5.2.8 Equipment Location

Once a trunk or circuit has been found faulty, it must be repaired or replaced. This requires knowledge of its physical location in the building. In past systems, a translation would be provided by a book. Operation of the TCN key will cause a machine translation and TTY printout of the equipment location and network appearance of the circuit last connected to the Master Test Line.

5.2.9 Position Testing

The transmission path to positions may be tested from the trunk test panel by means of reference trunks extending from the test panel to each chief operator group. Located at the position end of the reference trunk are a milliwatt supply and terminations which are used in conjunction with the test meters at the panel to calibrate the reference trunk for transmission testing. The Access 1 circuit is connected to the other end of the transmission path through the network. Two modes of testing are provided; one-man tests, and two-man tests.

The one-man position test consists of four separate parts; near-end to far-end loss and noise, and far-end to near-end loss and noise measurements. Before making these tests the reference trunk is calibrated. The calibration of the reference trunk consists of three separate parts; loss and noise calibration and a noise level check.

A two-man test may be made with one man at the trunk test panel and one man at the position, using the following portable test equipment which is plugged in at the position:

21A   Transmission Measuring Set
3B   Noise Measuring Set
132A   Test Set.

In this case the position is connected to Access line 1 but no reference trunk is used.

5.2.10 *Operator Keyed Trouble Reports*

Instead of filling out trouble tickets, operators key trouble codes into the TSPS when they encounter trouble or receive reports of trouble. These reports, and as much pertinent call data as is still in the system, are printed on the service bureau TTY and/or on the toll office TTY. Certain of the reports, which may be TSPS affecting, will be printed at the maintenance TTY. The craftsman is expected to monitor these reports and act accordingly if he detects a pattern indicating a common piece of equipment.

The authors acknowledge the work of W. A. O'Connell who designed the CDT frame and J. W. Seazholtz, P. J. Brendel, A. W. Robinson and R. A. Thompson for their many contributions to the program.

VI. THE TSPS POSITION SUBSYSTEM

The TSPS position subsystem links the peripheral address bus, master scanner, and link cut-through circuits with the eyes, ears, hands, and voice of the operators. Specifically, voice circuits are provided to connect an operator to the trunk; and data is sent to light incandescent lamps and multidigit *Nixie** displays. At the same time data is returned from operator key actions or alarm outputs. Provision is included to reach operators at remote locations as well as operators in the same building. Thus, small groups of operators can be employed in suburban "traffic offices" while the full access link and single common control provide the efficiencies inherent in a single large group of operators. A single position subsystem provides circuitry for up to 62 operators, 2 supervisors, and one chief operator. A maximum of 9 such subsystems can be provided although the total number of positions cannot exceed the original design intent of 310 and, moreover is dependent on the traffic mix at a particular location.

6.1 *Local Subsystem-Sending*

In the local case, shown in Fig. 20, the path from the address bus to the operators is comprised of a group gate, position signal distributor, position buffer for each operator, and a console for each operator. The group gate acts as a time buffer and also converts binary information from the high-speed bus to a set of 1-out-of-8 (1/8) codes applied to the position signal distributor (PSD). Transfer relays are included between the group gate and position signal distributor for use in fault recognition and to permit operation with two bad units. The PSD acts upon the 1/8 coded instructions to select the desired position buffer and operate or release a miniature wire spring (MWS) relay within that position buffer. Outputs from the buffer connect to incandescent lamps within the operator's console. The MWS relays of the buffer are predominantly magnetic-latching and are arranged on printed circuit boards of the A-pack type.

The "local" case actually provides for the operator group to be

PSD — POSITION SIGNAL
        DISTRIBUTOR
SSG — SENDING GROUP GATE
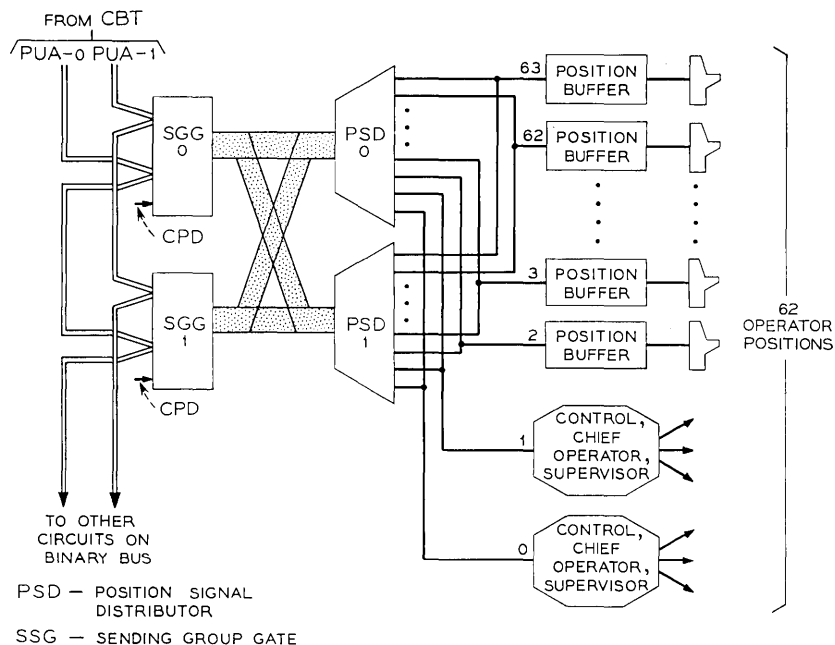
Fig. 20—Local subsystem—sending.

located up to about 4 miles from the group gate at the base location. The sets of 1/8 leads between group gate and PSD can be run as twisted pairs in cable facilities for that distance. Longer distances (the remote case) require additional circuitry interposed between the group gate at the base end and the PSD at the remote end. This remote operation is discussed in more detail later.

6.1.2 *Position Group Gate (Local)*

The group gate connects to the peripheral address bus and central pulse distributor in the same manner as other bus-connected circuits such as scanners, signal distributors, etc., as shown in Figure 1. It is actually made up of two independent halves with the usual 4 enables. Normally both halves are used simultaneously to send data to a chief operator group, providing two separate information paths. Should one half develop a fault, the remaining half is used to send all the information previously sent via the two independent paths. Input information is in the form of a 17-bit word arranged as shown below:

| 16 | 15 | 14 13 12 11 10 9 | 8 | 7 | 6 5 4 3 2 1 0 |

PAR NIXIE   ADDRESS   N/M O/R   INFO

As used here "address" means the operator position number and "info" refers to a particular relay in the associated position buffer. Bit 7 designates operate or release, bit 8 designates normal or maintenance order, and bit 15 is used when sending multidigit Nixie displays. The Nixie displays, which may be up to 12 digits in length, are used to give time, coin charge and initial period information, calling or called numbers, and other numerical information to the operators. A special part of the PSD is dedicated to Nixie displays as described later.

The group gate is reset and prepared for the next order upon receipt of a "check-back" pulse. For maintenance orders this pulse is generated within the group gate itself while for normal orders to a position the pulse originates at the particular MWS relay concerned and is regenerated and passed on by the PSD.

In the maintenance mode the information portion of the group gate order may be intended for the group gate itself or some subsequent circuit. Certain orders that are acted upon by the group gate could also cause some reaction (usually undesired) in subsequent circuits, and these maintenance orders are blocked by the group gate to pre-

vent this. With the exception of transfer and quarantine orders, these various maintenance orders are provided for diagnostic program purposes to allow the programs to temporarily set up checking configurations and comparison circuitry as a diagnostic check is performed.

6.1.3 Position Signal Distributor

The position signal distributor (PSD), shown in Fig. 21, and all circuitry following it is arranged the same for both local and remote traffic offices (operator groups). It is made up basically of two relay trees, one of which selects a set of cut-through relays to connect a multi-lead bus to a particular buffer while the other connects an apex pulser to a particular lead of this bus. It differs from a universal trunk frame signal distributor in that it is completely duplicated, including all relays of both trees. Thus, if any portion of one half of the PSD should fail it is still possible to reach every lamp in every position via the remaining half.

6.1.3.1 *Apex Pulser and Check-Back Pulse.* When a path is established to a particular MWS relay in a position buffer, an apex pulser sends operate or release current through the relay winding. As the relay armature moves, a momentary ground is generated from a de-
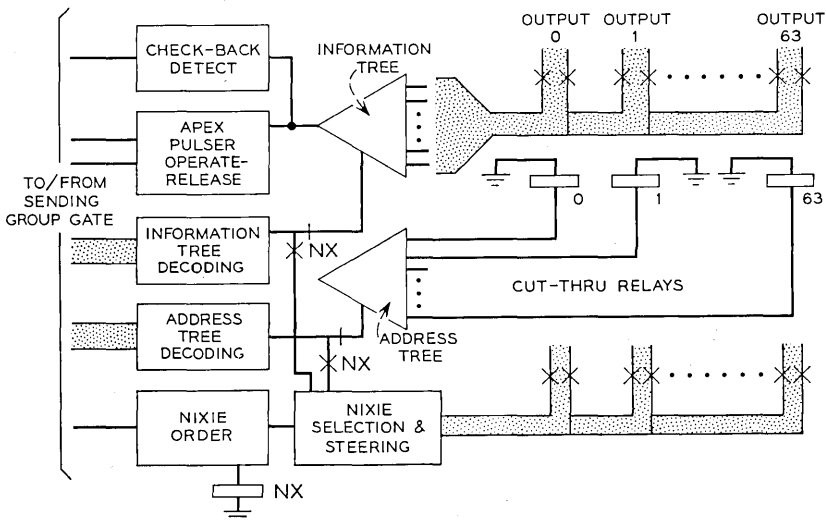
Fig. 21—Simplified schematic of position signal distributor.

liberate "bunching" of one contact set and used as a check that the relay did, indeed, operate or release. Since this check-back path is common to all MWS relays of a position buffer, a check for permanent ground (caused by a poor relay with "floating" armature) is made by the PSD prior to any relay operation. This check-back pulse sets a flip-flop in the PSD which causes pulser current to be cut off and a regenerated pulse to be sent to the group gate. In the event the MWS relay fails to function normally and no check-back pulse is generated, the group gate is not reset, and subsequent scanning of a maintenance ferrod (F point) will alert the SPC to the failure. The SPC then sends a system reset to clear the group gate. This releases the PSD and cuts off the apex pulser current. As a back-up to the above two methods of cutting off pulser current, a timer is started on each PSD order and is designed to cut off pulser current in about 12 milliseconds. Normal operate or release time for the MWS relay is about 5 ms. The timer will cut off the current even if a system reset is not sent. At the same time an alarm is generated and returned by the Alarm Sender (described later) to alert the SPC that a time-out has occurred. While the bulk of the PSD is comprised of relay trees and relay logic, the check-back circuitry is all electronic and provides rapid return of the reset signals.

6.1.3.2 *Nixie Display Orders.* A multidigit display such as a 10-digit number reaches the operator via Nixie tubes positioned near the top of her console. A total of 12 Nixie tubes are provided (to ultimately handle digit displays for overseas dialing). Since each tube contains 10 cathodes (digits), a total of 120 subsystem orders pertain to Nixie tube operation. Actually a few more orders are used since preparing for the display and releasing the display must also be considered.

Nixie displays are characterized by the necessity for sending several orders to one position in a short time. This differs from the incandescent lamp displays where lamps are lighted singly at a position with several seconds between each order as the operator takes various actions on a call. Because of this difference a separate portion of the PSD handles Nixie orders exclusively.

The technique used is to dedicate the PSD to a particular position for the length of time needed to send a complete Nixie display. A "priming" order is first sent to the PSD to establish this dedicated path. This order results in two 10 lead buses and one 6 lead bus being cut through to a particular position. A signal on one lead of the 6-bit bus selects a pair of Nixie tubes. This is immediately followed by

a pulse on one lead of each of the 10-bit buses to select a digit in each of the pair of tubes. The advancing to successive pairs of Nixie tubes is accomplished automatically by the PSD, while the selection of a particular digit is dependent upon receipt of orders from the stored program control. After advancing through all 6 pairs of tubes, or at any time the program reaches the end of the display being sent, a "release" order is sent to the PSD. This drops the connections to the position and restores the PSD to its normal undedicated state.

The pulses mentioned above are not applied to the Nixie tube elements directly but instead are used to break down (ionize) a miniature neon lamp provided in series with each Nixie tube cathode as shown in Fig. 22. This neon lamp, which is a very low cost device, has two well defined states and is used as the "memory" element as well as a "switching" element of the Nixie display. Thus a display is "locked-in" through the medium of the ionized neon lamps, and the Nixie display remains ON after the PSD releases from the position. It is later extinguished under either program control or operator actions by momentarily opening the holding battery. The entire display, including Nixie tubes and sockets, all neon lamps, and all biasing resistors, is mounted on one printed wiring board and can be easily removed for maintenance purposes. The complete display board is shown in Fig. 17 in the article "TSPS Physical Design" in this issue.

6.1.3.3 *PSD Maintenance Orders.* The PSD is arranged for several different maintenance checks. The electronic check-back circuitry can be tested for both steady ground conditions and for lack of ability to detect the check-back pulse from MWS relay. The timers can be checked to insure proper timeout and that alarms are generated and returned via the alarm sender. One of the most powerful maintenance checks, however, is the ability to take two output trees, one which normally selects a position, and one which normally selects a particular lamp at a position, and connect them to each other. Orders can then be sent which are expected to result in a match or are expected to mismatch with the resulting actual matches and mismatches used to pinpoint troubles.

If tests such as mentioned above result in trouble being detected in a PSD other maintenance orders can be sent to quarantine the half of the circuit found in trouble. Transfer relays can be operated to link the A half of the PSD with the B half of a previous circuit. Any such orders, upon execution, generate alarm codes which are returned
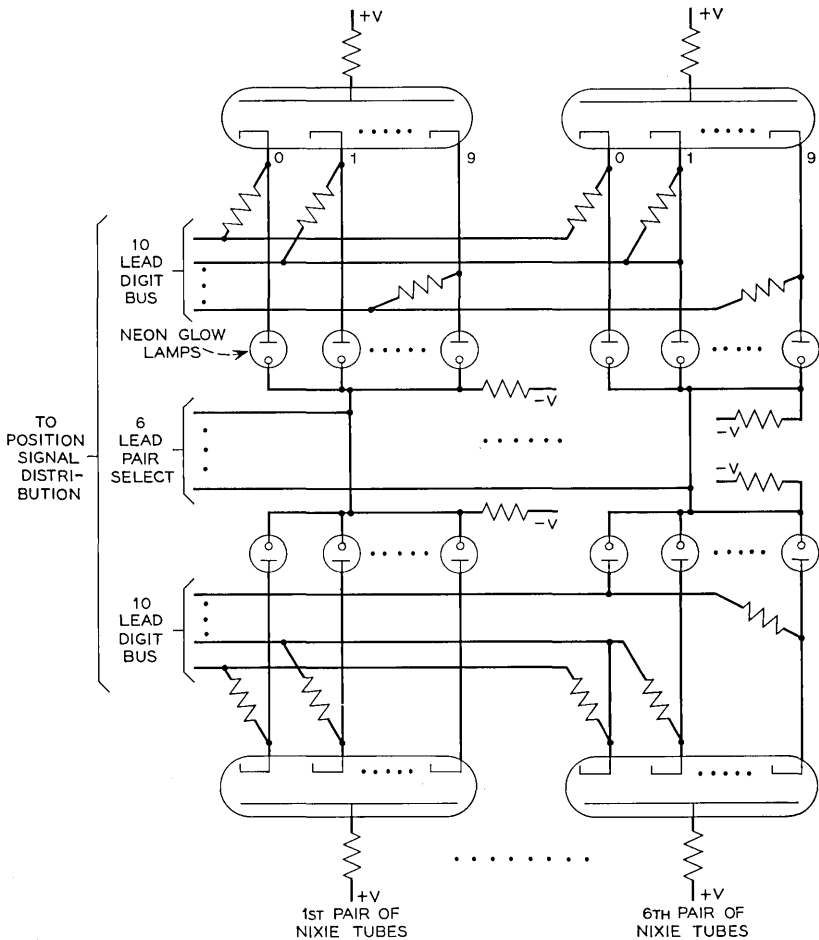
Fig. 22—*Nixie®* display printed circuit board.

to the SPC via the alarm sender and serve to verify that such orders were correctly executed.

6.1.4 *Position Buffer*

The position buffer consists of a number of A-type circuit packs, each of which contains from 3 to 8 miniature wire spring relays. The MWS relays operate incandescent lamps on the operator's 100B console. They are also used to cut through a transmission path and extend the transmission path to a supervisor or monitoring circuit

as desired. These miniature relays are predominately magnetic latching and operate or release on a short pulse of current from the PSD with no holding current required. A few are not magnetic-latching and are operated by a pulse from the PSD but lock to contacts of some other relay. This facilitates the release of position lamp displays since one or two key relay operations at the end of the operators work will release several relays. Provision for growth is made by providing for 16 buffers per frame and by permitting frames to be partially equipped. As positions are added, a set of 10 plug-in circuit packs, which constitutes a position buffer, is inserted for each position.

6.2 *The 100B Console*

The physical aspects of the new operator consoles are discussed in the article "TSPS Physical Design" of this issue.[4] The additional features covered here include arrangement of key codes for receipt of information from operators and transmission arrangements.

The information keyed by the operator utilizes a 3-out-of-9 code (3/9). That is, each of the keys has contacts which close ground to 3 leads of a 9 lead bus when the key is depressed. Use of a 3/9 coding scheme has several advantages: each key needs just 3 make contacts (a fourth contact is sometimes used to light a lamp as the key is depressed), 84 combinations are possible (allowing for future growth), only 9 leads need to be cabled to the position itself, and the code is self-checking as is any M/N code. A detector circuit mounted within the position insures that a 3/9 condition is present before any service request is sent to the position scanner. This prevents the system from dealing with false 1/9 or 2/9 codes generated during the downward travel of the key. Since valid information is present only while the key is depressed, the scanner receives a pulse of information and must be fast enough to reach the position while the pulse is present. In the worst case (fastest operators) these pulses will be about 35 ms in length.

The 100B console is arranged for 4-wire transmission. As already noted, the link and service circuits, as well as several trunks, are all 2-wire transmission circuits. A 24V4 repeater physically located near the link frames converts the 2-wire path to 4-wire and also provides sufficient gain to reach remote positions up to about 10 miles distant. Where T-carrier is used to reach remote positions, the 2-wire to 4-wire conversion is provided in the T1 channel banks. A transmission path to 3 separate supervisory trunk circuits connects to each position.

These are used to connect to a supervisor for needed assistance or to permit monitoring from either supervisor console. Another transmission path connects to all positions and permits monitoring from a specially equipped monitoring position.

6.3 *Local Subsystem Receiving*

6.3.1 *Position Scan and Gate*

The position scan and gate (PS&G) circuit, shown in Fig. 23, is arranged as an autonomous scanner continually looking for key signals from the operators. As mentioned in Section 6.2, the keys are arranged to provide a 3/9 code, and a detector circuit is provided in each console to discriminate against 1/9 or 2/9 codes. When a 3/9 code is detected, a "service request" lead to the position scan and gate is
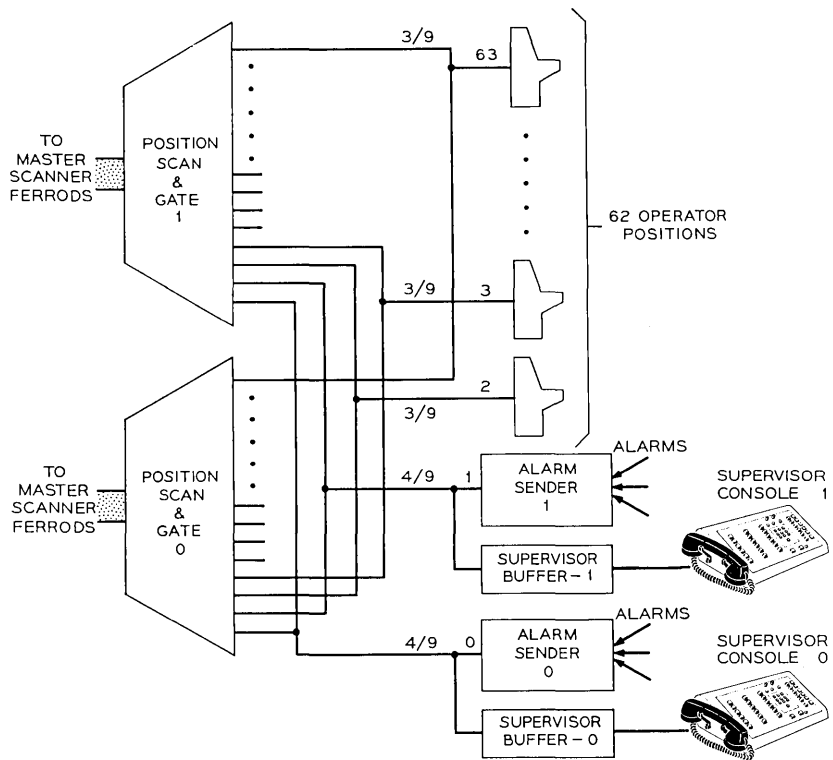
Fig. 23—Local subsystem—receiving.

energized. This service request lead is in addition to the 9 leads of the 3/9 bus. The scanner is continually checking these service request leads, and upon noting a signal, it stops at that point and accepts the 3/9 code information.

The 3/9 information, together with a 6-bit binary number indicating the position involved and a parity bit, is returned over a 16 bit bus to the SPC via a master scanner. Sixty-two of the 64 possible inputs to the PS&G come from positions. The other two inputs are shared between two supervisor consoles, a chief operator administration circuit, and an alarm sender circuit.

The scanner is controlled by a 40 kHz oscillator driving a 6-bit binary counter. The binary counter outputs are arranged in two groups of 3, and each 3-bit segment is expanded to a 1-out-of-8 (1/8) code output. The two 1/8 groups are arranged in a horizontal-vertical grid arrangement with a 2 input gate at each of the 64 intersections. Thus, 64 outputs are obtained with a pulse appearing sequentially on each. This scan pulse is combined with a service request, when present, to stop the oscillator and inhibit further scanning. The state of the 6-bit binary counter is sampled to generate a position number, the 9 leads from the position (or other circuit) are sampled to obtain the keying information, a parity bit is generated, and the resulting 16-bit output is sent to a master scanner ferrod row. When the SPC scans this row and accepts the information, it sends a "scan complete" pulse via the group gate to the position scan and gate. This restarts the oscillator to permit scanning to resume and sets a flip/flop to prevent the scanner from again stopping on the same point. When the operator releases the key, this flip/flop is cleared and any subsequent keying by that operator will cause the scanner to again stop on that position.

The interval allowed for a complete scanning operation as described above is 10 milliseconds. This is a compromise between sufficient speed to insure reaching key signals as keys are depressed and moderate scan rates to insure not placing a real-time burden on the SPC (fast scans result in *no* information most of the time). Thus, the scanner would expect a restart of the oscillator within 10 ms. If no restart occurs after about 150 ms, then the scanner is automatically placed in quarantine, alarms are generated, and diagnostics are requested. Checks are also made to insure that the scanner progresses correctly from position to position. If failure is noted, the quarantine state is automatically entered. The scanner is fully duplicated and operates by a "leap-frog" arrangement wherein one scanner skips a position if the mate scanner is momentarily stopped at that position.

6.3.2 Alarm Sender

The alarm sender receives alarm information from all subsystem circuits except the group gate and temporarily stores them until such time as the PS&G can accept them and transmit them back to the SPC. The storage and ultimate transmittal to the PS&G, as well as a progression arrangement to allow handling of several alarms at once, is accomplished primarily through relay logic.

Circuits sending alarms to the alarm sender quite often send 2 or 3 alarms in succession. For example, if the PSD sends a timer alarm indicating that an operation took too long, it will also send an indication of the state of the transfer relays and an indication of the operational status (normal, quarantine, power off) of the circuit. These are picked up by successive scans of the alarm sender and sent to the SPC. Alarms are arranged in a 4/9 code to distinguish them from operator key codes. Also, in the case of alarm codes, a maintenance bit is sent by the PS&G as well as the 16 bits of information mentioned in Section 6.3.1. This maintenance bit appears as a separate ferrod indication scanned by the SPC to insure immediate detection of maintenance information. The 6-bit position number becomes a "circuit number" indicating which circuit (PSD, PS&G, 100B console, interrupter, etc.) is generating the alarm.

The alarm sender also provides for input information from the position subsystem frame location by the maintenance man using a combination of thumbwheel switches. The thumbwheel switch allows the maintenance man to set up an order, verify visually that it is correct, and then have it scanned and sent to the SPC. The SPC then acts on the order perhaps causing some requested reconfiguration or some lamp display at a position. This gives a maintenance man concerned with position subsystem maintenance (which might be remote from the rest of the TSPS installation) some of the control normally provided by the MCC maintenance teletypewriter.

6.4 Remote Subsystem—General

The remote position subsystem, shown in Fig. 24, was designed for use with T1 carrier to reach operators up to 80 miles from the base location. The group gate, T-carrier, and Data Assembly and Check (DAC) circuit comprise a data link that is unique and worthy of discussion. T1 carrier terminal equipment encodes 24 analog circuits (voice) into a serial pulse stream and reconstructs the 24 analog signals at the distant end. The sampling rate is 8 kHz so that each of
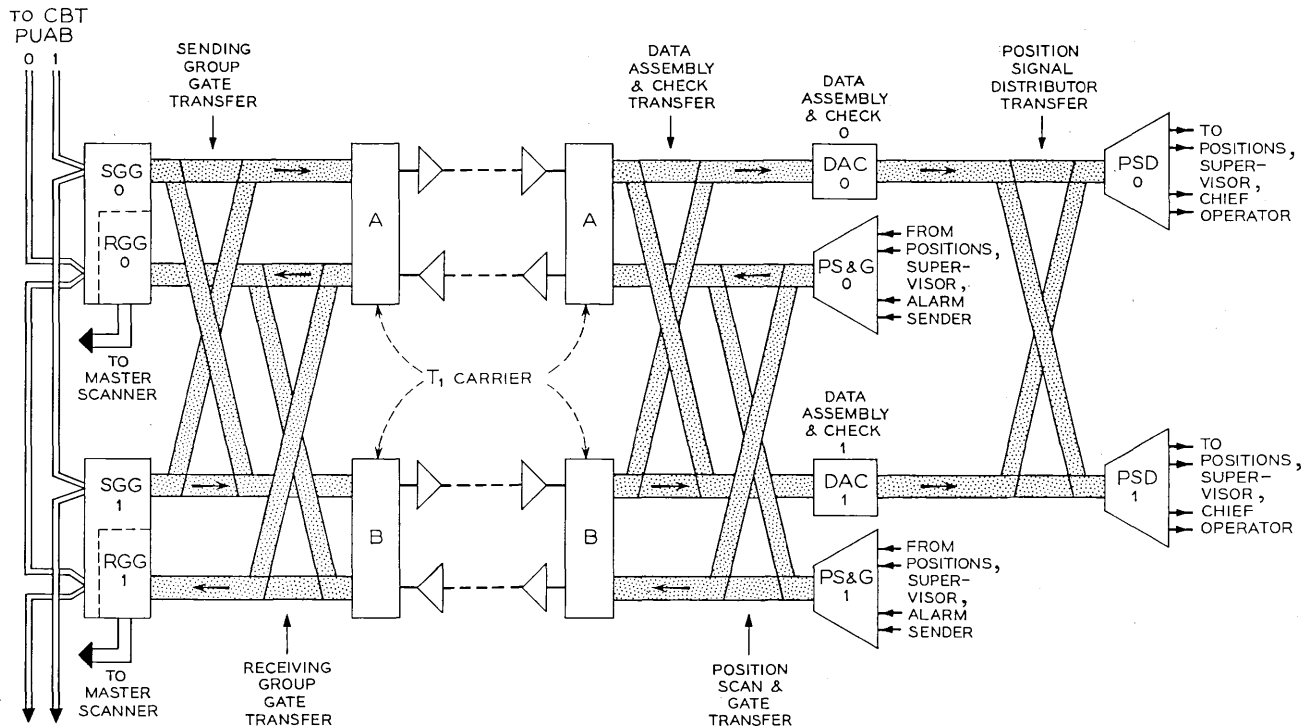
Fig. 24—Remote position subsystem configuration.

the 24 channels is scanned every 125 microseconds. The terminal equipment is designed to encode the amplitude of the analog input into a 7-bit binary number but actually transmits an 8-bit binary number. The eighth bit is normally used as a "signaling" bit to indicate a 0 or 1 supervisory state.

For TSPS the ability to encode 24 voice signals into digital form is used intact to provide voice circuits for 24 remote operators. The 8th bits from each of the 24 channel are combined to give essentially a 24-bit word which can be sent from the group gate at the base location to the DAC circuit at the remote location. It is possible to send a new 24-bit word every 125 μs, and thus considerable data (to control lamp displays and return key signals) can be sent to and from operator positions over the same path that is needed for the voice circuit. This is the rare case where it seems we are obtaining "something for nothing." In practice two T1 banks are used for reliability and either one alone can handle all data to and from a chief operator group (62 operators, 2 supervisors and 1 chief operator). These are designated T1-A and T1-B, with T1-A also carrying the voice circuits to 24 of the 62 operators and T1-B serving as a voice backup for those 24 operators. To serve up to 48 operators a third T1 bank is added and designated T1-C. T1-C handles no data but provides 24 voice circuits and is also backed up by T1-B via a relay transfer arrangement. For full groups of 62 operators, a 4th T1 is added and partially equipped. It, too, can be backed up by T1-B.

6.4.1 *Remote Subsystem—Sending*

The sending portion of the position group gate, when arranged for remote operation, acts only as a time buffer and does not change the format of the instructions received from the SPC. Thus the same 17-bit word received over the binary bus is applied to 17 of the 24 T1 channels. Other channels are used for synchronization, maintenance transfer of paired DAC circuits at the remote end, and a special TTY circuit to the remote site. The 17-bit word is sampled by T1 for 5 ms which results in 40 consecutive transmissions of the same information to the remote end. At the remote end the DAC circuit receives the information from T1 carrier and eventually converts it to the set of 1/8 codes required by the position signal distributor (PSD). The PSD, as mentioned before, is the same for both local and remote operation.

6.4.2 *Data Assembly and Check Circuit (DAC)*

The DAC circuit checks parity on the first T1 transmission and, if

good, temporarily stores the 17-bit word. It then checks the second T1
transmission (the second transmission of the *same* information) and,
if parity again checks, it compares the two 17-bit words. If the com-
parison test passes, it then converts the binary information into a set
of four 1/8 codes (plus a few other outputs) for use by the PSD. If
the comparison fails then the stored information is discarded and two
more T1 transmissions are checked and compared. This process con-
tinues until a valid comparison is found or until all 40 T1 transmis-
sions have been completed. A "comparison failure" alarm is then gen-
erated and returned to SPC via the alarm sender.

The combination of both parity and comparison guards against bad
data due to short noise bursts on the T1 line. The probability of a few
consecutive pulses being mutilated in a stream of pulses is not too low
and the DAC effectively increases the overall reliability of the sending
chain. In order to periodically check the operation of the comparison
test feature in the DAC the group gate is arranged, under control of
a maintenance order to send alternate "ones" and "zeroes" on con-
secutive T1 transmissions. It is also arranged to send bad parity to
check the parity circuits of the DAC. Other diagnostic checks used
for DAC maintenance depend upon the succeeding circuit (PSD) diag-
nostic tests to isolate trouble.

6.4.3 *Remote Subsystem—Receiving*

The use of the 8th bit on each T1 channel is the same for sending or
receiving. The output of the Position Scan & Gate (PS&G) is changed
slightly to connect to T1 carrier instead of ferrods on a master scan-
ner. The 17-bit output from the PS&G (3/9 code, 6-bit position num-
ber, parity, and maintenance bit) is applied to T1 carrier until a
"scan complete" pulse is received by the PS&G and scanning resumes.

The 17-bit information is sampled by T1 carrier every 125 μs and
transmitted to the receiving portion of the group gate at the base loca-
tion. This receiving group gate (RGG) accepts the data and checks
parity but does not make the comparison test as made in the other
transmitting direction by the DAC circuit. This is permitted because
the data is intended for the SPC where the correctness of 3/9 codes
and parity can be checked. The RGG does, however, require two trans-
missions of the information in case the T1 sampling at the distant
end occurred in the middle of the 17-bit data.

Once two transmissions have been received, the RGG generates an
"information present" signal to a ferrod scanned at a fast rate (10
ms) by the SPC. SPC then makes a directed scan of the 16 ferrods

containing position number, 3/9 code, and parity. It next sends a "scan complete" (CPD pulse) signal to the group gate which forwards it over T1 carrier to the distant position scan and gate circuit. This restarts the PS&G oscillator and permits scanning to resume as in the local case.

Thus for the remote receiving case the only additional circuit (other than the carrier system) is the receiving group gate. However, the maintenance procedure is more elaborate since two sets of transfer relays are involved; one at the output of the PS&G as it connects to T1 carrier, and another set at the input to the RGG at its connection to T1-carrier. Remote subsystem diagnostics use these transfer points plus portions of diagnostics on succeeding circuits to isolate trouble. The position subsystem maintenance programs are described in succeeding sections.

6.5 *Auxiliary Subsystem Circuits*

There are a number of small circuits used with the position subsystem for control, monitoring, and traffic data purposes. Circuitry is provided for two supervisor positions and one chief operator position as well as a wall-mounted traffic register cabinet. The traffic register cabinet contains registers and a jack field which permit traffic data to be recorded from any 9 traffic service positions. These register counts can be read and reset by the chief operator and supply information to supplement the regular traffic printouts on her TTY machine.

Up to four administration cabinets may be provided, one of which contains lamps and keys used to facilitate training of inexperienced operators plus a few lamps to indicate heavy traffic or system initialization. Five traffic service positions, identical to others in all respects, are designated in the stored program as training positions. By use of the proper keys, the chief operator can exclude certain types of traffic from each of the five positions. As new operators at these positions become more proficient, the chief operator can change the restriction to permit a wider variety of calls at any one or all of the five positions. Alternatively she may use them for all traffic just as the rest of the positions are used.

The supervisor consoles and associated circuits are more sophisticated than the wall-mounted cabinets just described since they permit the two supervisors to perform a variety of functions. The supervisory buffer contains a small switching network so that calls arriving

over one of three trunks from operators, or one of three transfer trunks from the toll office can be switched to either console or any one of four jack appearances positioned around the operating area. These jack appearances allow a roving supervisor to plug in her headset and be connected immediately without returning to her console. Flashing, wall-mounted lamps alert her to an incoming call.

The calls to the supervisor may be from the operator directly or may be from a customer who was transferred to the supervisor via a toll office connection. This latter type of call would have been set up by a TSPS operator after hearing the customer specifically request a transfer to her supervisor. The former type call, coming from an operator requesting assistance, appears on one of three trunks common to all operator positions. In addition to the small network, the supervisor buffer contains the relay logic to light lamps at the supervisory consoles, place calls on hold, permit monitoring by a supervisor of any operator talking path, and permit calls to be originated by either supervisor to any operator or to the outside world.

These circuits, together with the alarm sender, share the first two position numbers (0 and 1) in the 6-bit position address. That is, the first two position buffers on the PB frame are not intended for operator positions but instead provide outputs to these other circuits. Similarly, the first two inputs to the position scan and gate do not come from positions but instead arrive from these other circuits. Thus, in a TSPS traffic office the first physical position is designated position number 2 and the 62 positions use up the remainder of the 64 position addresses. This is illustrated in Figs. 20 and 23.

One other subsystem circuit worthy of note is the teletype channel and machine provided for each chief operator. In addition to several other TTYs used by TSPS for other jobs, each chief operator group (traffic office) is provided with a receive-only TTY. Traffic data for the associated operator team is sent to these machines at regular intervals by the SPC. Maintenance information pertaining to that group can also be sent when requested. In the local subsystem case the chief operator's TTY is handled in the same manner as other TSPS TTYs; i.e., a TTY buffer is reached via the PUAB and connects to the TTY machine. However, in the remote case use is again made of the T1 carrier signaling bit (that "free" eighth bit!). In this case the TTY buffer output is applied to a spare T1 channel ("spare" only in the data sense—17 of the 24 channels were used for group gate output—all are used for voice). The 125 μs sampling rate is much faster than

the TTY requires and permits simple reconstruction of the TTY signal at the distant end. Thus, no data sets or dedicated outside line are needed for remote chief operator TTYs.

The authors acknowledge the many contributions of S. Balashek, F. Luludis, J. Kunish, L. Caron and E. C. McIntosh on the position subsystem hardware.

VII. POSITION SUBSYSTEM MAINTENANCE PROGRAMS

7.1 *General*

The position subsystem maintenance program package is designed to complement reliable, duplicated hardware so that the hardware-software combination accurately passes information from the peripheral unit address bus to the operator and from the operator back to the master scaner without interruption of call processing operation. This package of programs includes a fault recognition program, diagnostic programs, automatic exercise programs, demand exercise programs, and hardware initialization programs.

The fault recognition program detects abnormal circuit responses of two types; errors and faults. An error is a statistical failure usually caused by noise which does not persist from a short term point of view, whereas a fault is a hardware failure. In both cases, it is the responsibility of the fault recognition program to:

(*i*) successfully complete the intended system operation,

(*ii*) remove a faulty unit from service,

(*iii*) identify the fault for maintenance personnel by teletypewriter (TTY) message or request a diagnostic to be run later which will identify the fault, and

(*iv*) return a working system to call processing as quickly as possible.

Diagnostic programs are written to facilitate quick repair by maintenance personnel to minimize the possibility of simultaneous failures in duplicated units taking a chief operator group out of service. These programs, running under control of the Maintenance Control Program (MACR), compare the actual test results with expected results, and pass the comparison results to the dictionary control program (DOCP) for data reduction and generation of a dictionary number which is printed on the TTY. Maintenance personnel look up the dictionary number to determine the location of the fault.

Since dictionaries are generated for single faults only, it is important to discover a single fault before a second fault develops. Some faults, particularly those in circuits which perform maintenance functions, will not be detected during normal call processing. Therefore, automatic exercise programs are run on a regularly scheduled basis to uncover those faults.

In cases where it is not economical to provide fully automated diagnostic capability (e.g., double fault and intermittent fault situations), demand exercise programs may be called by maintenance personnel to provide diagnostic test results (called "raw data") or test order sequences suitable for visual or test equipment checks to aid them in their analysis of the trouble.

Finally, initialization programs are provided which return either an individual operator position or the whole position subsystem to a known initial state. An individual position is initialized to prepare that position to receive calls whenever an operator plugs in her headset or upon request by the audit programs. A position subsystem and all associated operator positions are initialized when some major system failure has allowed the software records of hardware states to become jumbled.

7.2 *Position Access Fault Recognition Program (PAFR)*

As indicated previously, it is the responsibility of the Position Access Fault Recognition Program (PAFR) to respond to any indication of abnormal circuit response (a trouble—either an error or a fault), complete the intended system operation if one was in progress, remove a faulty unit from service, identify a fault for maintenance personnel, and return a working system to call processing quickly. In order to understand the details of the implementation of this philosophy, it is first necessary to examine the mechanisms for detecting troubles.

7.2.1 *Trouble Detection*

Each group of call processing orders (maximum of 17) to be sent to an operator position is loaded in a block of memory called a position information buffer (PIB) which is added to the linked list of PIBs of either half of the group gate for the operator group containing the position. Once every 25 milliseconds the PIB Execution Program (PIBE) gets an entry from PAFR in J level and distributes one order to each group gate half which has a PIB linked.

A distributed order to a group gate half normally results in an enable verify response from the enabled group gate half. Failure of the SPC to receive the enable verify results in an F level interrupt. The Central Pulse Distributor Fault Recognition Program (CPFR) passes to PAFR those failure reports that might be unique to a group gate. These might result from problems in the group gate itself, in the central pulse distributor (CPD), or peripheral unit address bus (PUAB).

Each sending group gate half (SGG) and receiving group gate half (RGG) has three master scanner points (F, S and T) which uniquely describe its state. When an SGG is enabled, its F point becomes a 1. If the order executes properly, the group gate is reset and the F point returns to 0. Failure of an SGG to be reset within 25 milliseconds indicates that the order failed.

All position subsystem circuits other than SGGs and RGGs transmit alarms through the alarm sender (AS) if abnormal circuit conditions are encountered. Every 10 ms the Group Gate Scan (GGSN) program scans, in J level, those RGGs whose information-present ferrods are set. If the RGG F point is a 1, GGSN interprets the data as an alarm and passes it to PAFR.

CPFR gives control in F level to PAFR to allow PAFR to save the data on the F level interrupt in memory dedicated for that purpose. Similarly GGSN enters PAFR in J level with alarms which PAFR saves in an area of memory called an alarm hopper. In both cases PAFR returns control after saving the data, and processing of the data is deferred to the next regularly scheduled entry to PAFR.

7.2.2 J Level Entry to PAFR

PAFR is entered from the executive control program in J level once every 25 ms. In general, the following operations are performed:

(i) A check is made that all SGGs are reset from orders sent by PIBE the previous cycle. If any failed to be reset, PIB execution is suspended on that group, and the failure is placed in the miscellaneous failure hopper for later fault recognition work.

(ii) RGG F points are checked for power off.

(iii) A test is made to verify that the SGGs can be reset if it was requested by the SGG diagnostic.

(iv) All SGGs are reset so that subsequent failures will be recorded.

(v) Control is given to PIBE to send another order to each group gate half which has an active PIB linked.

(*vi*) Continuing fault recognition work from a previous failure is resumed. The failing order is retried using various choices of bus, group gate half, and subsystem configuration. By analyzing the results, the faulty unit will be found. If at all possible, a hardware configuration will be established which will execute the order.

The handling of a failing order by PAFR actually varies according to a failure option specified in the PIB:

(a) option 0—if all efforts to execute the order fail, abandon the PIB and decrement the success address.

(b) option 1—if all efforts to execute the order fail, abandon the PIB, decrement the success address, and put the position in the "maintenance busy" state.

(c) option 2—make a single retry for an operate order only, decrement the success address if the retry fails, but allow PIBE to continue on with the PIB.

(d) option 3—ignore the failure and do no retries.

(*vii*) If fault recognition is not in progress, PAFR looks for new work: enable verify failure, teletypewriter request (Section 7.2.4), miscellaneous failure hopper entry, alarm hopper entry, or maintenance PIB request (Section 7.2.3).

PAFR must have complete control of a group before attempting fault recognition work. PIBE is stopped immediately from doing any more work on the group gate half which experienced the reset failure or enable verify failure. PIBE is also asked to stop work on the other group gate half as soon as the current PIB is executed. PAFR saves information about the PIBs in storage areas called "pseudo PIB," PIB-A for the failure half information and PIB-B for the good half information. After fault recognition is complete, PAFR uses this information to reestablish the PIB link lists as they were at the time of the order failure if the order failure was an error or to link the PIB to the working group gate half if one half contains a fault.

7.2.3 *Maintenance PIB*

PAFR provides special services for position subsystem diagnostic and exercise programs by administering a maintenance PIB. Up to 10 orders can be sent. Most of the time the maintenance PIB is executed concurrently with normal call processing PIBs. The results of each order are returned in the PIB to be interpreted by the program using the PIBs. The normal sequence is:

(*i*) Send the order with true parity.

(*ii*) Look for an enable verify failure.

(*iii*) Look for a change of FST points.

(*iv*) If no change, look for a reset failure.

(*v*) If a reset failure occurred, look for an alarm.

(*vi*) Store the result.

(*vii*) Repeat (*i*) through (*vi*) for each order.

One of 16 different variations of this sequence is selected on each order by specifying four control bits for the order, e.g., send bad parity, scan the diagnostic bus, or abort the PIB on a specific failure.

7.2.4 *Teletypewriter Requests*

Teletypewriter entries allow maintenance personnel to put a chain of position subsystem units (see Fig. 24) in service or out of service or operate transfer relays to switch unit halves between chains.

7.2.5 *Nixie Order Failures*

Because of the interdependence of orders in the order sequence required to establish a Nixie display, PAFR does not attempt to get failing Nixie orders to succeed. The operator will recognize that she does not have a correct display and request the display by depressing the time and charge key. If five Nixie failures have occurred within the hour, the failing chain is removed from service and diagnostics are requested on all units in that chain.

7.2.6 *Order Failures to the Service Observing Gate (SOG)*

The SOG is similar enough to a position group gate to be handled by PAFR in much the same way. If retrying the order over both buses does not successfully execute the order, the SOG half is removed from service and the maintenance personnel are notified via TTY message and the MCC lamp display. Because the loss of service observing does not immediately affect service and because the service observing system has its own established maintenance procedures, no automatic diagnostic is provided. However, a diagnostic procedure has been developed using the repetitive order sending program of the Position Access Exercise Program (Section 7.4).

7.3 *Position Subsystem Diagnostics*

The local and remote position subsystem hardware is designed such that the local subsystem can be considered a degenerate case of the

remote from the point of view of the diagnostic program designer. Therefore, the discussion which follows will be concerned with the remote subsystem shown in Fig. 24.

For units in the sending chains (SGG, DAC, PSD) input test access is obtained through the sending chains themselves. Maintenance circuitry is provided to return test results from each sending chain unit as independently as possible of other sending chain units. For units in the receiving chains (RGG, PSG), maintenance circuitry controlled by signal distributor applique relays and PSD applique relays is provided to generate input test sequences to each unit. Test results are returned over the receiving chains themselves. From consideration of paths traveled by test information, several conclusions can be drawn.

(*i*) It will not be unusual for a single fault to cause failures in diagnostic tests for more than one unit.

(*ii*) If several units in a chain fail diagnostics, the failure in the unit nearer the processor is probably more significant and should be repaired first by maintenance personnel.

(*iii*) A failure in a sending chain unit is probably more significant than a receiving chain failure and should be replaced first by maintenance personnel.

(*iv*) In designing tests for a particular unit, tests over shorter or more independent paths should be run first. This leads to diagnostic designs for sending units where input tests are run first, then internal tests and finally output tests. Similarly in the RGG diagnostic, tests using RGG test inputs are run before tests using PSG test inputs.

(*v*) In designing tests for a particular unit, it is frequently desirable to terminate the test sequence after a failure in a given group of tests. To continue with a test sequence dependent on circuitry already found to fail would result in inconsistent failure patterns and a dictionary number of doubtful value.

(*vi*) As a result of (*iv*) and (*v*), the first test failure in a diagnostic is probably the most significant and should be repaired first by maintenance personnel.

In all cases where a unit has two input or output routes, the same set of tests is run over each route. Where transfer relays are involved, a common set of tests is used for the output circuitry of one unit and the input circuitry of the following unit.

The T1 carrier and the PSG present unique problems. The sending

and receiving T1 carriers are standard Bell System equipments incorporated into the position subsystem. As such, they already have maintenance practices established for them. A Carrier Alarm Actions (CAAC) program has been written which supplements the existing maintenance procedures in four ways:

(*i*) provides teletypewriter reports of alarm status,

(*ii*) provides means for service removal and restoral of T1 data links as well as transfer of operator talking paths to the spare T1-B,

(*iii*) exercises transfer relays associated with operator talking paths, and

(*iv*) provides mechanization of a manual maintenance practice to isolate trouble to the local terminal, the remote terminal, or the interconnecting transmission line.

Because the PSG is an autonomous, asynchronous scanner on the remote end of the receiving chain, it is difficult to diagnose. Its diagnostic (PSGD) is relatively independent of other diagnostics, having its own control structure but sharing subsystem subroutines.

Diagnostic tests for the four remaining units, SGG, RGG, DAC and PSD are run under one control program which:

(*i*) provides common interface with the maintenance control program (MACR) and the dictionary number generation control program (DOCP),

(*ii*) initializes common scratchpad memory and critical subsystem maintenance circuitry,

(*iii*) assembles subsystem status information,

(*iv*) calls subroutines for the individual sequences of tests (called phases) for the unit being diagnosed,

(*v*) after completing a diagnostic, restores the subsystem to its state prior to the latest diagnostic but modified to reflect the results of that diagnostic, and

(*vi*) updates the master control center (MCC) lamp status.

Each subroutine for a phase initializes the circuitry to be tested prior to test to maximize the probability of obtaining consistent results in the presence of a fault.

7.4 *Position Access Exercise Program (PAEX)*

If a fault develops which is not detected by PAFR, the occurrence of a second detectable fault may create a situation in which a diag-

nostic will develop a dictionary number not appearing in the trouble location manual (TLM). This is because the TLM is generated by the insertion of single faults. Faults in circuitry providing infrequently used call processing functions or maintenance functions could cause this problem.

PAEX provides an automatic exercise which is entered once every 24 hours. All position subsystem diagnostics are run, supplemental tests are made, and the resulting data is analyzed. Any failure results in a TTY message indicating the faulty unit or failing circuit function. If all tests pass, the subsystem is reconfigured according to a predetermined sequence so that, over a period of time, units experience equal stress in different configurations.

PAEX also provides demand exercise functions to aid maintenance personnel in repairing subtle faults for which automatic maintenance is economically impractical.

(*i*) *Diagnostic Demand Program.* Any or all phases of a subsystem diagnostic may be requested once or repetitively with or without raw data printout by TTY request.

(*ii*) *Thumbwheel Switch Demand Program.* The DAC, PSD and PSG diagnostics can be requested with or without raw data printout from the thumbwheel switch located on the alarm sender frame. This allows maintenance personnel at a remote location to obtain data on a remote TTY.

(*iii*) *Remove/Restore Service Program.* This program lets maintenance personel put units in-service or out-of-service from the maintenance TTY.

(*iv*) *Repetitive Order Sending Program.* A request can be made from the TTY to have any order or combination or orders (10 maximum) sent once or repetitively to allow visual or instrument checking of circuit functions. The orders are sent via the maintenance PIB. The results from the most recent maintenance PIB execution are saved in a memory buffer. If the most recent result differs from the result obtained in the previous PIB, the number of the PIB, the number of the order, and the result are saved in a second memory buffer which has capacity for thirty such transient results. The transient buffer can be reinitialized from buffer bus keys on the MCC. The contents of either buffer can be read out on the TTY at any time. These features have been found to be particularly useful in diagnosing troubles resulting from component degradation causing marginal performance and intermittent failures.

7.5 *Position and Position Buffer Maintenance Program (PPBM)*

This program is designed to be a tool of maintenance personnel for maintaining the 100B console, the position buffer, and the position signal distributor applique circuits. In addition it controls the out-of-service state of positions with regard to signaling (not talking), insures that a position is initialized (only software) before being allowed to handle calls, and initializes a position (hardware and software) in response to a keyset request from the operator position.

PPBM will remove a position from service (*i*) if the position fails to initialize properly, (*ii*) if it is requested by a call processing program as a result of an order failure, (*iii*) if it is requested by maintenance personnel from the digiswitch, and (*iv*) if it is requested by an audit program. The out-of-service state is indicated to the operator by a flashing lamp. She then plugs out and moves to another position.

Plugging in to an out-of-service position by maintenance personnel puts that position in the maintenance mode. Any key code received from that position is passed to PPBM which then operates the position buffer relay to light the position lamp associated with that key. The operator keyset is used to input requests to test those relays, lamps and Nixie tubes which are not directly associated with keys.

If trouble with the position buffer relays is suspected, the relay card with the suspect relay is placed on an extender card so that the relays may be observed and the relays are tested by entering requests from the digiswitch. A digiswitch code is sent through the alarm sender and via the Group Gate Scan Program (GGSN) which recognizes it as an alarm to PAFR. PAFR translates the alarm, identifying it as a digiswitch code, and passes it to PPBM. PPBM translates the particular code and sends the requested order or orders. When the trouble has been isolated, the board is replaced, and the position is returned to service either by a digiswitch request or by a keyset sequence from the position.

The authors acknowledge T. F. Arnold and G. A. Backman, who developed the position subsystem fault recognition programs; R. N. Markson, J. J. Serinese and J. B. Seward who wrote the diagnostics; J. W. Herndon and L. E. Roberts who developed the exercise programs and T. P. McLoughlin who wrote the position maintenance programs.

VIII. TSPS SERVICE OBSERVING CIRCUITS

Service Observing (SO) is handled differently in TSPS than in other systems in that the information sent to the service observing operator

is taken from memory and is a repeat of the original signals. A special data link between the TSPS equipment location and the SO desk location is utilized for all signals except for the talking path. The talking path itself is also handled in a unique manner so as not to require additional trunk or link circuitry for service observing. The SO equipment is installed in the TSPS office and is subject to the same lead length restrictions on various inputs, but the drawings are part of a separate series used for traffic management and are not a part of the TSPS drawing series.

The service observing monitor circuit permits monitoring of the customer-to-customer talking connection. When it is known (by the SPC) that the next call (of a designated type) is to be observed, two link connectors are set up to reach an idle TSPS operator via an idle SO monitor circuit. An SO monitor circuit is furnished for each chief operator group and has appearances on both the trunk link and position link to permit this insertion into the talking path. The connection is shown on Fig. 25. Output from the high impedance connection provided by the monitor circuit is then sent over transmission facilities to a distant SO desk.
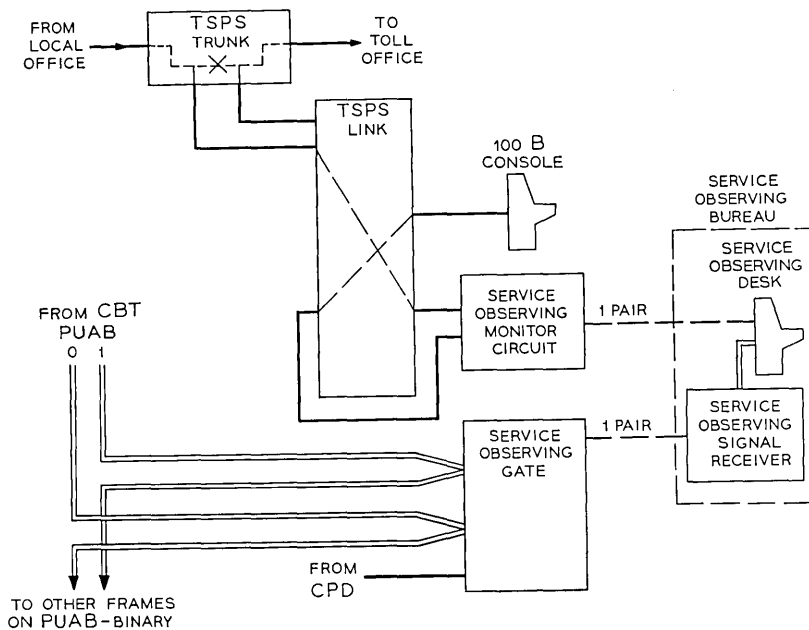
Fig. 25—Service observing data and transmission configuration.

All signaling, including dialed digits, ANI information, coin signals, etc., sent or received by SPC during the processing of the call is obtained from memory and transmitted to the SO desk to which the talking path connection was made. This is accomplished by means of the SO gate which connects to the PUA binary bus in much the same manner as a position group gate (Section 2.1). In fact the service observing gate (SOG) is treated by maintenance programs as another (the tenth) group gate. Only one (duplicated) SOG is provided per TSPS installation although up to eight SO monitor circuits may be provided.

Information received over the binary bus is temporarily stored and converted to 3 groups of 1-out-of-5 (1/5) codes. These 1/5 combinations are then sent as three frequencies over a single cable pair to the SO desk equipment. In practice, two transmissions are sent; and after reserving certain combinations for the idle (or rest) state, there are 1024 combinations available. This provides 128 combinations for each of the 8 possible SO monitor circuits and is sufficient to send all information required for service observing.

A few dc signals can be received over the pairs of wires used for the monitor circuits and these are recognized by the monitor circuits and coupled to ferrod scan points scanned by the SPC. This permits service observing operators at some remote location to indicate how many SO desks are manned and what type of calls are to be observed. Thus, observing can be concentrated on those types of calls felt to require special attention.

IX. CONCLUSION

The TSPS periphery combines the technology of existing developments, where applicable, with new designs unique to TSPS, as in the position subsystem. The necessary functions are provided in the most economical manner consistent with the needs of the Bell System for availability of TSPS and the state of the art at the time of the development.

X. ACKNOWLEDGMENT

Although the names of a few people have been mentioned, the design and development of the peripheral units and associated maintenance program required the efforts of many people in the TSPS laboratory and a number of resident visitors from Operating Companies and Western Electric.

REFERENCES

1. No. 1 Electronic Switching System, B.S.T.J., *43*, No. 5 (September 1964).
2. Durney, G. R., Kettler, H. W., Prell, E. M., Riddell, G., and Rohn, W. B., "TSPS No. 1: Stored Program Control No. 1A," B.S.T.J., this issue, pp. 2445–2507.
3. Connell, J. B., Hussey, L. W. and Ketchledge, R. W., "No. 1 ESS Bus System," B.S.T.J., *43*, No. 5 (September 1964), pp. 2021–2054.
4. Baldinger, W. R., and Clark, G. T., "TSPS No. 1: Physical Design," B.S.T.J., this issue, pp. 2685–2709.

TSPS No. 1:

Operational Programs

By A. W. KETTLEY, E. J. PASTERNAK and M. F. SIKORSKY

The Traffic Service Position System (TSPS) No. 1 Operational Programs provide the call processing logic which controls the handling of calls served by the system. The organization of these programs is described and the major events encountered in processing a sample call are covered in this article.

I. INTRODUCTION

The call processing programs of the Traffic Service Position System provide the logic and control for processing telephone calls. These programs supervise the calls, transmit and receive signals to and from other switching systems, send information to and respond to signals from operators, and record billing details on the calls. The programs that provide these and other functions are described in this article with emphasis on those areas that are new or substantially different from No. 1 ESS. To illustrate how the various programs interact and relate to hardware actions, a coin toll call is traced from origination to completion. At the end of the article some ancillary features are also described because of their novelty or because similar programs have not been previously described in the B.S.T.J.

Although the TSPS call processing programs are patterned after those of No. 1 ESS, the nature of the TSPS call dictates a number of changes in program and memory organization. Since TSPS handles toll traffic that enters the system via high-occupancy trunks, a software register is dedicated to each of the trunks to store the billing details on the calls as well as miscellaneous information required in handling the call. The high occupancy of these registers, which tends to minimize the memory-saving advantages of the traffic engineered ESS call registers, plus the simplicity of the control strategy for dedicated registers were important factors in the decision to use this

type of call register arrangement. Similar arguments apply to the position register, which is also dedicated. For the sake of uniformity all service circuit registers are similarly dedicated.

However, two other registers—the auxiliary trunk register and the path memory annex—are not dedicated. The auxiliary register is used to store details on coin calls, and the path memory annex stores information concerning a connection through the network. Since coin call details are not required on noncoin calls, and since TSPS network connections are made for a small part of the duration of a toll connection, both of these registers are engineered items and are linked to the trunk register as required.

Another difference between No. 1 ESS and TSPS is the significant amount of data exchanged between the system and the operators. This exchange is required, for instance, to set the stage for the operator on each call by identifying the type of call she is to handle and giving her other pertinent information about the call. The system in turn receives information from her as she operates keys and responds with some action to indicate the successful reception of her key signal. Since the operator has freedom to operate her keys in a variety of sequences for the same type of call, the programs that react to individual key operations are required to determine much about the nature of the call and its current state before selecting the appropriate course of action. It is also necessary to recognize invalid key sequences and to ignore them or to defer their recognition until some other action has taken place. These requirements considerably complicate the organization of the programs that respond to the operators key actions. In spite of widespread use of subroutines in this area, over 10,000 words of program are required for these functions alone.

One last major difference between No. 1 ESS and TSPS lies in the use of the network. Unlike the ESS network, the TSPS network does not switch calls but is used to attach the appropriate special-purpose circuit to the connection as needed. These circuits typically detect and identify the called and calling numbers, outpulse the called digits, collect and return coins, apply audible ringing, reorder, or a recorded announcement, and connect an operator as needed. The TSPS network requirements result in a simpler network than in ESS and a simplification in network control program strategy.

The programs associated with call processing may be classified in three categories: (*i*) input-output programs which detect and report changes in circuits outside of the processor and those which control

signals to similar external hardware units; (*ii*) call control programs which have only call-related functions and whose purpose is to advance a call to completion; and (*iii*) programs of a general nature which perform frequently used functions and which may be considered to be service routines used by call processing programs and others.

II. INPUT-OUTPUT PROGRAMS

2.1 *Input Programs*

The programs which are responsible for administering inputs to the system are designed to be relatively simple, highly efficient programs which run during H- or J-level interrupts and are responsible only for the acquisition of data. The information is passed on to other base level call programs responsible for the analysis of the input data. This method of operation is used because of the large number of input sources which must be scanned periodically and from which, on the average, a relatively small amount of data is obtained.

The trunk supervisory scan program is responsible for detecting changes of state in the supervisory ferrods of circuits on universal trunk frames. Because of the variety of trunk circuits found on a universal trunk frame such as incoming trunks, delay call trunks, and operator service trunks, a change of state in a ferrod can mean several things, depending on the trunk type with which it is associated. While it is not the responsibility of the supervisory scan program to determine the meaning of a change in state, it does separate the changes into two categories for purposes of reporting to other call programs. The changes from the on-hook to off-hook state are reported directly through the trunk seizure and answer hopper. The changes from the off-hook to on-hook state represent the beginning of a potential disconnect or flash. Consequently, the trunk supervisory scan program reports the latter change of state to a hit timing program which will time the duration of the on-hook signal to discriminate between flashes and true disconnects. The results of this timing are then reported to a call control program. It is the function of the call control programs receiving these reports to interpret the data and initiate the appropriate action.

The digit scan program scans the digit receivers and transmits the digits, as received, to the appropriate call control program. The call programs analyze the digits and determine when digit reception is complete and when to terminate scanning of a particular digit receiver.

The group gate scan program periodically scans ferrods associated

with the operator groups to obtain signals from keys depressed by operators at the traffic service positions. This data is passed to call control programs which interpret the key signal and take appropriate action.

The sender attached scan program scans the ferrods in the digit transmitters which indicate when a digit receiver has been attached at the toll office and is ready to receive digits. It reports this to a call control program which then activates outpulsing of the called number to the toll office.

2.1.1 Output Programs

When signals are to be sent from the processor to peripheral units, the system must buffer data to allow for the difference in speed at which the processor can process data and the speed at which the external circuits can operate. This buffering is accomplished by the output programs. The call control programs, which determine the control signals to be sent, store the data for these signals in various types of buffers which are then administered at the proper frequency by output programs.

In sending signals to control circuits such as trunks, link networks, etc., the call control programs load a peripheral order buffer (POB) with orders which are then distributed to the appropriate peripheral unit by the POB execution program. This program also reports to the call control program at a later time that the order was or was not successfully executed.

The call control program responsible for outpulsing digits to the toll office loads a register associated with the outpulser with a sequence of central pulse distributor addresses containing the called number. The output program associated with multifrequency outpulsing then distributes these at the rate of one digit every 133 milliseconds or 7.5 digits per second to the toll office.

The relays of the position circuits are controlled in a manner similar to those of trunks. The call control program loads a position information buffer (PIB) which is then executed by the PIB execution program.

2.2 Call Control Programs

Similar to No. 1 ESS, the programs which administer the processing of a call are separated into functions each of which is related to a stage in the progress of a call. This separation allows the programs to be of a manageable size and perform a well-defined function.

On a normal call, the programs responsible for handling the call are: (*i*) the call connections program, used to set up connections to the digit receivers, positions, and outpulsers; (*ii*) the digit analysis programs, used to record and analyze the digits received and determine the general type of call; (*iii*) the operator actions programs which are responsible for administering a call during the time it is connected to a position; and (*iv*) the disconnect program which controls the disconnect of the call, causes recording of call billing data, and returns the trunk to an idle state.

2.2.1 *Call Connections Program*

The call connections program is the first in the line of call control programs to handle a call. It receives the report from the supervisory trunk scan program that a trunk has been seized by the local office. To determine what must be done to serve this request, the program consults the parameter register associated with the incoming trunk.

Having obtained the necessary information, the program selects a network path and completes a connection over that path from the trunk to a digit receiver. In the case of multifrequency trunks, it also causes a signal to be sent to the local office indicating that a receiver is connected and transmission of digits can start. It finally activates digit scanning for this receiver. After digit scanning is initiated, the digit analysis program assumes control of the call.

At the completion of digit analysis, control is returned to the call connections program which determines whether automatic number identification (ANI) information is to be received from this office. If it is, the call connections program establishes a connection to a multi-frequency receiver if it is not already connected and generates a signal to the local office requesting ANI digit transmission. The ANI digit analysis program now assumes control until reception of the calling party's number is completed.

When the calling party identification has been received, control is returned again to the call connections program. At this time, a general analysis is performed on the information obtained. The customer dialed digits are checked and the call is marked as 0+, 1+, etc.

There are three basic call states at this time: position assisted, position assisted customer-dialed, and customer direct distance dialing (DDD). For the DDD call, a connection is established to an outpulsing circuit and the called number will be outpulsed forward. On completion of outpulsing, the call connections program establishes the talking connection through the trunk and administers supervisory

reports so as to establish answer. Upon receiving an answer report, the call start time is loaded in the trunk register and the register is placed in the appropriate talking state so that all further reports and actions on this trunk will be handled by the proper disconnect programs.

2.2.2 *Digit Analysis Programs*

The digit analysis programs are composed of three routines. Two are responsible for reception and analysis of the called party's number for multifrequency and dial pulse digit trunks. The third is for reception and analysis of the ANI digits transmitted by the local office.

2.2.2.1 *DP Digit Analysis.* The dial pulse digit analysis program counts and stores each digit as it is received in the register associated with the incoming trunk. For most digits no analysis is required, but in some situations an analysis is made of the digits that have been received. For example, when the first digit is received, it is checked to see if it is a zero or a one. Depending on the type of trunk, this could terminate dialing as a dialing error, be a valid condition, or institute timing to determine if more digits are to be received. Any other first digit is merely stored and the digit counter incremented by one. Upon receipt of the third digit, analysis of the first three digits is made. If the number 800 is received, it is known that the customer is dialing an INWATS call and that seven more digits are expected. If the first three digits are an office code in the area of the originating party, it is known that four more digits are expected. If the first three digits are a foreign area code, seven more digits are expected.

In the above instances, the number of digits to be expected is immediately evident and no more action is taken until the total number of digits expected are received. However, there are certain instances where a conflict must be resolved. For example, the first three digits dialed may be both an office code in the area serving the originating subscriber and a foreign area code. In this case, timing for 2.5 to 4.5 seconds is initiated after receipt of the 7th digit to determine whether more than seven digits are to be received. If an 8th digit is received before time out occurs, it is assumed that the foreign area code was intended and that a total of 10 digits will be received. If a time out occurs before the receipt of an 8th digit, it is assumed that the office code was intended and that all digits have been received and control is returned to the call connections program.

2.2.2.2 *MF Digit Analysis.* The multifrequency digit analysis program performs much the same analysis of the called digits but with a slightly different procedure. In transmitting the called digits by multifrequency pulsing, the local office sends a KP pulse at the beginning and a ST pulse at the end. Therefore, it is possible to wait until all digits are received and then perform the appropriate analysis, thereby avoiding in the case of conflicts the necessity to time for receipt of digits. Thus, when all digits have been received, the first three digits will again be analyzed, as in the case of dial pulse digits. However, if the first three digits are an office code, it is necessary only to check that seven called digits are received. In the case of a conflict, the resolution of the conflict is based again on the number of digits received. If a foreign area code were received, 10 digits are required. Although the originating office in this case must be a common control office and, therefore, should be screening the various illegal combinations that might be dialed by a customer, these checks are made as a safeguard against possible digit transmission failures which might have resulted in a partially valid check and have caused misrouting or failure in the toll network. Once a valid check of digits has been received, control is returned to the call connections program.

For multifrequency trunks the ST pulse that terminates called digit pulsing provides a traveling class mark. A discussion of the traveling class mark is given in the section describing ANI digit analysis.

For both dial pulse and multifrequency digits, the digit analysis programs are responsible for guarding against both invalid digit combinations and transmission failures. In the case where an invalid sequence of digits is received, as in a customer dialing error, control will be transferred to a program responsible for supplying an announcement to the customer informing him of this. In the case where it is apparent that a transmission failure has occurred, one of two courses is taken. For multifrequency trunks not carrying dial zero traffic and for all dial pulse trunks, the call will be routed to reorder. For multifrequency trunks where the trunk group can carry dial zero traffic, the call will be marked as dial zero and forwarded to the call connections program for operator handling as a dial zero call. The latter course of action is a safeguard against cutting off an originating office in the event of a wholesale failure of multifrequency senders in that office.

Both programs are also responsible for detecting calling party abandon during digit reception and transferring control to the appropriate call connections routine. Call connections will remove the

receiver and then transfer control to the disconnect program for idling of the trunk.

2.2.2.3 *ANI Digit Analysis.* The ANI digit analysis program is responsible for reception of the ANI digits from the local office. These digits identify the calling subscriber's line number for purposes of billing. No analysis, per se, is done on the calling subscriber digits themselves. They are merely recorded in the register associated with the trunk. However, certain information is transmitted to the TSPS through an information digit which is a part of the ANI digit transmission. The information digit informs TSPS of such things as identification failures in the local office, the fact that the originating line is a multiparty line and identification cannot be made, and whether or not the call has been locally service observed. In the case where the information digit indicates that identification was not made for either reason, the trunk register is marked so that when an operator is seized she will be informed that she must obtain the calling number and key this information into the system before advancing the call.

The ST pulse of the ANI transmission is used as a traveling class mark for dial pulse trunks carrying a mixture of one plus (station to station) and zero plus (operator assistance) traffic. Since the initial digit zero or one which is dialed by the subscriber is never seen by TSPS, the traveling class mark is used to indicate which digit was dialed. Also, for the case of trunks serving both coin and noncoin lines, the local office must forward information as to what type of line is being served on each call. This information is provided through one of four different ST pulses and is recorded by the ANI digit analysis programs in the trunk register. After receipt of the ST pulse, the ANI digit analysis program deactivates digit scanning and returns control to the call connections program. In the event of a transmission failure during ANI digit reception, the ANI digit analysis program will mark the trunk register to indicate an ANI failure, and then transfer control to the call connections program.

2.2.3 *Operator Actions Programs*

The operator actions programs are those which are responsible for a call while it is associated with an operator position. They may be divided into four groups, each having a separate function: (*i*) the key actions subroutines that are responsible for actions taken in response to a key operation on the position, (*ii*) the supervision program which is responsible for administering supervisory reports.

(*iii*) the monitoring programs which accomplish the duplication at the monitoring position of signals sent to a position being monitored, and (*iv*) the supervisor programs which handle the connections of a position to a supervisor console.

2.2.3.1 *Key Action Subroutines.* The key action subroutines handle all reports of keys operated at the position and must interpret these and respond appropriately to them depending upon the status of the call at the time the report is received. These programs are associated directly with the processing of a call. The strategy by which these programs are designed is based largely on the fact that there are few restrictions placed on the operator with regard to the sequence in which she may supply information for the handling of a call. For example, while she must provide all data necessary for handling and billing of the call before releasing the call, the order in which she supplies this data is not restricted. On a call in which she might have to supply the calling number, a class of charge and a billing number, the order of supplying these items may largely be determined by the situation she encounters. For example, a customer might announce he wants to make a credit card call and give his credit card number and then announce that it is to be person-to-person or vice versa. Therefore, each key on the position has its own program, the entry into which is independent of what has passed before. The logic of the program consists of determining what has passed before by interrogating data in the software register associated with the position. Each key operation, as it is received and acted upon, will set one or more status bits or bytes so that a record of status is kept. In this manner, each key program can determine exactly what actions are to be taken. For example, on a dial 0 coin call, coin rating will take place when both the called number and the class of charge have been received. Depending on which is received last, it would be either the start key (denoting end of called number keying) or the class of charge key which would obtain the rating. Both of these programs interrogate the coin rating status bits of the register to determine if rating has already been accomplished. These same coin rating status bits are set when a customer dialed call is received and rated prior to obtaining a position, thus making the checks in these programs actually independent of the type of call which originally reached the position.

It is also possible that the called number required manual rating either on a dial 0 or 0+ basis. In this event, an additional piece of information is required from the operator and that is the rate treatment number which she acquires from her rate schedules or from a

rate and route operator. The ST key program which terminates the rate treatment keying would then initiate the coin rating. In a similar manner, all the keys use the data recorded in the position register to determine their ultimate course of action and in this way are independent of the sequence in which the operator handles the call.

The various lamps on the position are used to communicate responses to the operator from the system. For example, a lamp under a key is usually lighted in response to a key action when it is accepted. The most common indication that the system has not accepted a key is either to flash the lamp or not light it at all. The flashing lamp usually indicates that the system is in a state to accept keys, but this particular key action is out of sequence or cannot be accepted for lack of data that should precede it. For example, if the operator were to depress the ST timing key without first having a class of charge, the ST timing key lamp is flashed to indicate missing data. A key lamp that is not lighted after a key operation is usually an indication that the system is unable to accept any keys at that time, usually because it is still acting on some previous key action. Certain keys also cause indications to be given on the operator's numerical display panel. Examples of numerical displays given to the operator are: charge and minutes on a coin call, a time display in response to the time key, and a called digit number display in response to the called number display key.

2.2.3.2 *Supervision Routines.* The supervision routines administer reports of changes in the switch hook state of the calling and called parties during the time that a position is associated with a call. The routines control the state of two lamps on the operators console that reflect the supervisory state of the customers and record information in the position register that indicates the state of these lamps. Under certain circumstances, the routines also record connect time in the trunk register.

When the call is active at the position (not placed in hold), reports of changes in supervision from the calling party are received by the supervision routines and they cause the state of the supervisory lamp to follow these changes. When the initial called party off-hook report is received during the establishment of the call, the supervision program initiates timing to determine if the off-hook is a true answer or the beginning of a busy or reorder. If a true answer is determined, the supervision routine extinguishes the called party lamp and records

the answer in the position register. If a busy or reorder is detected, the supervisory lamp remains lit, but the operator is able to hear the busy tone from the called party direction. After a true called party answer has been determined, control of the supervisory lamp is provided in the same way as for the calling party.

If the call is placed in hold by the operator, the initial off-hook report from the called party is handled as described above. A busy or reorder signal in this case would be heard by the calling party only with no change in state of the called party's supervisory lamp. Subsequent to a true called party answer the supervision routines would control the state of the supervisory lamp as described above.

With a held call in a position the supervision routines will initiate timing if an on-hook report is received from the calling customer. If a steady on-hook is detected, the supervisory routines will light the supervisory lamp steady. If flashing is received from the calling party, the supervisory lamp will be flashed at a fixed rate until further customer action is determined.

2.2.3.3 *Monitoring Programs.* The work of an operator at a position can be observed at a second position by the duplication of the lamp indications being sent in response to the actions of the operator handling the call. A special position equipped for this function is called the monitor position, and the monitoring program administers the duplication of the signals. While the displays at the monitor position are dependent on actions taken at the monitored position, the program is designed so as to be effectively independent and noninterfering with the programs actually handling the call.

The monitor position is activated by means of a key operated switch at the monitor position. In response to operation of this switch, the monitoring program displays the monitoring position number on the numerical display. This is an indication to the monitoring operator that her signal has been received and the program is now prepared to monitor whichever position she specifies. She operates two digit keys denoting the position number which she wants to monitor. The monitor program then checks the position register associated with the monitored position and updates the lamps on the monitor position to the present status of those at the monitored position. It also sets a bit in the monitored position register indicating that that position is being monitored. Subsequent signals sent to the monitored position are then passed to the monitoring position as a result of the

key action programs checking this monitoring bit. The monitoring routines return control to the key action routines in such a way that, to those programs, it is as if monitoring were not even taking place. Thus, the monitor program does not interact with the key operations nor in any way affects the processing of a call.

Once having established a monitor link with a given position, the monitor may observe as many calls from that position as she likes. If she desires to monitor another position, she merely operates her POS RLS key and keys in another two digits. If there were any displays at the monitor position pertaining to the previously monitored call at the time of POS RLS, they will be extinguished, and with the reception of the two digits the linkages will be reestablished to the new position as previously indicated. The only key actions which are accepted from the monitor position while it is in the monitoring mode are the digit keys as indicated above and the position release key. Any other keys which might be depressed while the position is in the monitoring mode are ignored by the system.

2.2.3.4 *Supervisor's Console Program.* TSPS provides for any operator to contact her supervisor or service assistant for assistance at any time by means of a supervisor (SR) key at her console. The service assistant answers such calls or may originate calls to any position in that group or to other selected points from a call-director-like console. Monitoring of the operator's voice connection is also provided from the supervisor's console.

Operation of the SR key at the operator's position causes a key lamp to flash at the supervisor's console alerting her to the waiting call. Depressing the key at the console completes a connection to the operator and steadies the flashing lamp. If the operator has a customer call in access, the supervisor's connection is bridged on the call, allowing a three-way conversation. Release from the connection can be effected by either the operator or service assistant.

The service assistant can originate a call to a position for talking or monitoring by depressing the talk key or monitoring key respectively and then keying a two-digit number to identify the position. For a talking connection the SR key lamp is flashed at the desired position under program control. The operator responds by depressing the SR key, at which time the lamp is steadied and the connection established. On a monitoring connection, a low-loss bridge is established on the operator's talking circuit with no indication to the operator. Release

from the monitoring connection is effected by reoperation of the monitor key at the supervisor's console.

2.2.4 Disconnect Program

2.2.4.1 *Call Talking States.* The processing of a call after it has been released from a position for the first time is determined in part by the call state that is established for the call at the time it is released from the position. After initial position seizure, the call state is determined by the actions taken by the operator and the states of the calling and called parties at the time the call is released from the position. The four major states are: (*i*) noncoin notify, (*ii*) coin paid, (*iii*) residual call, and (*iv*) general.

The noncoin notify state is established upon a customer's request to be notified at the end of a prescribed interval chosen by the customer. The operator sets the notify state by operating the KP NFY key and keying in the notify interval. After position release and called party answer, the call is entered into a timing list, and is subsequently handled by the disconnect program.

The coin paid state requires that a timing function be initiated at the time of position release. In this case the interval is established to cause the call to be returned to the disconnect program to collect the initial period coin deposit.

The residual call talking state is established for certain calls such as mobile, marine, and overseas calls that require connection to a cord board operator via the toll switching system serving the TSPS. Control over these calls lies with the cord board operator and a TSPS operator is never brought back on the connection.

With the exception of the residual type call, calls that are subsequently reconnected to a position after being released, retain their call state identity unless the operator takes some special action during the reconnection. Special action might be initiated by a customer's request to terminate or cancel the call and begin a new one. This situation can be handled by changing the call state identifier and treating this connection to the operator as an initial position seizure. Further action on the call is taken as if the call had reached the position as a new seizure.

The general state applies to calls which do not fit into one of the other three states. In this state the initial actions of the disconnect program are common for all calls in this state. However, as will be seen

later, there are some variations in the disconnect actions that apply to this state.

Once a call has reached the talking state, it is under the control of the disconnect program. Calls in this state are divided into the four major classes described above. These four classes of calls constitute the major legs of the disconnect program. A description of the actions performed by the disconnect programs for these calls is given below.

When a notify timeout is received on a noncoin notify call, the disconnect program obtains a position and, having established the connection to the position, transfers control to the operator programs once again at a point which signifies that this is a notify seizure. Should the customer disconnect prior to the timeout, the disconnect program proceeds to take normal disconnect actions in recording the call and releasing the trunk. If the customer flashes to recall an operator, the disconnect program establishes a connection to the operator. However, the operator programs are entered at a point which signifies a flashing recall and not a notify timeout. Also, certain data is recorded in the auxiliary register, which allows the operator programs to resume the notify timing at the point at which the flash occurred.

For coin calls, the disconnect program is responsible for handling the various initial period and overtime interval timeouts. Initially, the call is placed in a timing list whenever answer has been established, or upon release from the position, which provides a timeout 18 seconds before the end of the initial period. When this timeout occurs, the disconnect program establishes a connection to a coin control circuit and transfers control to the coin control programs which will execute a coin collect sequence and then return control to the disconnect program. At the conclusion of the coin collect sequence the call is returned to the timing list for the remainder of the initial period. At the end of the initial period, the disconnect program establishes a connection to an operator and transfers control to the operator programs at a point which signifies a coin notify seizure. Upon release from the position after the coin notify seizure, the call is placed in a timing list for a grace period of 6 seconds. If the customer disconnects before this grace period is up, the call will be disconnected with no further charging. If a grace period timeout occurs before disconnect, overtime timing is continued. The disconnect program receives timeout returns from the timing program for each ensuing overtime interval. Each time such a timeout occurs, a counter

in the auxiliary register is incremented to keep track of the number of overtime intervals for charging purposes. A maximum of 10 overtime intervals can pass before any system action is taken. When the 10th overtime interval timeout occurs, a position will be seized and a charge due seizure indicated. If the customer again remains off-hook after the position releases, a similar timing cycle is started again. This will be repeated for each 10 overtime intervals as long as both parties remain in the off-hook state. During any of these overtime cycles, if the customer disconnects or flashes, indicating to the system that he has completed his call, a position will be seized for a charge due request. However, in this case the forward connection is released as it is assumed that the call has been terminated. When the operator releases the call, control is transfered from the operator programs to a point in the disconnect program which will record the billing information and restore the trunk to the idle state.

For a call in the residual call state, disconnect actions are under joint control of the operator and the calling customer. The disconnect program, upon receipt of a flash or a disconnect from the calling customer, does not take disconnect actions but relays the signal to the cord board operator and waits an indication from her as to the disposition of the call. Should the called end go on-hook at this time, indicating that the cord board operator has released the connection, the disconnect program proceeds with the usual disconnect action. However, if a wink is received, the disconnect program attaches an MF receiver to the outgoing side of the trunk to receive inband signals which must then be repeated via a coin control circuit to the local office. The signal could be coin return, coin collect, or ringback. If an on-hook is received from the outgoing side of the trunk first, the disconnect program waits to receive a disconnect from the incoming side of the trunk before taking disconnect action.

All other calls in a talking state not specifically of the types already discussed are administered by the general talking state leg of the program. If an on-hook is received from the calling side of the trunk, flash timing will always be performed to discriminate between flashes and true disconnects. If a disconnect is established, a billing record will be made, the trunk connections released, and the trunk restored to the idle state. If a flash is detected, the disconnect program checks the talking state index to see if this type of call has flash recall allowed. If it has, a connection is established to a position and control returned to a point in the operator program which indicates a flashing

recall. If flashing recall is not allowed on the call, the flash is ignored and the call returned to a talking state. At the time a disconnect on a call in this category is detected, the disconnect program also determines if it is necessary to collect or return a coin. If collect or return is required, it will perfrom the function before completing the other disconnect actions.

2.2.5 *Other Call Processing Programs*

There are several additional call processing programs which do not directly control the processing of a call but which are significant attributes of TSPS No. 1. Among these are the coin rating and computing programs and the service observing program. A brief description of these programs is included here for completeness.

2.2.5.1 *Coin Charging Programs.* One of the types of calls the TSPS No. 1 must handle is the coin paid call. A major advantage of TSPS over cord boards is that the initial period and overtime charges on these calls can be automatically computed and displayed to the operator, saving her time and effort and providing a more accurate result than humanly possible. These functions are performed by two programs, the coin rating program and the coin charge computing program.

The cost of a coin call is based on several factors: (i) the appropriate rate schedule, (ii) person or station class of charge, (iii) distance, (iv) time of day, and (v) possible reduction of rates for certain holidays or days of the week. The rate schedules are a means of dividing the rates which might be applicable to a call into categories based on the relation of the originating point to the terminating point. Calls which originate and terminate in the same state are subject to intrastate schedules. Calls which originate in one state and terminate in another are subject to an interstate schedule. Calls which originate within the continental boundary of the United States and terminate outside the continental boundary are subject to international schedules. TSPS No. 1 allows for up to five schedules to be rated in any given installation in any combination of the following:

(i) one interstate schedule,
(ii) canadian schedule (international),
(iii) mexican schedule (international), and
(iv) three intrastate schedules.

A numerical index has been established which defines the charges applicable to a call, accounting for the aforementioned variables.

This index is referred to as the rate treatment number (RTN). The RTN can be broken down into three components: (*i*) that portion which defines the applicable schedule, (*ii*) that portion which indicates that person or station rates apply, and (*iii*) that portion referred to as the rate treatment index (RTI), which is an index into a table defined by (*i*) and (*ii*) containing information in the form of charges and minutes for the initial and overtime periods. The process of rating a call, then, is one of determining the proper RTN.

The rate schedule which applies is determined by interrogating memory relating to the originating and terminating points and arriving at one of the schedules defined above.

The person or station status is indicated to the coin rating programs by the program requesting the rating. In the case of customer dialed calls where rating is done prior to seizing a position, the class of charge is inferred from the prefix dialed by the customer. If a 0+ call has been dialed, it is assumed that person rates apply. If a 1+ call has been dialed, it is assumed that station rates apply. Requests that are generated while a call is connected to a position use the class of charge entered by the operator to establish this status.

In order to derive the RTI, a rate line is first determined. A rate line defines a set of rates applicable to a call and is based on the distance between originating and terminating points. The way in which this item is obtained varies depending on how the numbering plan area (NPA) containing the terminating central office (TCO) is rated.

To economize on memory requirements for coin rating, a given TSPS installation may not rate all NPAs. In general, those NPAs which receive the largest volume of traffic will be rated. If the NPA is not automatically rated, the call will be taken to the operator with a manual rate indication. The operator will then obtain the RTN via her bulletins or from a rate and route operator and key it into the system.

An NPA which is rated falls into one of two categories, a single rate area or a multirate area. A single rate area is one such that a single rate line applies to all calls terminating in it from the TSPS installation in question. If this is the case, the rate line is determined immediately. For multirate areas the rate line must be determined based on the TCO code and the originating NXX code. The TCO code itself can fall into one of four categories: (*i*) vacant (probable dialing error), (*ii*) manually rated, (*iii*) V and H ratable, and (*iv*) exception. The first case would be routed to reorder if it were customer dialed, and the second would be taken to an operator with a manual

rate indication as before. The last two will be described briefly below.

The V and H system is a coordinate system superimposed on the United States, Canada, and Mexico. The V stands for the vertical coordinate and the H stands for the horizontal coordinate. By using the location of the originating and terminating points specified in the system, a measure of the airline distance between the points is calculated. This measure is then used to examine a table (for the schedule which applies) to obtain the rate line.

For the exception case, the rate line is determined by a direct look up method. Instead of finding V and H vector data, an address is found of a location which contains the rate line.

Having obtained the rate line by one of the above methods, the RTI can then be calculated. This is accomplished by applying the remaining variables of time of day, holiday or day of week considerations. These variables are determined through a series of table look ups. The three components of the RTN are then determined and the initial period and overtime charges can be determined. In the case of a manually rated call, the operator supplied RTN is identical to what would have been obtained had the call been automatically rated. In either case, the RTN is stored in the auxiliary register for subsequent use.

To obtain the initial period (IP) charge and minutes, the RTN is used to obtain data from a table for the specified rate schedule. This data consists of: (*i*) the IP charge after tax and rounding, (*ii*) timing information, (*iii*) the IP charge before tax and rounding, and (*iv*) indicators as to what tax rates apply. The first two items are used to provide the display to the operator. The first three items are stored in the auxiliary register to be used as a basis for subsequent calculations if the call goes into overtime.

When an overtime calculation is to be made, the coin computing program retrieves the RTN from the auxiliary register and, again the same data table for the specified rate schedule is read. This time, however, the overtime charge is obtained. Using the count of elapsed overtime intervals that are supplied by the client program, the overtime charges are calculated and returned to the client.

In performing the rating and computing functions, the coin programs provide considerable flexibility in allowing for a variety of federal, state, and municipal tax structures, with the calculated amounts rounded to the nearest nickel. Considerable flexibility is also allowed in making changes in coin rating and computing data. The addition

of a new office in an area rated by a TSPS installation requires that
new data be added in memory. A rate change also affects the contents
of memory depending on the extent of the change. These and other
modifications to coin data can be made by means of recent change
techniques or via a magnetic tape that can be prepared for massive
changes well in advance of their introduction. (See Section VII, Re-
cent Changes, and Section VIII, Program Tape Unit Control.) Hence,
the operating company can introduce changes in coin rating with ease.

2.2.5.2 *Service Observing Program.* The function of a system such as
TSPS No. 1 is, of course, to provide a good quality of service to the
telephone customer. Therefore, it is necessary to provide a means for
determining that the service being provided is acceptable by the stan-
dards which have been set for it. One of the means by which the service
can be measured is through service observing. This is a feature whereby
both the system and the operators are monitored to see if the standards
are being met.

The service observing function in TSPS has been implemented so as
to allow observing on any trunk in the system and any operator con-
nection which is established to that trunk. At a point in the call con-
nections program just prior to seeking an idle position to serve a call,
control is transferred to the service observing program. At this point,
the service observing program performs a directed scan of a control
ferrod to establish whether service observing is in effect for the sys-
tem at that time. If it is, a second check is made to see if the class of
call which is being handled is a candidate for observing. If either of
these checks are negative, control is returned to call connections pro-
gram and the call is handled in the normal manner. If, however,
service observing is to be made on this call, the service observing pro-
gram will then secure the next idle TSPS position. At that time, a
third check is made to see if that particular chief operator group is
being observed. If it is not, control is again returned to the call con-
nections program and the call handled in the normal manner. If all
the checks pass, the call is to be observed and the service observing
program will reserve a path to the operator. However, instead of con-
necting directly from the trunk to the operator cut-through circuit,
it will connect via a service observing monitor circuit which has a
double appearance on the network. The circuit has one appearance on
the trunk side and one on the position side as shown in Fig. 1. At
this time, it sends a seizure signal via the monitor circuit which causes
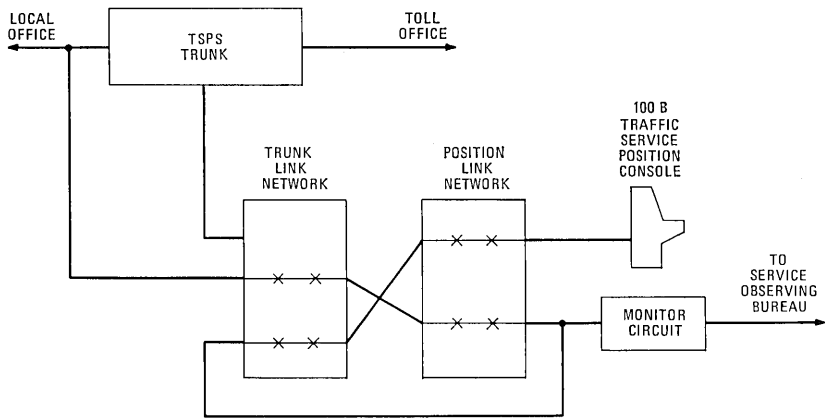
Fig. 1—Insertion of service observing monitor circuit into position connection.

any idle service observing desk to be associated with this monitor circuit. When this connection is successfully established, information relating to the call such as calling and called number, chief operator group and position number, etc., is transmitted to the service observing desk. When this initial data transmission is complete, the connection to the operator is established and control transferred to the operator programs at the initial position seizure point. The trunk register associated with this call has been marked to indicate that the call is being service observed. All subsequent actions on the call for which indications must be sent to the service observer check for this mark and, finding it set, transfer control temporarily to the service observing program which transmits data to the service observer. Control is returned to the programs at a point such that they complete their functions as though the service observer were not attached. Thus, interfacing with service observing is accomplished with no interference to the normal processing of the call except for the delay in sending the initial set of data to the TSPS position.

The service observer may release herself from the connection at any time. When the service observing program detects the release signal from the observer, it releases the service observing desk and administers the network connection to the monitor circuit according to the state in which the call is established. If the call is still connected to a position, for example, the connection to the monitor circuit cannot be removed since this would disrupt the connection to the operator. In

this case, the service observing program marks a bit in the trunk register indicating that the service observer is no longer attached. Upon position release, the position release key program then releases all the connections associated with the service observing monitoring circuit. If the service observer is still connected at the time of position release, the position release key program recognizes this condition and releases the operator cut-through circuit connection but retains the connection through the network to the service observing monitor circuit. If the service observer release signal is received when the call is no longer connected to a position, the service observing program releases the connection to the monitor circuit and erases the indications in the trunk register indicating that the call is being observed. Further handling of the call after this proceeds without any interfacing with service observing.

III. TEMPORARY MEMORY

During the processing of a call, numerous data is transmitted to and from the various input-output and call control programs. This data is kept in temporary memory associated with the various call processing or service routines. The recording of information in temporary memory is the means by which the various parts of a call are linked together and continuity is maintained.

3.1 *Input-Output Oriented Memory*

Similar to ESS No. 1, each scan point of a trunk has two bits of temporary memory associated with it. These two bits are used to indicate: (*i*) that the facility is idle and may originate a request for service; (*ii*) that the trunk is in the talking state and being monitored for disconnect; or (*iii*) that the scan point state is being monitored by another means temporarily, and that the supervisory trunk scan should disregard any changes that occur.

The receiver scanning program has several blocks of temporary memory to administer scanning of receivers for digits. One block is used to store signal present indications from the receivers. Another block contains receiver activity information that indicates that a receiver is connected to a trunk and ready to receive digits.

Another register associated with input programs is the timed scan junior register which is a block of memory used in timing on-hooks to filter out hits from true disconnects. It is also used in performing timing to detect calling party flashes and called party disconnects.

A third register, the flash scan timing register, performs a similar function for the called party side of the trunk to distinguish between answer and off-hook flashing, indicating the return of a busy, reorder, or circuit busy signal from the toll end of the trunk.

3.2 *Hoppers*

As was mentioned in the description of the input-output programs in Section 2.1, input data to the system is obtained during H- and J-level interrupts and is analyzed by ground level programs. Hoppers are used to buffer the data between the input programs and their base level processing programs with the data served on a first-in—first-out basis. Each input program has one or more hoppers by which it relays information to base level, and each entry contains some identification that associates the data with a particular call. The data and its identification varies with the function performed by the program. For example, the supervisory trunk scan program reports an off-hook in the trunk seizure and answer hooper by loading the trunk scan number which identifies the trunk on which the change of state has occurred. In the case of digit scanning the trunk register address is loaded in the digit hopper along with the digit received.

3.3 *Output Buffers*

Data that is generated by base level programs for execution or transmission by interrupt level programs to the peripheral system requires output buffers. Peripheral order buffers (POBs) are the principal means by which information is buffered before being sent to units such as networks and signal distributors. Each POB is a fixed length table with space reserved for call identification, an address to which control is to be returned after the data has been transmitted, and the data itself. The number of these buffers is traffic dependent and must be engineered according to the needs of each office.

Similar information that is sent to the position subsystem is buffered by position information buffers (PIBs). The arrangement of the PIB is similar to that of the POB except that PIBs are designed to store orders whose format is peculiar to the requirements of the position subsystem. PIBs must also be engineered for each office. Other buffer areas include those for buffering digits to be outpulsed, TTY characters to be sent to teletypewriters, and call billing information destined for the AMA tape but unlike PIBs and POBs are not engineered items.

3.4 *Call Control Registers*

Call control registers are blocks of unprotected memory that are used to record transient information about a call at various stages in the handling of the call. As was pointed out in the introduction, TSPS differs from ESS No. 1 in the provision of these registers. In general, the TSPS registers are dedicated to the circuit with which they are associated in the processing of a call. The format of these registers is standardized although their size and information content may vary, and to conserve memory some areas of registers may serve different functions at different phases in the call.

The standard format of these registers defines the first five words of all call registers. Included in the information stored in these words are the paramater register address, a link word used to link other service registers, two queue words to link the call register to various link lists, and a call state identifier word to define the state of the call. Additional space may be required in some call registers for more information, and this space uses standard layouts as much as possible.

Briefly, the three major call registers are the trunk register, position register, and coin control trunk register. The trunk register is used to store control data and all of the billing information with the exception of coin charging information, credit card number and charge to third number. The position register stores control data on all calls in access or hold on the associated position. This data includes memory of what keys have been operated and what lamps lit. The third register is the coin control trunk register and is associated with the coin control circuit. Unlike other service circuits, this circuit was given a call register to vest control of the coin actions (collect, return, and ringback) in one program.

3.5 *Service Registers*

In addition to the call registers described above there are a number of registers associated with service circuits or service routines used in processing a call. Service registers provide dedicated memory for MF receivers, DP receivers, outpulsers and the like with a standard format employed in the first three words. In these words is stored the address of the associated parameter register, a link word to connect the service register with the call register, and several status bits related to maintenance states. Again, some registers such as the MF outpulser register require added memory for storing additional information.

Other service registers associated with certain program functions include the time scan junior register and the flash scan junior register, the path memory annex, and the auxiliary trunk register. The time scan and flash scan registers are similar to those in ESS No. 1 and are used for detecting answers, disconnects, hits and flashes. The path memory annex is linked to a call register during a network connection to store the identity of the connected circuits, the paths used, and their status. Finally, the auxiliary register is linked to the trunk register to provide additional storage for coin paid charges, the third party number on charge to third party calls, and credit card and other special billing information. The numbers of these service registers are engineered according to the needs of each office.

IV. PROCESSING A TYPICAL CALL

The following is a description of a typical 0+ coin call from a Step-by-Step office which illustrates many of the actions that take place in handling a call with TSPS.

4.1 *Detection of Origination*

Figure 2 portrays the memory and program associated with the detection and processing of an origination. The supervisory trunk scan program detects a change of state in the ferrod of the incoming side of the trunk and enters the trunk scan number into the trunk seizure and answer hopper. The trunk scan program scans MF trunks at a 200 millisecond rate and DP trunks at a 100 millisecond rate. Thus, detection of an origination is at most 200 milliseconds after
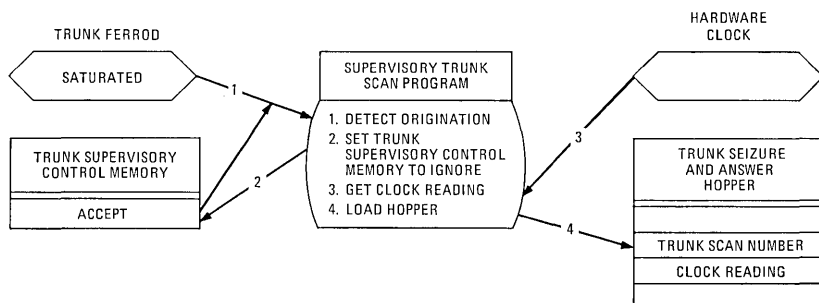
Fig. 2—Detection of origination.
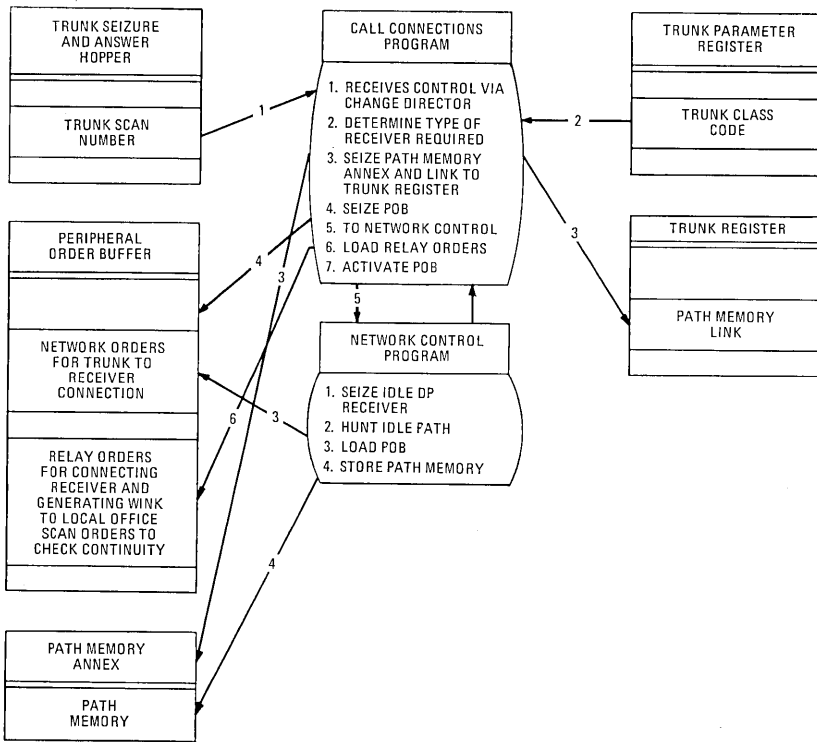
Fig. 3—Initial actions for connection of trunk to digit receiver.

the time the trunk is seized for the MF case and 100 milliseconds for the DP case.

4.2 *Connection of a Trunk to a Digit Receiver*

The initial actions of the call connections program are illustrated in Fig. 3. The change director program, entered periodically from executive control, unloads the trunk seizure and answer hopper and via translation determines that the scan point is associated with an incoming trunk. It thereupon executes a transfer via the state word of the trunk register. The state word of the trunk register directs transfer of control to the originating portion of the call connections program. At this point the call connections program interrogates the parameter register associated with the trunk to determine the various pieces of information necessary for connecting the appropriate type

of receiver. The call connection program then seizes an idle POB and a path memory annex. Control then transfers to the network control program along with information defining the type of receiver to be connected and identifying this particular trunk. The network control program loads the POB with orders to connect the trunk and the receiver and records the corresponding linkage data in the path memory annex. The call connections program also loads the POB with the appropriate relay and scan orders associated with establishing a connection to the receiver. After loading orders in the POB, the call connection program then activates the POB. As shown in Fig. 4, the POB execution program causes the orders in the POB to be executed. After successful execution of the orders loaded in the POB, the connection as shown in Fig. 5 is established. Also, the relay orders in addition to establishing the facility connection as shown generate a signal to the local office that the TSPS is ready to receive digits. Upon completion of the execution of the POB (as shown in Fig. 4), control is returned to the call connections program which then idles the POB and activates the receiver by setting a bit in the receiver scan activity control memory. This causes the digit scanning program to detect and transmit digits as they are received via the digit hopper to the digit analysis program.

4.3 Digit Analysis

4.3.1 Reception of Digits

As shown in Fig. 6, the receiver scan program detects the appropriate change of state in the digit present ferrod of a receiver denoting that a digit is present. It then scans the digit ferrods of the receiver and stores the digit along with the trunk register address for this trunk in the digit hopper.
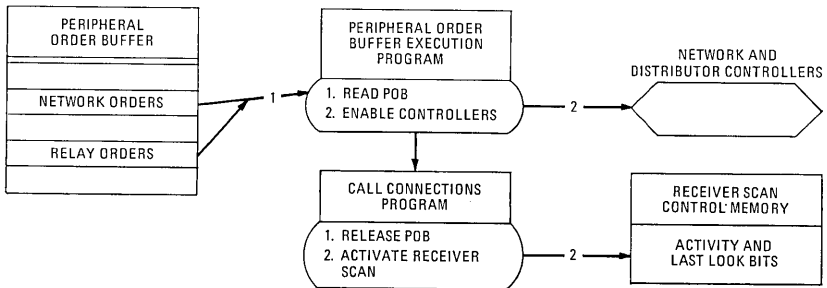
Fig. 4—Actions for connection of trunk to receiver.
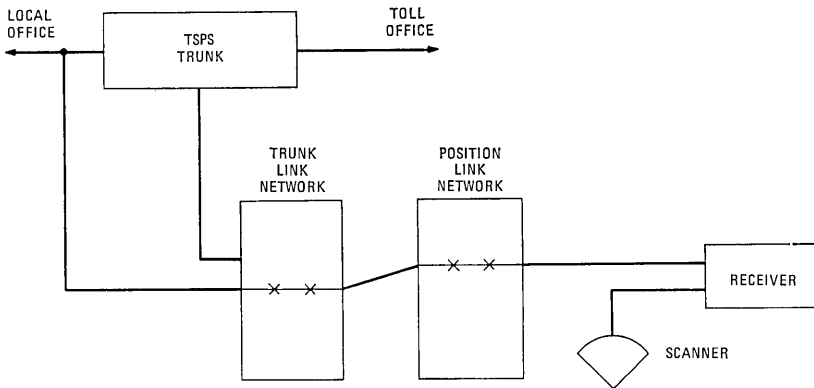
Fig. 5—Receiver connection.

The digit analysis program, which is shown in Fig. 7 and which is scheduled by an executive control program, treats the entry in the hopper, stores the digit in the trunk register, and increments the digit reception counter. When the digit analysis program recognizes that the third digit has been received, it transmits these first three digits to the called digit translation program which returns information for the analysis of the dialed digits. The type of information returned here defines the first three digits as being an area code, an office code, an invalid code, etc. For our example it will be assumed that the digits are recognized as a foreign area code which means that 10 digits are ultimately to be received. For this case the digit analysis programs marks the trunk register to indicate that no further action is required until receipt of the 10th digit.

4.3.2 Completion of Digit Reception

In the case of the 10 digit call (as shown in Fig. 8) when the digit analysis program determines that the 10th digit is received, it shuts off the receiver scan program for this particular receiver, marks the register as dialing complete, and returns control to the call connections program.

4.4 Reception of Calling Party Identification

4.4.1 Establishing the Receiver Connection

Upon receipt of control from the digit analysis program, the call connections program again interrogates the parameter register of the
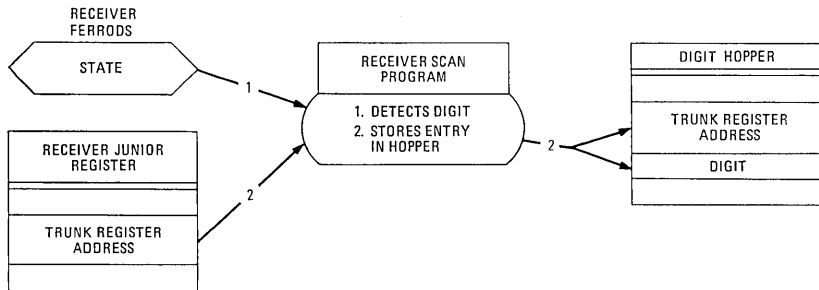
Fig. 6—Reception of digits.

trunk to determine if automatic number identification (ANI) is pro-
vided for this office. For our example it is assumed that the local
office is an ANI office.

In this case, the call connections program seizes a peripheral order
buffer and again transfers control to the network control program
which loads orders to break the connection between the trunk
and the dial pulse receiver and to establish a connection to a multi-
frequency receiver for reception of the ANI digits. These actions are
shown in Fig. 8. At the completion of the loading of the POB the
call connections program activates the POB.

After the POB execution program successfully completes the execu-
tion of the orders in the POB (Fig. 4), control is returned to the call
connections program.

The successful execution of the POB actions establishes an identical
connection as shown in Fig. 5 except an MF receiver is now connected.
The relay actions in this POB also generate a signal to the local office
to indicate that the TSPS is now ready to receive the ANI digits. The
call connections program (as shown in Fig. 4) then idles the POB
and activates the receiver scan program to scan for and transmit to
the ANI digit analysis program via the digit hopper the digits as
they are received.

4.4.2 *Reception of ANI Digits*

As shown in Fig. 6 the receiver scan program again periodically
interrogates a signal present ferrod of the receiver to detect when a
digit is present. Upon detection of the appropriate change of state in
the signal present ferrod of the receiver, the receiver scan program
scans the tone ferrods of the receiver and stores this information along
with the trunk register address in the digit hopper.

DIGIT HOPPER

TRUNK REGISTER ADDRESS

DIGIT

DIGIT ANALYSIS PROGRAM

1. READ HOPPER
2. INCREMENT DIGIT COUNTER
3. STORE DIGIT
4. OBTAIN THREE DIGIT TRANSLATION
5. SET DIGIT ANALYSIS DATA IN TRUNK REGISTER

TRUNK REGISTER

DIGIT COUNTER

CALLED DIGIT STORAGE

DIGIT ANALYSIS DATA

TRANSLATION PROGRAM

CALLED DIGIT TRANSLATION
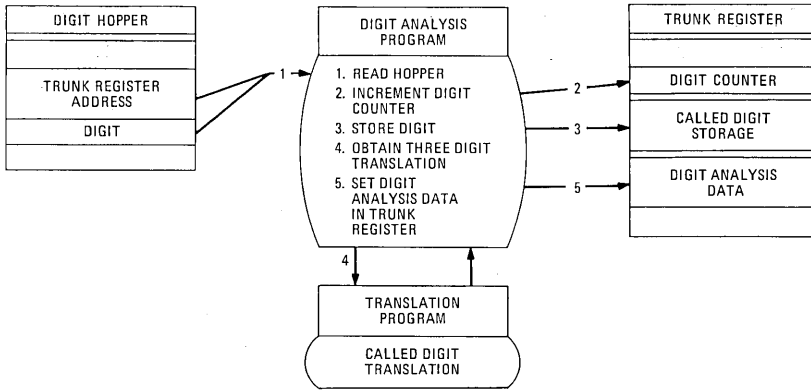
Fig. 7—Analysis of third digit.

DIGIT HOPPER

TRUNK REGISTER ADDRESS

DIGIT

RECEIVER SCAN CONTROL MEMORY

ACTIVITY BITS

PERIPHERAL ORDER BUFFER

NETWORK ORDERS TO RELEASE DP RECEIVER AND CONNECT MF RECEIVER

RELAY ORDERS TO GENERATE ANI REQUEST TO LOCAL OFFICE

DIGIT ANALYSIS PROGRAM

1. DETECT TENTH DIGIT
2. STORE DIGIT
3. STOP RECEIVER SCAN
4. MARK DIALING COMPLETE IN TRUNK REGISTER
5. TRANSFER CONTROL TO CALL CONNECTIONS PROGRAM

CALL CONNECTIONS PROGRAM

1. DETERMINE IF ANI IS TO BE RECEIVED
2. SEIZE POB
3. TO NETWORK CONTROL
4. LOAD RELAY ORDERS
5. ACTIVATE POB

NETWORK CONTROL PROGRAM

1. SEIZE IDLE MF RECEIVER
2. HUNT IDLE PATH
3. LOAD POB
4. UPDATE PATH MEMORY

TRUNK REGISTER

CALLED DIGIT STORAGE

DIGIT ANALYSIS DATA

TRUNK PARAMETER REGISTER

TRUNK CLASS CODE

PATH MEMORY ANNEX

PATH MEMORY

Fig. 8—Actions upon receipt of last digit.

The ANI digit analysis program obtains the ANI digits from the hopper and performs the appropriate analysis actions. The first digit which is received must be a KP pulse to insure that none of the first digits have been missed. This is not stored in the register, but the fact that it has been received is indicated in the control words used by the analysis program. The next digit received is the ANI information digit. This is analyzed and depending on its value certain information may be recorded in the trunk register. For example, the information digit may indicate that the call has been locally observed. If so, a service observed mark in the trunk register is set. This digit may also indicate that the local office is unable to identify the calling party due to an equipment failure or to the fact that the originating party has a multiparty line such that his number cannot be determined. In this case bits in the trunk register are set to indicate that operator identification must be requested and which type of indication is to be given to the operator. This would also indicate to the program that a 7-digit line number is not to be expected. In the example being used here of a dial pulse office, the last signal to be received, the start (ST) signal, is the traveling class mark. There are two possible ST signals that may be received—one indicating that the customer had prefixed the number he dialed by a one, the other indicating that he prefixed his number with a zero. In this example, we assume that the customer prefixed a zero. Upon receipt of the start signal, the receiver is deactivated and control is transferred to the call connections program.

4.5 *Establishing a Connection to a Position*

At this time the call connections program (as shown in Fig. 9) examines the data recorded in the trunk register and determines that a position should be attached to serve the call. Before proceeding to seize a position, the coin rating data is obtained and stored in the auxiliary register. It then seizes an idle POB and transfers control to the network control program which loads the POB with orders to establish a connection to both a position and an outpulsing circuit. If either of these circuits is unavailable, the call connections program queues until both circuits are available. The call connections program loads the orders to perform the appropriate relay operations required to complete the connection and alert the operator with a zip tone when she is connected. After completion of the loading, the call connections program then activates the POB.

As shown in Fig. 10, the POB execution program causes the orders
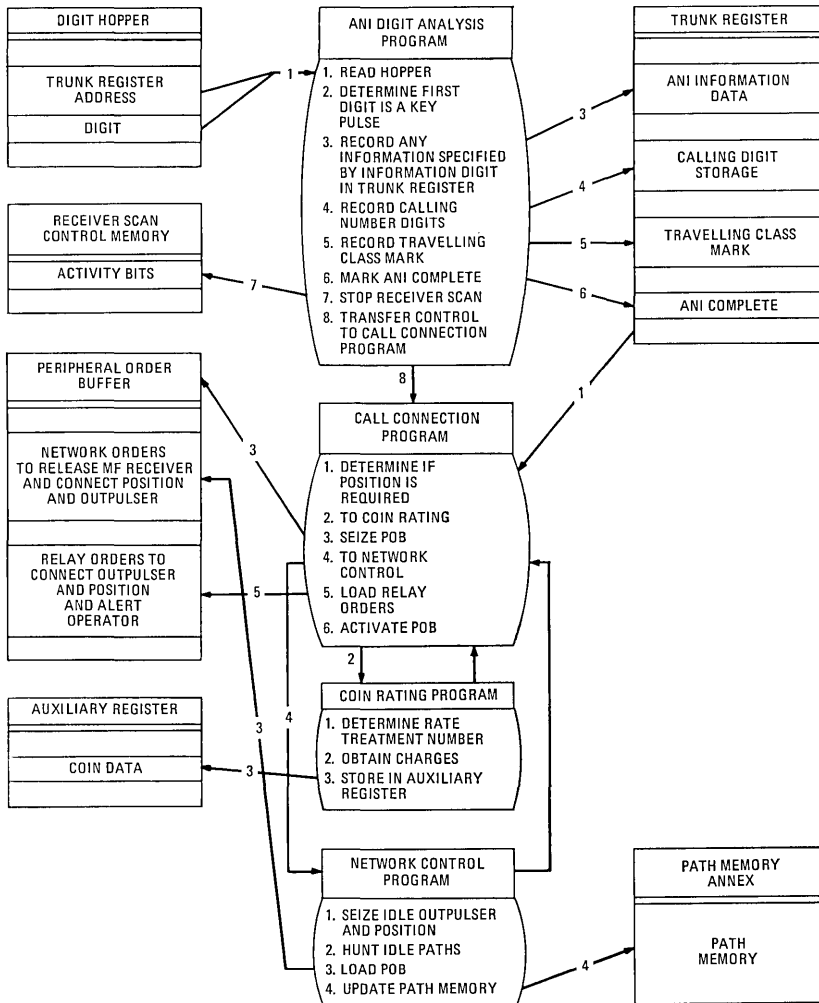
Fig. 9—Actions for receiving ANI digits and seizure of position and outpulser.

in the POB to be executed. The successful execution of these orders results in the configuration shown in Fig. 11, where the position is connected to the calling customer side of the trunk and the outpulser connected to the toll side of the trunk. Upon completion of the POB, control is returned to the call connections program. The call connection program idles the POB, transfers to the outpulser loading routine
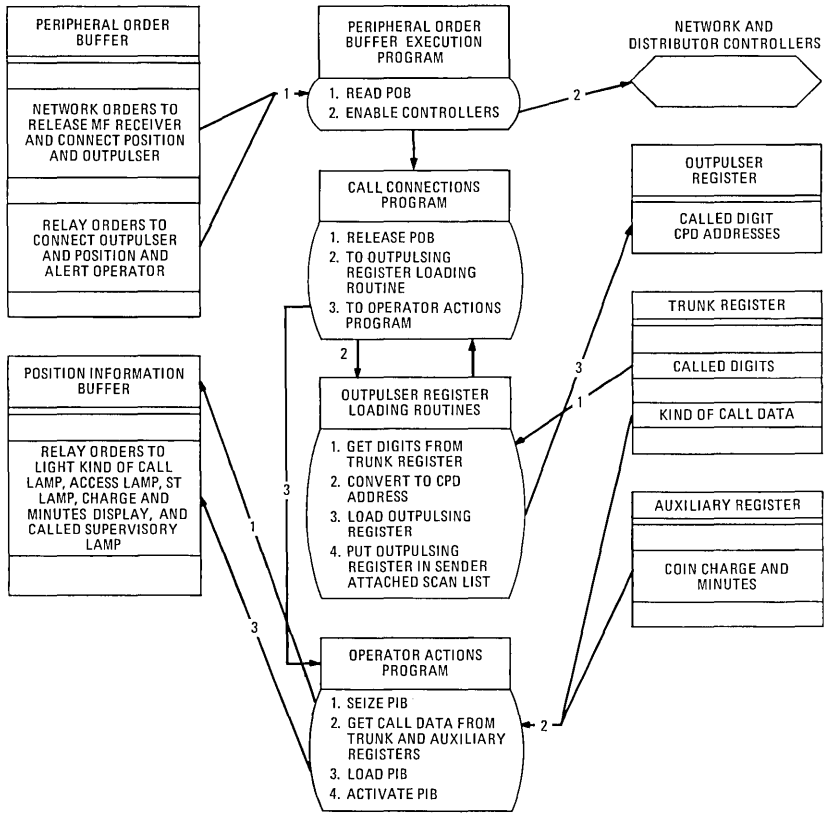
Fig. 10—Actions for connecting position and outpulser.

for loading of digits to be outpulsed, activates sender attached scanning for the outpulser, and transfers control to the operator actions program.

4.6 *Operator Actions*

4.6.1 *Initial Display*

As shown in Fig. 10, the initial function of the operator actions program is to seize a position information buffer (PIB) and provide the operator with a display on her position which indicates the type of call that she is to handle. In this case the "0+ coin" kind of call lamp will be lighted. The access lamp for the loop which is to be used
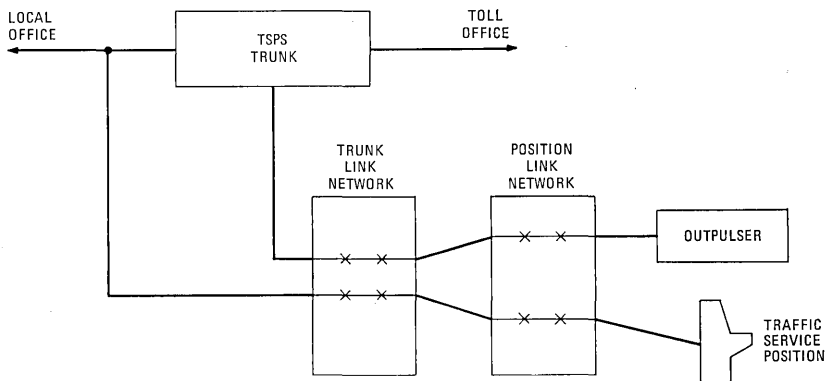
Fig. 11—Outpulser and position connection.

for the call and the ST lamp to indicate that an outpulser is attached are also lighted. Since this is a coin call, the initial display routines go to the auxiliary register where the call connections program stored the initial charge information it received from the coin rating programs and uses this data to generate a charge and minutes display on the position's numerical display panel. After completion of loading the PIB, the PIB is activated.

Upon successful execution of the PIB (as shown in Fig. 12), control is returned to the operator actions program which then idles the PIB. The next action taken by the operator actions program is dependent upon the sequence of keys operated by the operator. Other inputs to the operator actions program will be receipt of sender attached notification, followed by outpulsing complete notification, and called party answer. While these three stimuli follow each other in sequence, they
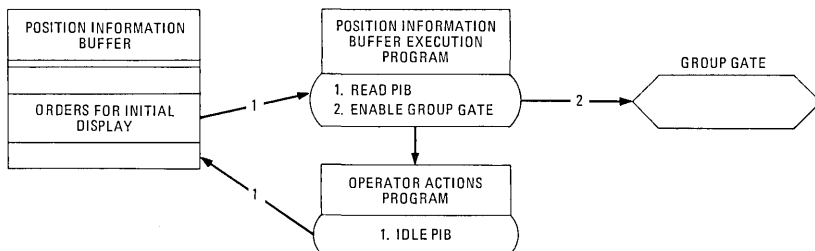
Fig. 12—Final actions for connecting position.

could be interspersed with certain key actions initiated by the operator. For sake of simplicity in this example a particular sequence of events will be assumed.

4.6.2 Receipt of Sender Attached

As shown in Fig. 13, the sender attached scan program upon detection of sender attached at the toll office deactivates sender attached scanning and loads a report in the miscellaneous scan report hopper. Sometime later, when this report is unloaded from the hopper, control is transferred to the operator actions program. At this point (Fig. 13) the operator actions program activates outpulsing of the called number.

4.6.3 Outpulsing of the Called Number

The outpulsing program is entered periodically and sends one digit at a time to the toll office until it determines that all digits have been sent. Upon completion of outpulsing the called number (shown in Fig. 14), the outpulsing program deactivates outpulsing for this outpulser and loads an outpulsing complete report in the miscellaneous scan report hopper. Sometime later when this report is unloaded from the hopper, control is returned to the operator actions program. At this time a POB is seized and control is transferred to the network control program to load orders to release the connection to the outpulser circuit. The operator actions program loads orders to close the
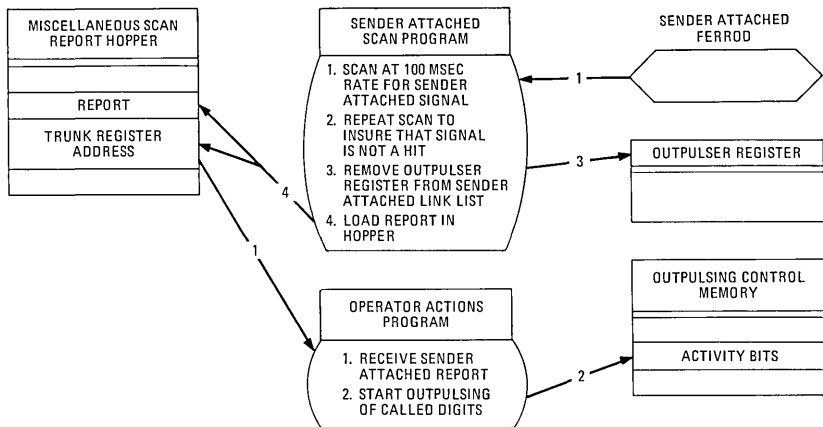
Fig. 13—Receipt of sender attached signal.
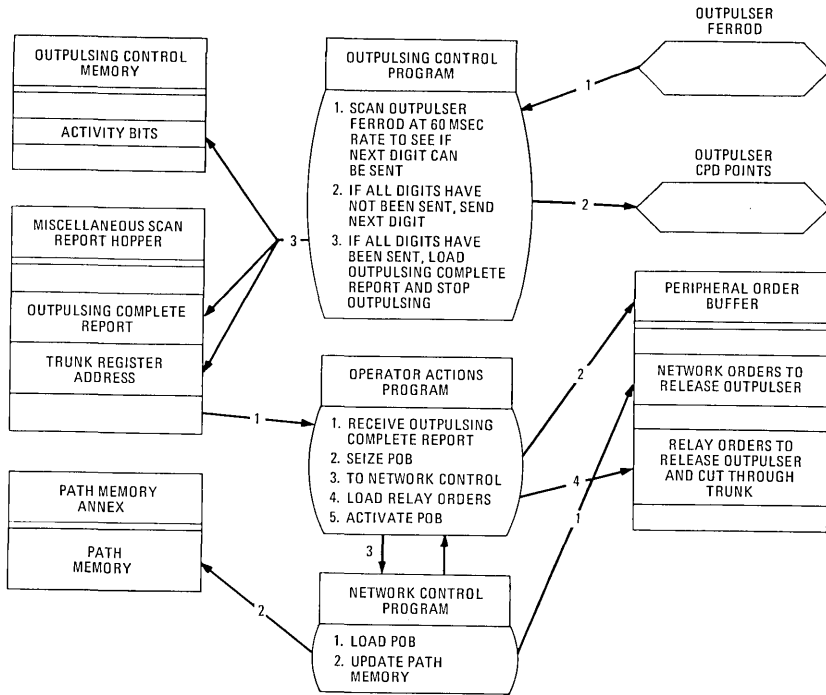
Fig. 14—Actions to release outpulser and cut through trunk.

cut-through relay of the trunk connecting the calling and called sides of the trunk. After loading these orders is completed, control is returned to the operator actions program which then activates the POB.

As shown in Fig. 15 the POB execution program causes the orders in the POB to be executed resulting in the configuration shown in Fig. 16. Upon successful completion of execution of the POB, control is returned to the operator actions program. At this time the operator actions program seizes a PIB and loads an order to extinguish the start lamp at the operator's position. This signifies to the operator that outpulsing is complete. Upon successful completion of execution of this PIB, control is returned to the operator actions program which then updates memory in the position register and trunk register to reflect that called party answer may now be expected. The temporary memory associated with the scan point of the outgoing side of the trunk is set to cause the supervisory trunk scan program to recognize when an off-hook occurs and transmit this information to the operator actions program.
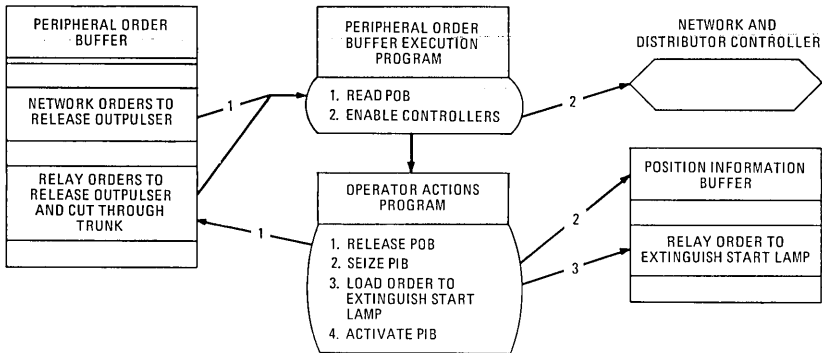
Fig. 15—Final actions to release outpulser and cut through trunk.

4.6.4 Receipt of Class of Charge

In this example it is assumed that the next stimulus to the operator actions program is receipt of a person paid class of charge key.

Upon receipt of the person paid key, the operator actions program records in the position register that a class of charge has been received and that this particular class of charge requires called party answer before position release can be accepted. It also records in the trunk register a specific class of charge index for billing. An idle PIB is seized and an order loaded to light the person paid class of charge lamp to indicate to the operator receipt and acceptance of her key signal. For this example it is assumed that the operator will wait for called party answer and identification before executing any more key operations. She will also at this time request from the customer deposit of charges as displayed to her at the time her position was seized.

4.6.5 Receipt of Called Party Answer

Upon detection of an off-hook on the outgoing side of the trunk (shown in Fig. 17), the trunk supervisory scan program loads a report in the trunk seizure and answer hopper and sets the temporary memory associated with the outgoing side ferrod to ignore any further changes of state until directed otherwise by some base level program.

Sometime later this report is unloaded from the hopper and control transferred to the operator actions program. At this time (as shown in Fig. 17) the operator actions program seizes a flash scan timing register (FSTR) and initializes it to scan the outgoing ferrod of the trunk at a 100 millisecond rate to determine if this is a true answer
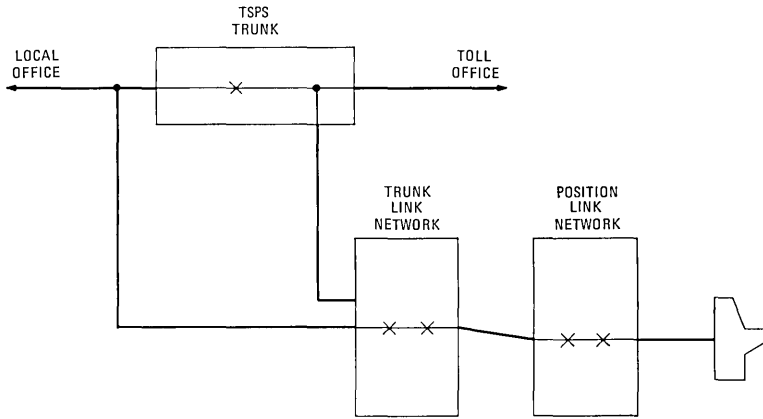
Fig. 16—Trunk cut through with position attached.

or the beginning of flashing signifying a busy or reorder condition. Upon establishing that the off-hook was in fact an answer, the flash scan timing program (as shown in Fig. 18) loads an answer report in the miscellaneous scan report hopper.

Sometime later this report is unloaded from the hopper and control returned to the operator actions program. Upon receipt of this report the operator actions program records in the position register that a called party answer has been received and executes a PIB to ex-
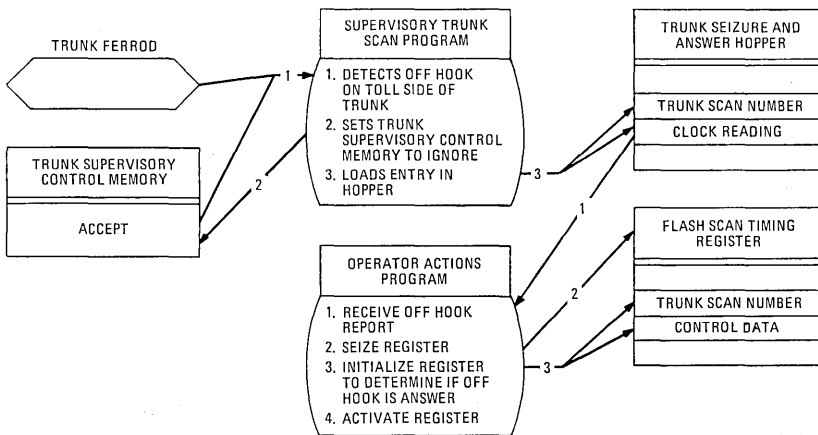
Fig. 17—Receipt of called party off hook.
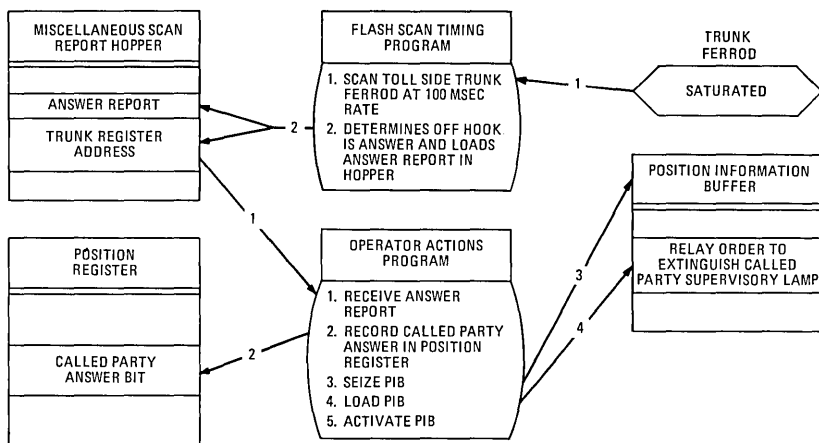
Fig. 18—Receipt of answer report.

tinguish the called supervisory lamp at the position to indicate to the operator that the called party has answered.

4.6.6 Final Operator Actions

The operator now establishes when the desired party has been reached and conversation has started. She then depresses the start timing and position release keys.

Upon receipt of the start timing key, the operator actions program interrogates memory to establish that all necessary billing information has been provided by the operator. As this example has evolved, the result will be positive and a PIB will be executed to light the start timing lamp indicating to the operator acceptance of the key.

Upon receipt of the position release key, the operator actions program performs a sequence of tasks to release the connection of the trunk to the position. The first action (as shown in Fig. 19) is to perform a scan of the hardware clock and record this in the trunk register as the call start time. It then seizes a POB and transfers control to the network control program to load orders in the POB to release the connection to the position. After completion of the loading of these orders, control is returned to the operator actions program which loads relay orders to idle the operator cut-through circuit and then activates the POB. As shown in Fig. 20, the POB execution program causes the orders in the POB to be executed. Upon successful

execution of the POB, control is returned to the operator actions program.

At this time (as shown in Fig. 20) the operator actions program sets the trunk register memory to reflect that the trunk is now in a talking state, activates supervision on both sides of the trunk to watch for disconnect, and enters a subroutine to place the trunk register on a coin timing list. The coin timing program will later cause a return to the disconnect program approximately 18 seconds prior to the end of the initial period.

After putting the register on the timing list, the operator actions program loads a PIB and causes it to be executed which will extinguish all the lamps on the operator's position that were related to this call. Upon completion of this PIB the operator program causes the position register to be returned to the idle link list.

V. DETECTION OF DISCONNECT

In this example it is assumed that the call terminates during initial period, prior to the initial period coin collect sequence, and is initiated by a calling party disconnect.

The supervisory trunk scan program scanning the trunk ferrods at a 100 millisecond rate detects the change of state in the incoming side
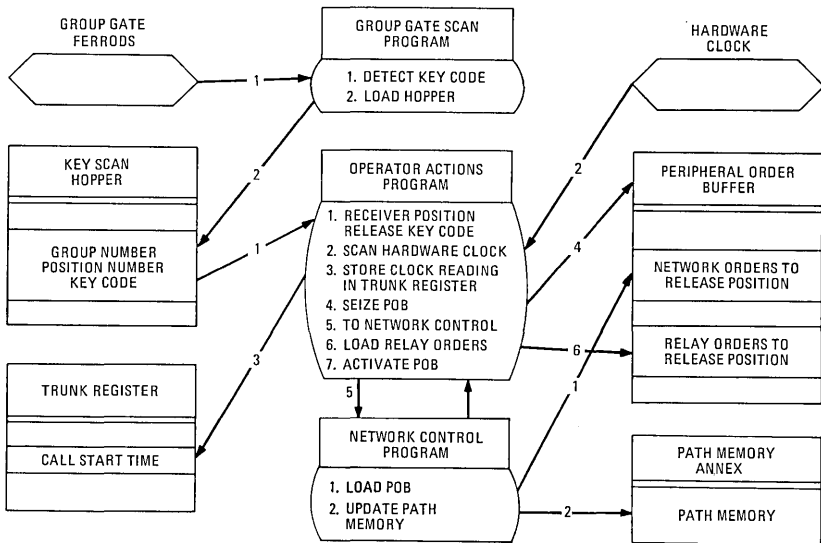
Fig. 19—Initial actions for releasing position.
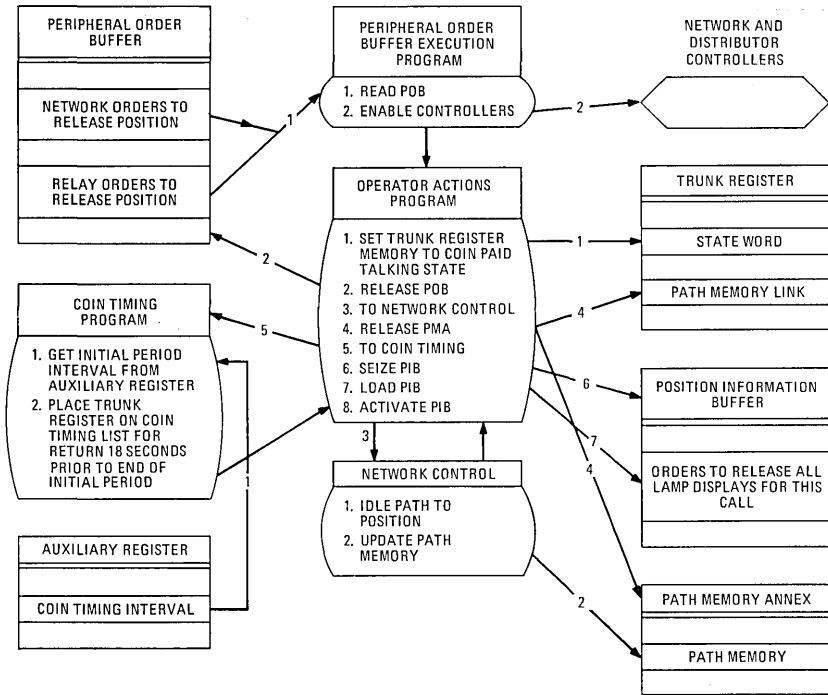
Fig. 20—Intermediate actions for releasing position.

ferrod from off-hook to the on-hook state as shown in Fig. 21. The supervisory trunk scan program then seizes and initializes a timed scan junior register (TSJR) by loading the trunk scan number and control data in the register to start a directed scan of the incoming ferrod.

Two hundred milliseconds later, the hit scan program reads the incoming ferrod as indicated by the trunk scan number, and if the ferrod still indicates on-hook, it recognizes a potential disconnect. The hit scan program then loads the trunk scan number along with the TSJR address into the hit scan result hopper with an indication that the on-hook was not a hit.

Later, the hopper entry is unloaded (as shown in Fig. 22) and control is given to the disconnect program. The disconnect program stores the disconnect time obtained from the TSJR in the trunk register and removes the trunk register from the coin timing list. It then begins its initial disconnect actions. An idle POB is seized and initialized, and after loading orders to release the forward connection

on the trunk, control is transferred to the network control programs to establish a connection to a coin control circuit. When these orders have been loaded, control is given to the coin trunk program, and this program completes the loading of relay orders in the POB and activates it. Upon successful execution of the POB, control is returned to the coin trunk program as shown in Fig. 23. The coin trunk program at that time seizes a multibit scan register to perform a periodic directed scan of a ferrod in the coin control circuit which indicates when the coin signal sequence of the circuit is complete. The multibit scan program performs a directed scan of the coin control circuit ferrod at a 100 millisecond rate. When the appropriate change of state is detected, the multibit scan program loads a report in the miscellaneous scan result hopper.

Sometime later (as shown in Fig. 24) the report is unloaded from the hopper and control is returned to the coin trunk program. The coin trunk program seizes an idle POB, loads orders for the idling of the coin control circuit relays, and activates the POB.
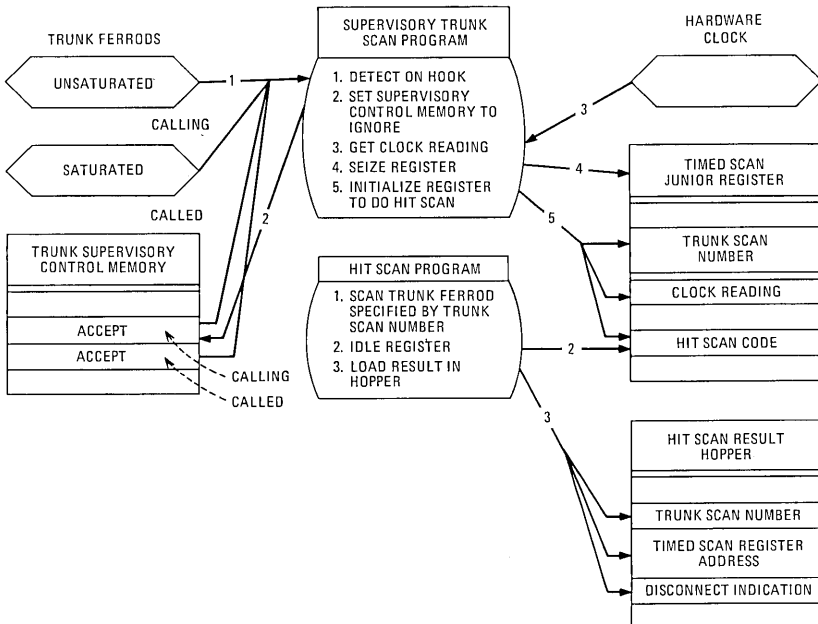
Fig. 21—Detection of on hook and hit scan.
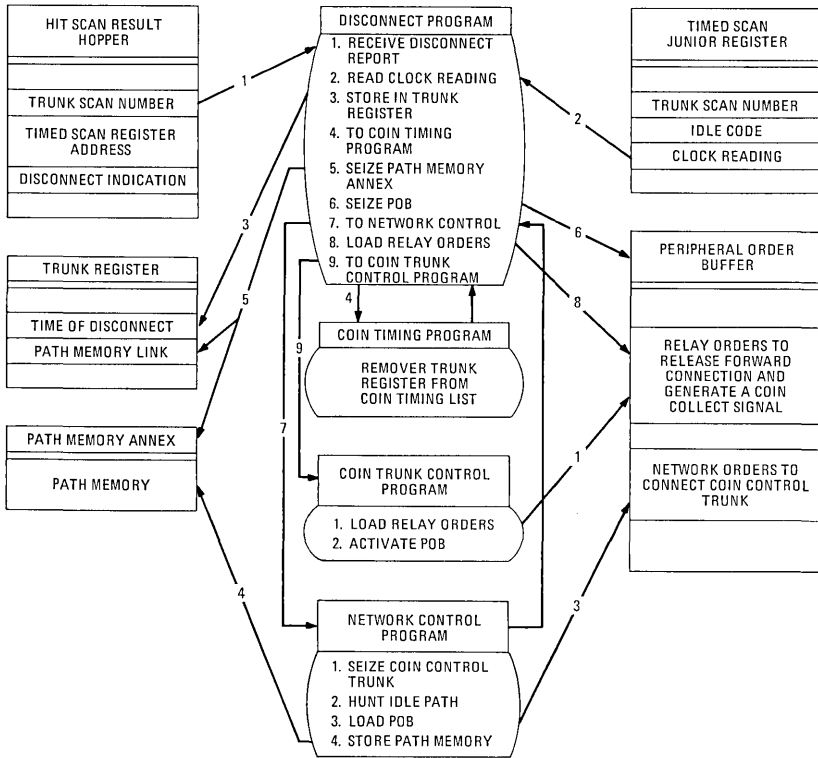
Fig. 22—Initial disconnect actions.

Upon successful completion of this POB, control is returned to the coin trunk program which then returns to the disconnect program indicating successful completion of coin actions. Using the same POB, the disconnect program transfers to the network control program which loads orders to release the connection to the coin control circuit. The disconnect program then activates the POB.

As shown in Fig. 25, after the orders in the POB are carried out, the disconnect program releases the POB and transfers to the billing accumulation program. The billing accumulation program transfers all pertinent information from the trunk register to a buffer area from which the data will be later transferred to magnetic tape. After the billing information is transferred, the disconnect program seizes an idle POB and loads orders to idle the relays of the trunk which re-

leases the off-hook condition at the outgoing trunk in the local office allowing it to return to normal and to serve new calls.

Upon successful execution of this POB, control is returned to the disconnect program as shown in Fig. 26. Since this disconnect action could have been initiated with the calling party off-hook and the local office trunk not yet returned to normal (i.e., on-hook), a scan of the incoming side ferrod is performed in order to prevent a false seizure. If the incoming ferrod is off-hook at this time, the trunk register is placed on a high and wet list where the incoming side ferrod is supervised at a 100 millisecond rate looking for an on-hook condition. When the on-hook condition is detected, idling of the trunk can be completed.

In this example, it is assumed that the calling party disconnects first. Hence, the trunk can be idled immediately. This is accomplished by initialization of the trunk register to indicate that it is in the idle state. The supervisory scan control bits are initialized to the state
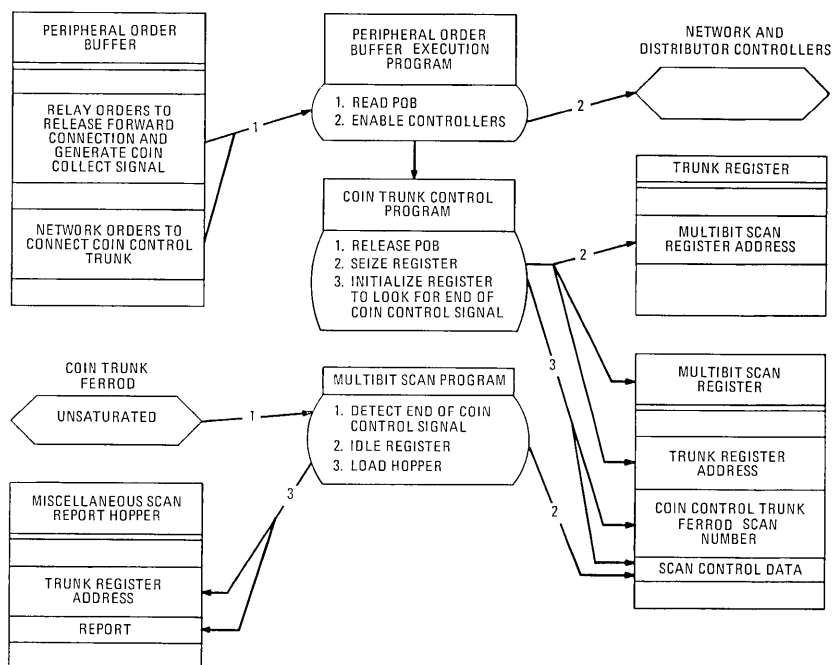
Fig. 23—Monitoring of coin control signal.
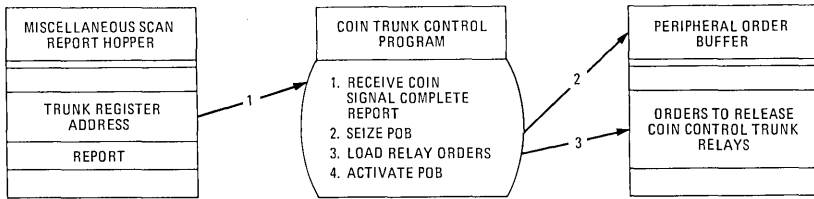
Fig. 24—Receipt of coin signal complete report.

which will permit the supervisory trunk scan program to detect a new seizure. At this time the trunk is ready to serve a new call.

VI. AUDIT PROGRAMS

6.1 Purpose of Audits

In a program controlled system, such as TSPS, continuous system operation depends not only on the maintenance of a working hardware configuration but also on the maintenance of a working software configuration. Outages in software facilities, a condition commonly known as "data mutilation," can render the system inoperative. To guard against such mutilation, a group of programs, called audits,
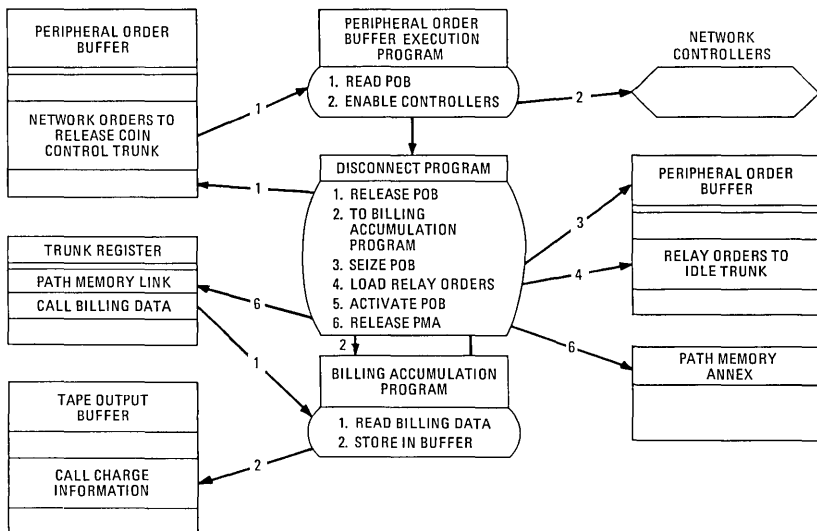
Fig. 25—Charging actions and intermediate disconnect actions.
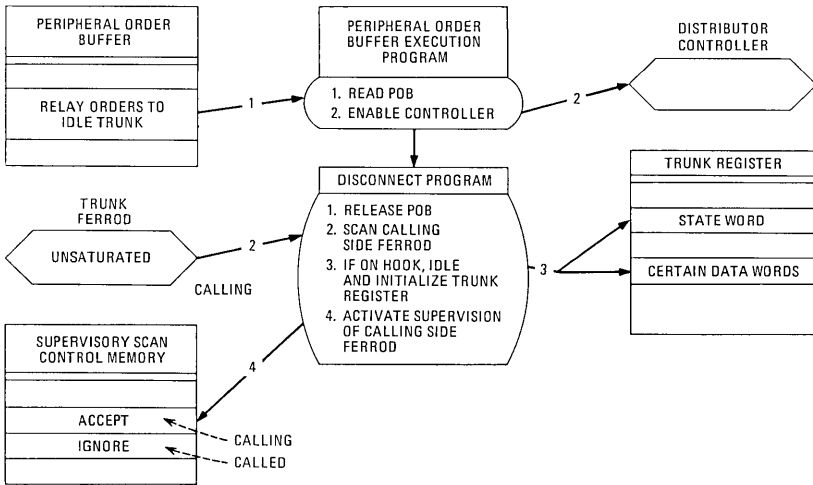
Fig. 26—Final disconnect actions.

monitors the software for error states, takes corrective action when appropriate, and initiates teletypewriter messages that record the nature and the number of the errors encountered.

Most of the audit programs run on a quasi-continuous basis as a maintenance time filler, operating whenever no other maintenance work is pending. A complete cycle of all time filler audits nominally takes one minute, with some of the more critical programs cycling on a shorter interval. Other audit programs which monitor memory areas whose error states are not critical to system operation run less frequently on an hourly or daily schedule.

6.1.1 Use in System Initialization

Audit programs are also used for initializing memory associated with call processing and maintenance programs. In many instances, this initial state is all zeros. In other cases, such as for idle link lists of common software facilities, it is non-zero. Initialization procedures first zero all unprotected memory. The non-zero initial states are then established either by special initialization programs or by audits which recognize all zeros as an error state and consequently correct the condition by placing the memory in its initial state. Memory initialization is performed not only for the initial startup of the system but also is performed in an operating system when major outages

beyond the detection and correction capability of the audits requires special reinitialization. The latter action, known as Call Processing Recovery, is described later.

6.2 Examples of Audit Programs

The TSPS No. 1 system has about 45 distinct audit programs. The following is an example of one of these.

6.2.1 Link-List Checking

Many call processing programs require software facilities to be structured in linked lists. For example, the software registers which are dedicated to idle multi-frequency receivers are linked as shown in Fig. 27.

The head cell, located in a permanent location, contains the memory address of the first idle register. A link word in the first register, in turn, contains the address of the second, the second of the third, and so forth. The last idle register on the list is marked by an all-zero link
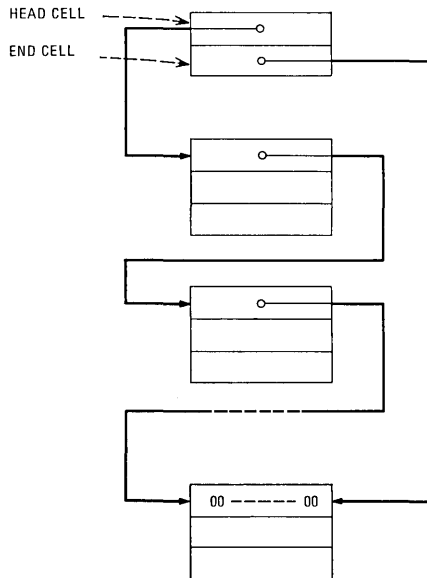
Fig. 27—Link-List Structure. Memory addresses are stored in a word of each member on the list to link that member to the next on the list. The head and end cells contain, respectively, the addresses of the first and last members on the list.

word. The end cell points to the last register. When call processing programs require an idle MF receiver, they obtain one by taking the register at the top of the list, after which the head cell is adjusted to point to the next on the list. When the programs no longer need the receiver, they restore it by placing its register at the bottom of the list, as determined by the end cell.

Audit programs maintain the list by insuring that:

(i) only idle registers are on the list,

(ii) registers not on the list are truly in use, and

(iii) the list structure is proper. All linkage addresses are of memory locations dedicated to multifrequency receivers, and the end cell points to the last on the list.

Any detected error state causes audits to take corrective action. The exact corrective procedure depends on the specific error detected. In general, the action is to restore all idle registers to the idle link list where they will then be available for future call needs. In the process, any register with an inconsistent state will be set to the idle state, placed on the idle link list, and the relays in its associated multifrequency receiver reset to idle. The restoral procedures do not alter any register in use on a call, provided that the software status of that register is consistent.

Since idle registers are marked by an all-zero status word, the audit which rebuilds the idle link list can also be used to initialize the idle link list. Initialization procedures will have first zeroed all scratch memory, thereby making all registers appear to be idle. Consequently, these registers will all be placed on the idle list.

6.2.2 *Call Processing Recovery*

On occasion, memory mutilation can become so severe that it renders the call processing system inoperative. In these instances the normal recovery action of routine audits is inadequate. Depending on the situation, they may not be able to gain control to correct the trouble. For example, when a program goes into an infinite loop, they may not be able to restore the data as fast as it is being mutilated; or they may not be powerful enough to detect the trouble. Whatever the cause, when such a situation occurs, all call processing is suspended so that the system can be devoted entirely to recovery actions. Depending on the severity of the mutilation, recovery may be performed quickly, with little disruption to service, or it may require lengthy ac-

tions with considerably more disruption, both to new calls attempting to obtain service as well as to calls being processed.

6.2.2.1 *Software Sanity Checks.* Improper software operation is detected by a set of heuristic tests, known as the Software Sanity Checks. These checks, while not capable of recognizing all possible trouble situations, are usually able to recognize improper operation. Checks are made for the following situations:

(*i*) Base-level cycling does not exceed prescribed time limits. Under normal full-load conditions entries to the base level E-priority class work are made at an average rate of once every 300 ms. If the total duration of three successive entries to this class exceeds 12 seconds, it is assumed to be due to a software problem, such as an infinite program loop.

(*ii*) Low priority J-level interrupt tasks are not called within a prescribed time limit. This check recognizes if high priority J-level activity takes an excessive amount of time, or if base-level activity is performed with the hardware in a J-level interrupt. Under these circumstances, the low priority J-level work will not be entered.

(*iii*) Relative frequency of base-level jobs is improper. A-priority jobs must be performed twice as often as B-priority jobs, B-priority jobs twice as often as C-priority jobs and so forth. If not, the system is skipping over some of its work functions.

(*iv*) Requests for interject work not answered. Base level work which must be performed within a strict time tolerance is executed upon a demand request for interject work. This check periodically makes such requests to insure that the interject mechanism is operating properly.

(*v*) Excessive maintenance interrupts occuring. This condition can be symptomatic of improper data in memory, for example, improper enable codes. If more than 15 interrupts occur within twenty seconds, this check triggers recovery operations.

(*vi*) Excessive out-of-range store addresses occuring. If memory write operations into protected (nonwritable) memory or if read or write operations of non-existent memory addresses are made, it is the direct result of data mutilation or improper program sequencing. Three such occurrences within twenty seconds triggers recovery operations.

6.2.2.2 *Description of Recovery Phases.* Recovery operations are performed in segments, known as phases. The initial request for Call

Processing Recovery brings in the lowest phase, Minor Audits. The actions of this phase are short and, as the name implies, minor. When the actions are completed, control is returned to the normal call processing programs. If the phase actions were successful, call programs will operate properly and no further Call Processing Recovery actions will be needed. But if unsuccessful, the software sanity triggers will again detect system malfunction. If the interval of call operations is less than twenty seconds, recovery proceeds to the next higher phase, Major Audits, where more extensive recovery operations are performed. Upon completion of the phase actions, normal call processing work is resumed. This process of calling in succeeding higher level phases continues until recovery is successful. The highest level phase, consequently, must be designed to be able to achieve recovery under all conceivable conditions of data mutilation.

The severity of mutilation, and hence the degree of recovery actions needed, cannot be predicted. For minor problems, recovery is rapid and disruption to call service minimal. But for major problems, recovery is more lengthy and the disruption to call service is more severe, due to the long duration of the phases and to the extent of the actions performed on calls by the phase operations. For those major problems which require the full sequence of recovery phases, overall recovery is further delayed as the system sequences through the lower level phase operations. If the severity of the problem could be predicted in advance, recovery operations for major problems could be shortened by skipping the lower level phases and jumping immediately to the highest. Unfortunately, such a prediction cannot be made.

An optimal recovery strategy, hence, must place sufficient recovery power in each phase to insure good probability of success over all possible problems. The inclusion of each phase in the recovery structure is thus based on its probability of success when compared with its effect on call service.

It is also important that the most severe system initialization phase leave as many calls undisturbed as possible without jeopardizing the effectiveness of the phase. One of the serendipitous advantages of the simple design of the TSPS universal trunk circuit permits calls that are in the customer-to-customer talking state to continue undisturbed through all phase recovery actions. The talking state is maintained by the continued operation of a single relay, and the state of the trunk can be reconstructed by determining the state of the incoming and outgoing supervisory ferrods.

The following phases have been included in the TSPS No. 1 Call Processing Recovery structure:

(i) Minor Audits

The lowest level phase is called Minor Audits. Its duration is of the order of a few seconds. This phase performs audits on a few short but critical software facilities. Among its actions, it verifies the enable tables, and checks all Executive Control status indicators. No call-associated data is checked in this phase. Consequently, no telephone calls are affected by its actions. However, calls in a real-time sensitive state, such as calls in a digit transmission mode, are affected by the suspension of call processing actions. These failures are not detected during the phase, but are discovered later when call processing resumes and, for the above example, recognizes that an insufficient number of digits were received. The condition is then cleared by normal call processing failure actions.

(ii) Major Audits

The second level of recovery phase, known as Major Audits, is nominally about thirty seconds in duration, but varies with office size. This phase performs all logical audit checks available, including the example described in Section 6.2. All calls with consistent memory states are left untouched.

For those calls which have mutilated data, the trunk ferrods are examined for off-hook states on both the calling and called sides. If both sides are off-hook, indicating a "talking connection" between customers, the connection is maintained and the software set to a state wherein the customers' eventual disconnect will be recognized. Under these circumstances, to minimize system dependence on mutilated data, all other call data associated with the call, including billing data, is destroyed. Consequently, the call continues free of charge.

(iii) System Initialization A

Since the Major Audit phase utilizes all available audit checks, its failure to achieve recovery signifies that the problems are uncorrectable by audits. Consequently, the next phase, System Initialization A (SIA), must initialize existing data and hence affect certain telephone calls. In this phase, all unprotected (call data) areas of memory except for a few select areas such as the system software clock are zeroed. Memory is then rebuilt to an initial state and all hardware is initial-

ized. As part of the memory rebuilding and hardware initialization process, all calls which had been in a customer-to-customer talking state are preserved, and allowed to continue free of charge. As described under the Major Audit phase, this state is recognized by noting that ferrods on both sides of the TSPS No. 1 trunk are in the off-hook state.

Although customer-to-customer talking connections are retained, other calls are disrupted. Talking connections to operators as well as all connections to digit receivers, announcement circuits, audible ringing tone, and the like are disconnected. The actions of the phase require about one minute to complete.

(iv) System Initialization B

Since the SIA phase places very little reliance on past data, the main cause for it to fail would be hardware troubles. Consequently, the main differences between System Initialization B (SIB) and an SIA lie in the hardware actions. In an SIB hardware initialization orders are sent over all possible, and hence redundant, paths. Since this phase is the final phase possible in the sequence, all areas of memory, including those retained on the previous phase, are zeroed. However, similar to the previous phase, calls in a customer-to-customer talking state are retained.

Because of the added hardware actions, the phase duration is longer, requiring about $1\frac{3}{4}$ minutes to complete. In accord with the assumption that the phase is needed only when hardware problems are contributing to the data mutilation, maintenance interrupts are left inhibited upon completion of the phase. This action allows the system to continue processing calls even though a peripheral hardware unit might be creating excessive interrupts.

In the unlikely event that none of these phases achieves recovery, problems other than unprotected memory mutilation would have to be corrected. Hardware outages of duplicate units, such as both processors, would naturally render the system inoperative. Should there be a mutilation of memory in the protected storage areas, which is highly unlikely, the bootstrap techniques described later under the Program Tape Unit Control programs would have to be used.

VII. PROGRAM TAPE UNIT CONTROL

One of the major advantages of the SPC System is the ease with which the contents of memory can be changed. This feature extends over protected memory as well as unprotected storage and permits

the alteration of programs and protected data. While the teletype-
writer is available to the craftsman for changes in protected data on
a small scale, that instrument is unsatisfactory for many reasons
when large amounts of data or program changes are necessary. The
Program Tape Unit (PTU) is provided with the SPC system to facili-
tate large scale data and program changes as well as a tool for re-
trieving and storing on magnetic tape the contents of memory. This
section describes the control program for the PTU and its use.

7.1 On-Line Operation

The Program Tape Unit normally operates in an on-line mode on
a time-shared basis with other programs. This mode is used for both
memory to tape, as well as tape to memory operations. Data transfers
to and from the tape are performed every 5 ms at J-level interrupt.
The transfers are made between the processor and the tape unit, with
the program logic performing parity checks and software buffer load-
ing or unloading operations. Nominally, 5 tape characters or one SPC
40-bit word are either read from or written on the tape during each
interrupt.

Whereas the input/output transfer operations occur at interrupt
level, the processing operations are performed at base level. If data
is to be transferred to tape, a record size of 128 words, or 640 charac-
ters, is formed at base level and stored in an output buffer in memory.
The tape is then started and the J-level transfer program activated.
When all characters in the record have been transferred to tape, the
tape is stopped and another record formed, as above. This process is
repeated until all data has been transferred.

Similarly, to transfer data from tape to memory, the tape is read one
record at a time by the J-level transfer program, and the characters
stored in a memory buffer. After each record is transfered, the tape is
stopped while the base level program checks the parity of the input,
assembles the characters into 40-bit words, and updates memory or
matches with memory. In the latter case, when a compare only func-
tion is being performed, all discrepancies are reported via the system
maintenance teletypewriter.

Transferring data from tape to memory is performed to introduce
program changes. In this mode the tape contents are written in only
one memory bus system. The bus to be loaded is first forced to the
standby state by controls at the Master Control Center. This manual
operation results in the SPC system configuration for normal memory
operations shown in Fig. 28. The loading process is performed on only

ACTIVE STORE BUS

SEND · RECEIVE · RECEIVE

ACTIVE
PROCESSOR ← - - - NO DATA - - - → STANDBY
PROCESSOR
MATCHING

SEND

STANDBY STORE BUS
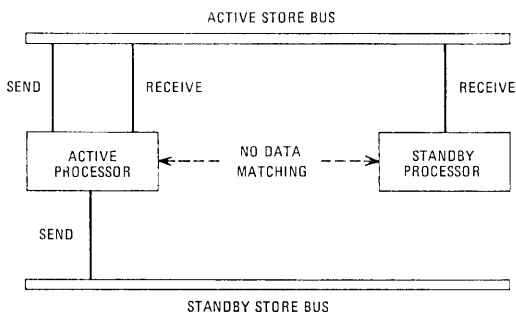
Fig. 28—SPC system configuration for memory operations.

the standby bus by use of special instructions. While the loading
process is in progress, the active bus continues to process telephone
calls. Since the active processor, by sending data to both store buses,
maintains the unprotected memory on the standby bus, it is possible,
if the changes to the standby bus are minimal, to reload memory and
switch buses to the new issue without loss of call processing continuity.
However, for some types of changes a system initialization (SIA)
phase will have to be induced after the switch to the new bus since
the scratch data in unprotected memory will probably be inconsistent
with the new program changes. Normally, a new program load is made
only during slack traffic hours (e.g., after midnight) to minimize the
number of customers affected by the SIA. It should be noted that if
the new program issue includes an expansion of protected memory, the
active processor will try to write in protected memory locations on the
standby bus after the protection boundary is changed. An all-seems-
well write failure will result on this bus and the write will not be
performed on the standby bus. But since the processors are not
receiving from the standby bus, the ASW failure is ignored, and sys-
tem operation is not affected.

   Upon completion of loading, the standby and active buses are
switched by controls from the Master Control Center, thereby activat-
ing the new program. Should the new program operate improperly, the
old program can be restored by switching the unaltered bus back to
the active state. The two bus systems are kept dissimilar until testing
of the new program is completed to the extent made possible by the
simplex bus configuration. At that time, the buses are brought into
agreement by copying the contents of the new bus to the old. This
is accomplished via a special store program which utilizes the normal

bus-to-bus update feature of the store diagnostic program. Then, further testing including the full store diagnostic sequence is initiated and duplex operation is restored.

Because protected memory contents are occasionally altered, primarily by the Recent Change programs, memory is copied onto tape periodically to generate a new back-up of the protected memory. This operation is typically performed monthly. The new tape thereby made is available for use should a need for reloading the memory arise and is periodically verified against the contents of memory.

7.2 *Off-Line Operations*

Off-line program tape unit operations are available for tape to memory transfer only. In this mode all other program activity is suspended. A special program, called the "bootstrap" program, operates in A-level interrupt to perform the sole function of loading memory from tape. Bootstrap operation is used primarily to load memory for the first time during installation of the system. In the event of a catastrophy it can be used to rewrite a mutilated program.

After a system is in operation, bootstrap is used as a last-resort recovery operation in the event that protected memory is mutilated to a degree that normal call programs cannot operate. In this event the memory back-up stored on tape is loaded to regenerate the memory. This is accomplished through use of a special bootstrap transfer card extender which is plugged into the active processor. Once inserted, a manually induced A-level interrupt at the Master Control Center causes the processor to enter the bootstrap program. In this use of the PTU, the new program or data is written into the stores on both buses simultaneously. The process is as efficient as possible to minimize the time to restore the system to normal operation. In fact, the limiting factor in the operation is the data transfer rate of the PTU itself.

When it is used to load the system for the first time during installation, the bootstrap program is manually written into memory using the test cart. The full system loading operation described above is then used to complete the operation. In order to protect against mutilation of the bootstrap program itself, which would require manually reloading the program in a working office, special precautions are taken. Four copies of the program are stored in memory with two copies on each bus. Any one of these copies can effect the loading operation in case of partial mutilation of the bootstrap program. Selection of the

copy of the bootstrap program to be used is manual and is made by means of switches on the bootstrap transfer card.

VIII. PURPOSE OF RECENT CHANGE PROGRAMS

Within the protected area of memory are stored items known as office data parameters. These items contain data specifications which are unique to a specific office. The contents of these locations, while varying from one office to another, generally do not change within an office over periods of months or even years. Some locations, however, do occasionally require alterations. While the occurrence of such changes is infrequent, the advance notice is often short and the necessity great. For example, if traffic conditions change, trunks have to be added, deleted, or moved. As new telephone central offices are established and tariff changes enacted, coin rating tables must be modified.

A group of programs, known as Recent Change programs, are designed to permit the Telephone Companies to alter such office data parameters. The Recent Change programs accept functional requests for data changes, as transmitted over the Recent Change teletypewriter. The programs check the request for accuracy and alter office data tables as required.

Recent Change programs are provided for only those functions which might require variations within the normal office engineering period of two or three years. Major data updates, or recompilations, which would be required at the end of the interval will be done off-line and will not be discussed in this article.

8.1 *Control Structure*

Because data alterations require Recent Change programs to unlock and alter protected memory, special precautions are taken to insure that the entire change procedure is performed correctly. Alterations of protected memory locations are particularly critical. For example, an incorrect write at a program location could render the entire program system inoperative. In the event of such mutilation of protected memory, recovery could be achieved only by a "bootstrap" reload (see the section on Program Tape Unit Control).

8.1.1 *Teletypewriter*

Recent Change requests are transmitted to the system over the Recent Change TTY channel. This teletypewriter unit is equipped

with a paper-tape reader to speed up the input operation and to increase the accuracy of the data through off-line preparation and checking. A control character is placed on the tape at the end of each line to stop the tape reader to allow time for the Recent Change programs to process the input.

As each line is received, the normal TTY programs verify the format of the line, translate the input fields to a form expected by the Recent Change programs, place the result in the Recent Change Buffer, and activate the Recent Change program appropriate for that message. When the Recent Change programs have completed their processing, they instruct the TTY programs to transmit back the last line (known as a "print-back"). Thus, each input line is typed twice, the first time when transmitted to the system, and the second time on the print-back. The telephone craftsman can then verify the correct transmission of the message by noting that the two lines are identical. If the Recent Change programs expect another input line to follow, they cause a control signal to be transmitted back to the teletypewriter unit to turn on the paper-tape reader. The next line is then transmitted and processed in the same manner as described above. This process continues, line by line, until all data has been received.

8.1.2 *Message Sequencing*

A functional change requires multiple lines of input data. For the change to be processed correctly, the lines must be transmitted in a predetermined order. The Recent Change Control program monitors this sequence. Each change message must start with a control line, known as the "BOC," for Beginning Of Change. This line establishes the type of change expected. Only a subset of all Recent Change messages can follow the BOC. Should any other message be transmitted, it would be rejected by the Recent Change Control program. The programs which process each line establish the set of messages which may legally follow.

When all lines necessary to complete a change function have been transmitted, the Recent Change programs are so informed by an EOC (End of Change) input message. After transmission of the EOC, the paper-tape reader is stopped to permit further verification. At this time, depending on the change, there may be special verify messages available to test the change. If the verification is satisfactory the entire message is then activated by typing in an "ACT" message. This message signals the system to update the permanent translation tables.

All updating is performed in duplex on both memory systems. If, however, the message is not to be activated, it can be cancelled by typing in a "CAN" message.

As the change programs process the input data, they compute the addresses of locations to be changed and the data to be stored in those locations. The addresses of the locations and the new data are stored in the Recent Change Work List. When the ACT message is received, the Recent Change Control Program uses the Work List to update permanent memory. As each word is updated, the former contents of that location, known as old data, are saved with the associated address for possible restoration.

If any maintenance problem should cause the undating process to terminate before completion, memory is restored to its prior state by restoring the old data. If any major system problems occur within fifteen minutes after completion of updating, such as a Call Processing Recovery phase or a maintenance interrupt, it is assumed that the Recent Change activity was responsible for the trouble and the change is automatically erased from the system and the old data is restored.

8.2 *Verification of Change Messages*

Each Recent Change message performs extensive checks on its associated input message to verify the reasonableness of the change request. Range checks are made against office parameters to insure that they are within bounds. Each input message must include information, known as old data, that describes the present status of the area to the changed. Verification of the old data insures correctness of the office records on which the change is based and insures that the processing programs are altering the correct locations.

8.3 *Types of Changes*

Two general types of Recent Changes can be made. New requirements at the local office can require changes to trunk circuit data, and new tariff requirements or establishment of new telephone central offices can require changes to coin rating tables. An example of the latter follows:

8.3.1 *Coin Rating Data Change*

The TSPS No. 1 system provides automatic computation of charges for coin telephone calls. The calculations are made by computing air-

line distances between the originating and terminating offices and obtaining the appropriate charges for the distance, type of call (person-to-person or station-to-station) and time of day.

Extensive office data tables are stored in protected memory to provide the geographic location of all originating and terminating offices. The locations are stored as vertical and horizontal distances from a fixed reference point. Whenever new central telephone offices are placed in service, or when changes about an office in service are made, data tables have to be altered. The following is the sequence of messages required to provide coin rating for a new central office:

    BØC-99-ØRD:023,TYP:ADCØ,08:01:69.
    RCØ-20-NXX:949,NPA:201,ØRTP:VC,NRTP:VH.
    RCØ-21-DATA:NEW,VERT:05086,HØRZ:01380.
    EØC-99-ØRD:023.
    ACT-99-ØRD:023.
    VER-48-NPA:201,NXX:949. (Not required, for vertification only.)

The BOC-99 message signifies the beginning of the change. The ORD field specifies an order number, as given by the Telephone Company. The type field, ADCO, indicates that an addition of a central office is to be made. The remaining field gives the date. The RCO-20 message identifies the new office code as 949 in the 201 numbering plan area. The program checks that the old rating type (ORTP) for the office, as stored in memory, is vacant (VC), and it changes this data to the new rating type (NRTP) of V and H (VH) for Vertical and Horizontal coordinates. The next line gives the coordinate values, as measured from a standard reference point, and identifies the data as NEW. Since the office code was previously vacant, there is no OLD coordinate data to be checked.

The EOC-99 message informs the Recent Change programs that no further data is to follow. The change is activated by the ACT message, after which the change can be vertified by typing the VER-48 message. The verification program checks the office data to determine the status of the office code, and prints back a response listing the rating type and V and H vector values. The response, of course, should match the change request made in the RCO-20 and RCO-21 messages.

IX. ACKNOWLEDGMENTS

described in this article. They are too numerous to list completely here, but several deserve special mention: E. E. Hanna for his work in coin call rating and computing, Mrs. C. Millheiser, G. Giani, and S. Lederman in call processing, R. D. Kroning and H. E. Voigt in operator actions, D. J. Sidor in recent changes, and J. J. Victor in program tape unit control.

TSPS No. 1:

Physical Design

By W. R. BALDINGER and G. T. CLARK

Traffic Service Position System (TSPS) No. 1 is designed to make toll customer service and operator call handling more efficient. This equipment is designed to give improved and dependable service. Connectorization is added to permit more efficient shop testing, installation simplification and a ready means to effect quick growth in a working environment. The switchroom equipment and operator facilities are specially designed for pleasing proportion and harmonious color schemes.

I. INTRODUCTION

The Traffic Service Position System (TSPS) No. 1 (see Fig. 1, which shows a portion of a chief operator unit, Miami, Florida) utilizes modular design that enables customers to order equipment orderly and economically as needed. Frames in modular multiples of 1 foot, 1 inch in length permit a highly standardized floor plan arrangement. This in turn makes possible a higher degree of connectorization. In TSPS close to 90,000 interframe and intraframe leads are connectorized in a maximum sized office.

The TSPS development was undertaken with the intent to use as much "in production" apparatus, wired equipments and frameworks as possible. In addition to existing conventional electromechanical hardware, basic No. 1 ESS apparatus such as the semiconductor circuit packs and mountings, ferreed switches, ferrod sensors, bus transformers and terminal strips were used. Also adopted with minor modifications were power distributing, master scanners, signal distributor, central pulse distributor, tone and distributing frames, sheet metal frameworks, cable racks, fuse panels, filter units, and control panels.

The adoption of this hardware offered considerable advantages: (*i*) reduced development effort, (*ii*) immediate availability, (*iii*)
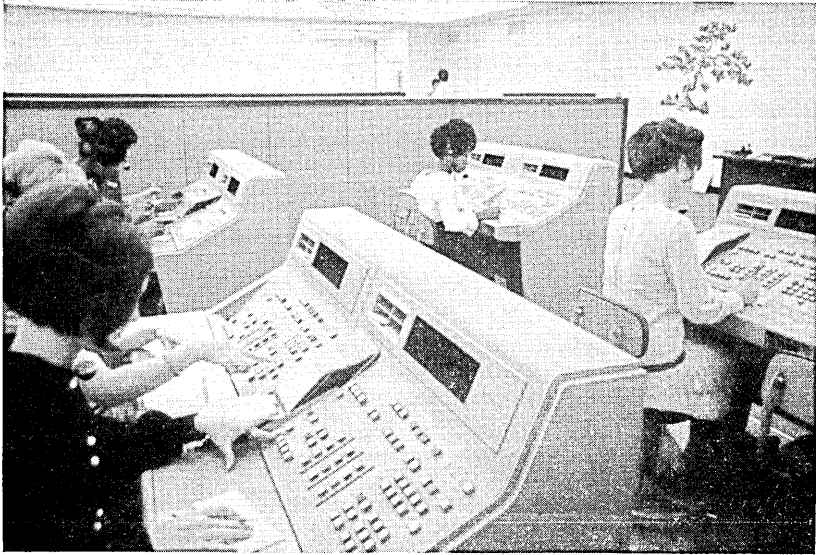
Fig. 1—Portion of a chief operator unit, TSPS No. 1.

established prices, (iv) economics inherent with large scale production, (v) increased standardization of product, and (vi) training and experience of Western Electric people for manufacturing, testing, and installation.

The equipment schematic, Fig. 4, depicts TSPS equipment components and their association. Equipment in dotted outlines constitute the No. 1A Stored Program Control (SPC), while those in solid line outline the TSPS. The heavy interconnecting lines represent transmission paths and the lighter lines emanating from the SPC 1A frames and the Communication Bus Translator are the communication bus paths that permit communication with the SPC 1A circuits, as well as the peripheral circuitry. Both local and remote operator units are shown, and these are known as position subsystems.

The typical floor plan arrangement as shown in floor plan Fig. 5 represents a maximum size office. Variables that affect the office arrangements are:

(i) Number and location of operator units equipped local, semi-remote, or remote.
(ii) Maximum number of trunks equipped.

(*iii*) Number of store frames equipped.

(*iv*) Location of transmission facilities—especially in 4-wire offices.

(*v*) Limiting conductor conditions.

II. CONNECTORIZATION AND NEW FRAMEWORKS

2.1 *Connectorization*

Connectorization of interframe cabling and some intraframe wiring is provided to simplify growth additions in working offices, installation of equipment and shop testing. In most cases cables have one end hard wired to frames and the other end terminated on plugs or connectors. These are joined to mating fixtures on connecting frames. In the case of communication bus cable pairs, they must be terminated at each end with a 100-ohm resistor to provide characteristic impedance. These Bus Terminating Resistors (BTRs) are mounted on small printed wiring boards which in turn can be mated to connectors, as shown in Fig. 2.
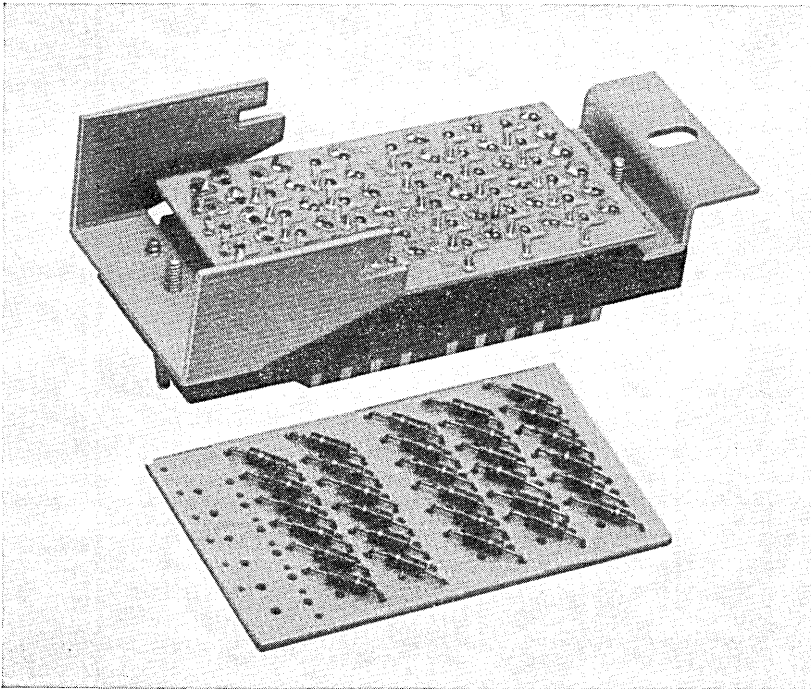
Fig. 2—Bus Terminating Resistors mounted on printed wiring board, and connector.

2.1.1 Hardware

Figure 3 shows a plug (at left) and a connector (at right). They are provided with 60, 66 and 72 terminals and include cable clamps and screw-down spindles which mate into associated locking hardware. The connector terminals are of a brass alloy and have a base coat of hard gold finish. Superimposed over the entire terminal is a soft gold finish. The plug terminals are of phosphor bronze stock with a coating of commercially pure tin.

2.1.2 Application

Connectorization is provided for all store frames. Each frame is equipped with bus switchboard cables approximately 7 foot long hard wired to transformers and terminal strips located at the top of the frame. The other end of this cable is terminated on connectors. There are also short bus local cables hard wired to transformers and terminal strips with the other end terminated on plugs. During the installation of a frame, the connectors of the switchboard cable are mounted on the cable rack above the adjacent and preceding store frame. The plugs from that preceding frame are mated to these connectors. Should
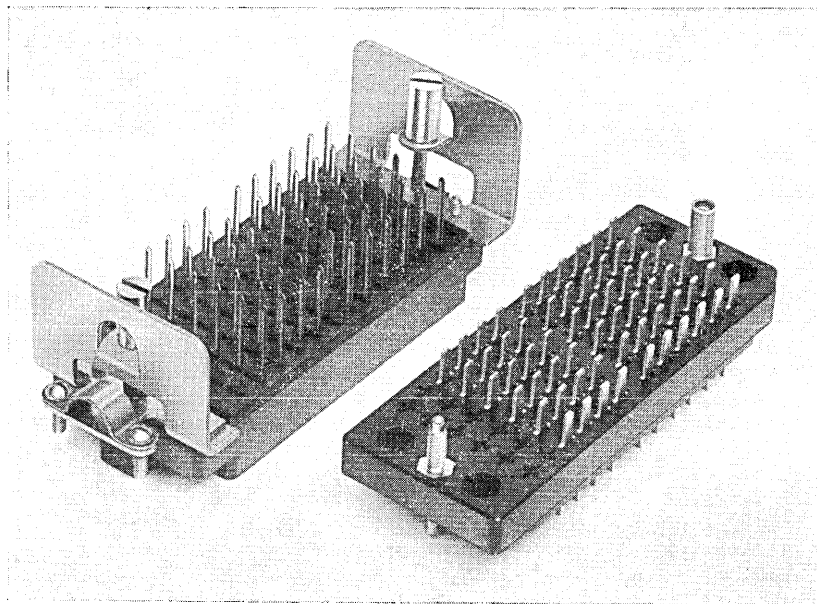
Fig. 3—Plug (left) and connector.
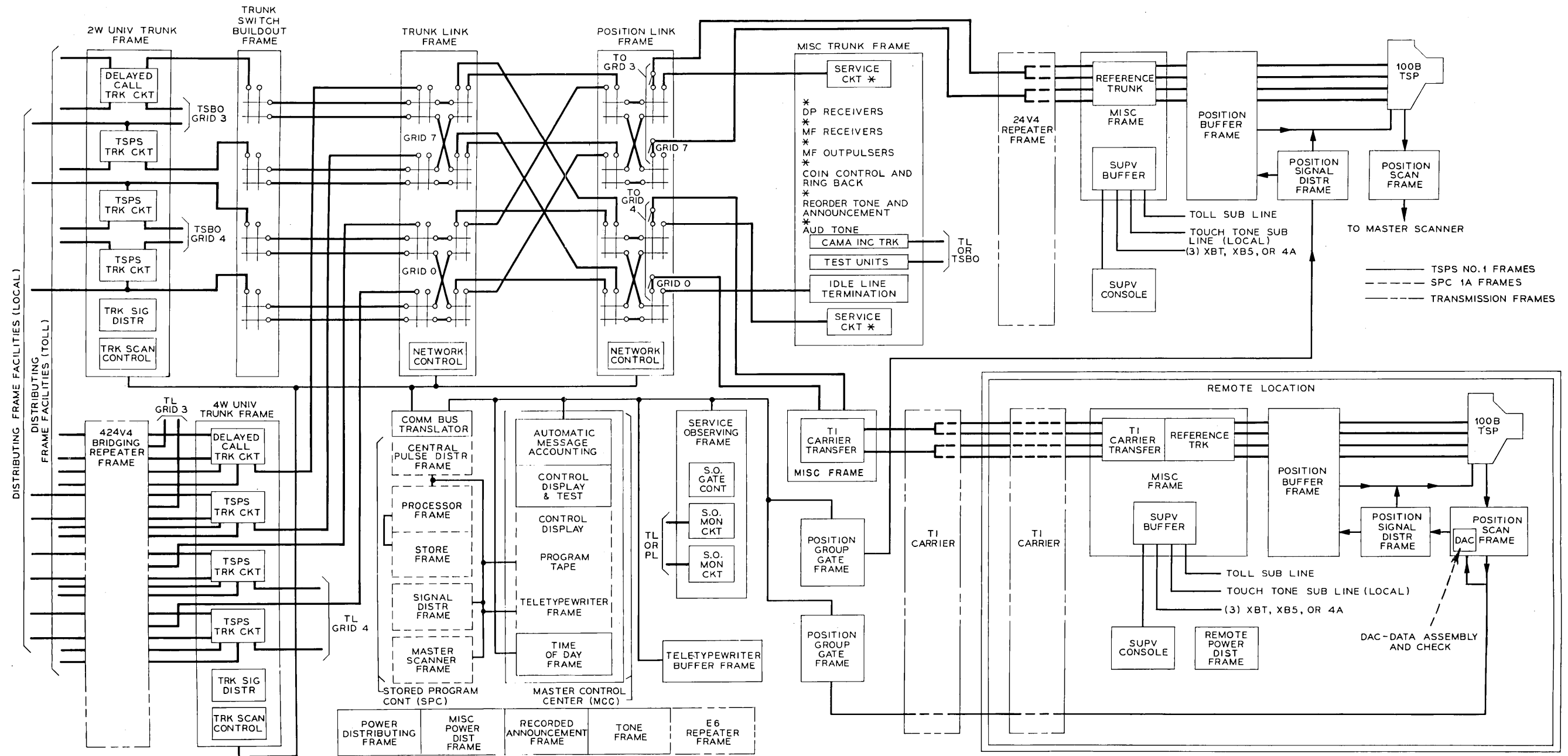
Fig. 4—TSPS No. 1 equipment schematic.
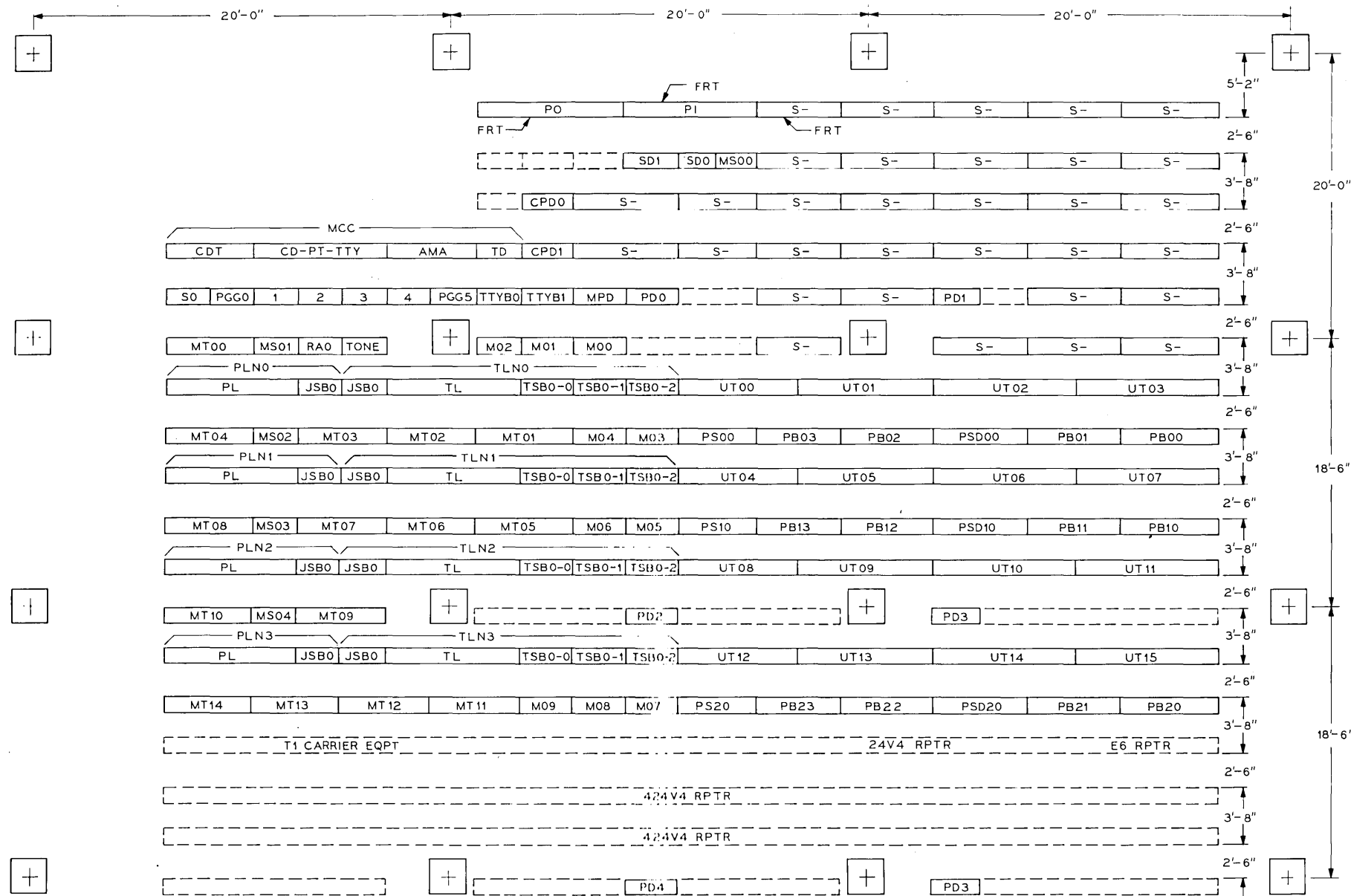
NOTE:
1. TYPICAL FLOOR PLAN FOR TSPS NO. 1 EQUIPMENT AT REMOTE LOCATION FOR TWO CHIEF OPERATOR UNITS (124 POS MAX).

PB50	PB51	PSD50	PB52	PB53	M
PB43	PB42	PSD40	PB41	PB40	M
PS40	M	PS50	M	RPD	M	M	M	M

T1 CARRIER EQUIPMENT
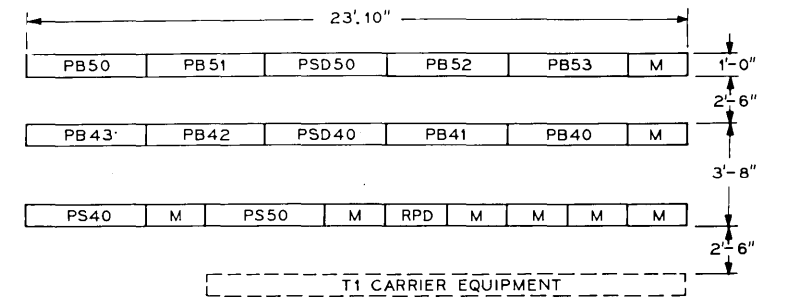
2. TYPICAL FLOOR PLAN FOR TSPS NO. 1 EQUIPMENT AT REMOTE LOCATION FOR ONE CHIEF OPERATOR UNIT (62 POS MAX).

| PS30 | M | M | M | M | RPD | T1 CARRIER |
| M | PB33 | PB32 | PSD30 | PB31 | PB30 |
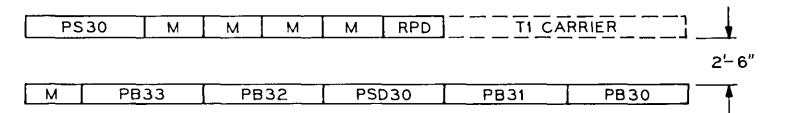
3. IN BUILDING WITH 19'-6" COLUMN SPACING. WIRING AISLES TO BE INCREASED TO 2'-0".

Fig. 5—Typical floor plan arrangement, maximum-size office.
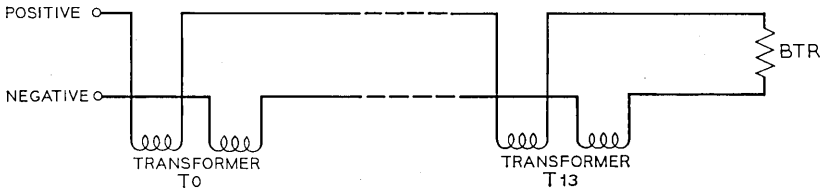
Fig. 6—Hardwired bus, setup 1.

the added frame be the last frame on the bus, BTR equipped con-
nectors will be mounted on the cable rack and the local cable plugs of
the added frame mated to these. A total of 5 connectors, 5 plugs and
512 bus leads constitute the bus connectorization at each store frame.
Another 3 connectors, plugs for 273 master scanner, signal distributor
and miscellaneous leads are also furnished. Where it is necessary to
connect other than adjacent in-line frames, connectorized patch cables
are used to provide the additional length. Metal covers are used to
protect those plugs and connectors that lay in the cable racks.

The processor frame is connectorized similar to the store frames
except they must accommodate twice the bus plugs, connectors and
leads per frame since each processor must serve the 0 and 1 buses.

Connectorization is also provided for the trunk link, position link,
universal trunk and position buffer frames.

2.1.3 *Tests*

Tests were conducted to determine the effect of connectors on the
pulses sent over a simulated system bus. The following hardwired
and connectorized bus setups were prepared.

2.1.3.1 *Hardwired Bus.* In setup 1, Fig. 6, two twisted wire pairs are
each connected serially through 14 transformers (cable receivers) to
simulate a store address bus and are terminated with a bus terminating
resistor (BTR).

In setup 2, Fig. 7, one twisted wire pair running to 24 terminal
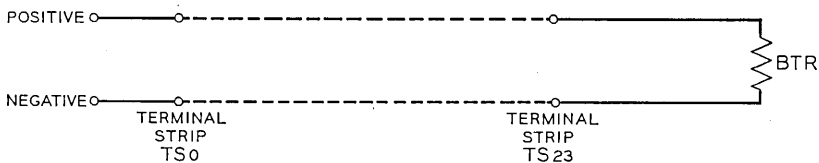
Fig. 7—Hardwired bus, setup 2.
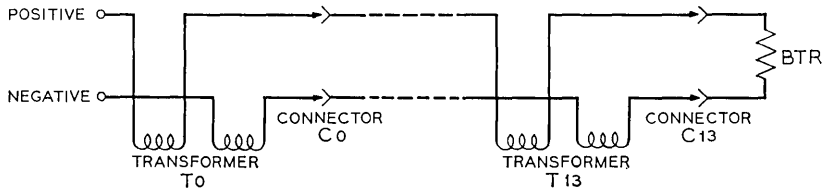
Fig. 8—Connectorized bus, setup 3.

strips to simulate a store answer bus was wired and terminated with a BTR.

2.1.3.2 *Connectorized Bus.* In setup 3, Fig. 8, two twisted wire pairs, each connected serially through 14 transformers and through 14 connectors and plugs, were wired and terminated with a BTR.

In setup 4, Fig. 9, one twisted wire pair, running to 24 terminal strips through 24 connectors and plugs, was wired and terminated with a BTR.

Setups 1 and 2 had bus lengths of 84 and 100 feet, respectively, while setups 3 and 4 were 144 and 186 feet, respectively.

A 1/2 microsecond pulse was generated and measurements were made at selected points along each bus as well as at each terminating resistor. It was found that each of the hardwired and connectorized buses had virtually identical pulses. Comparison of the pulses showed no appreciable attentuation or distortion of the pulse and no added noise on the bus. The only significant difference was the time delay between pulses of the hardwired bus and the connectorized bus. This is the result of the increased bus length due to connectorization. The maximum bus length of 100 feet per leg will accommodate 10 store frames on each of the right and left legs of a bus or, considering full duplication of buses and stores, a total of 40 connectorized store frames. This is considered a satisfactory limitation.

2.2 *Framework*

A new 3 bay frame 6 feet 6 inches long was introduced to accom-
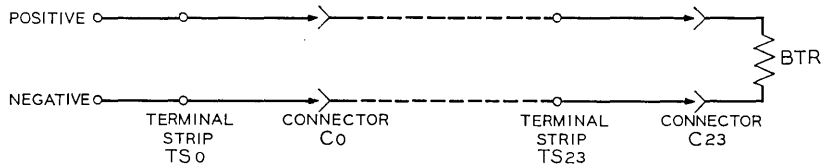
Fig. 9—Connectorized bus, setup 4.

modate large circuits. The new frameworks are used for the processor, universal trunk, trunk link, position link, and control display-program tape-teletypewriter frames. These comprise approximately 22 percent of the total frames in the basic TSPS office. The use of the new framework reduced field testing, cabling and wiring.

2.2.1 *Tests*

Initial stress calculations showed that the 3-bay frame was theoretically sound for the proposed applications. Mechanical tests simulating loading, shop handling, and field installation were also conducted and stress measurements at critical frame areas supported previous theoretical conclusions.

2.2.2 *Cable Rack Covers*

New cable rack covers were designed to reduce the number of parts and simplify installation. They are shown in Fig. 10. The covers are visually attractive and offer coverage of all connectors and plugs mounted on the cable racks. The covers (new or old) do not affect the electrical characteristics of the cable rack arrangement, as verified by considerable testing, and are furnished primarily for appearance.
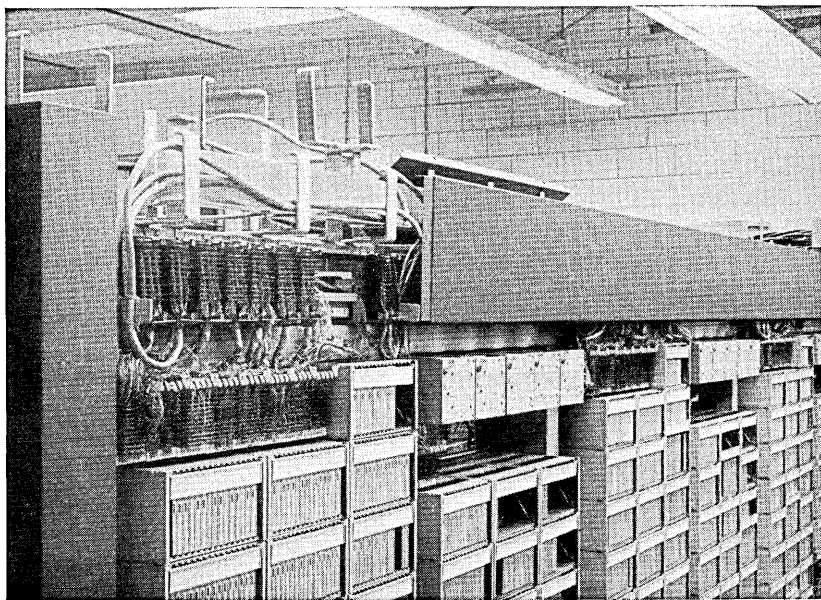
Fig. 10—New cable rack covers.

III. NEW EQUIPMENT DESIGNS

3.1 *Processor Frame*

The processor frames shown in Fig. 11 provide control for the system and execute program instructions from the stores. Each instruction requires from 1 to 10 processor cycles of 6.3 microseconds in length. To perform its duties the processor requires approximately 1780 circuit packs, mostly of the low level logic (LLL) type, and about 40,000 interconnecting leads.

In normal operation, the duplicated processors (P0 and P1) work in parallel, although only one is active and controlling the peripheral equipment. In order to insure that the processors are doing exactly the same job, a dc matching facility is used. This facility is capable of providing three matches per 6.3 microsecond cycle. It is necessary to keep wire length between matching circuits as short as possible to minimize propagation delays. Since the matching circuitry is located in the rightmost location on the frame, the two frames are placed adjacent to each other and in a front-to-back manner. This places the
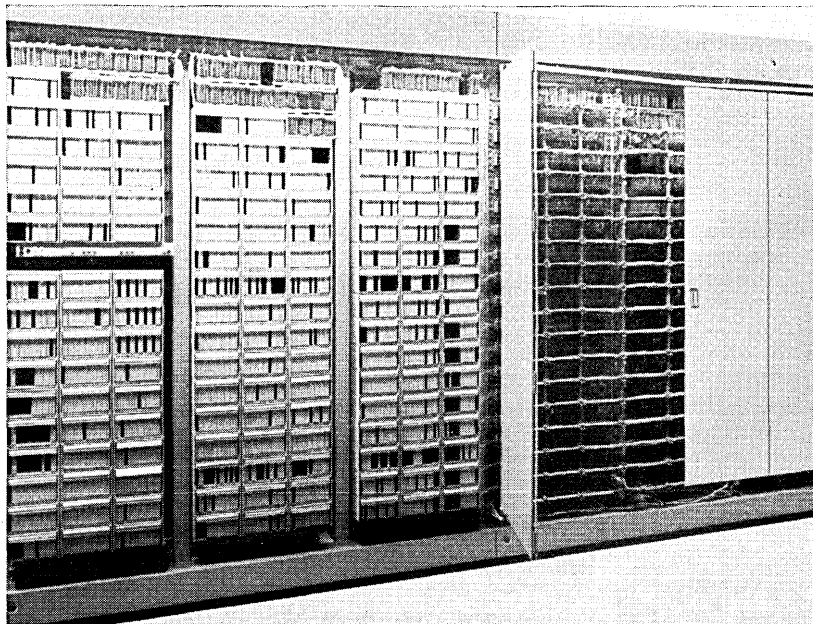
Fig. 11—Processor frames.

matching circuitry of one processor immediately adjacent to the matching circuitry of the mate processor thus avoiding the need for a duplicate mirror image processor. The use of through type terminal strips along the rightmost frame upright permits direct interframe wiring. Metal folding doors are provided to enclose the wiring side of each processor frame.

The high speed at which electronic circuits operate and their sensitivity to noise pickup along parallel lead runs required that the equipment design for these circuits be completed with careful attention to the problems and solutions involved. The design of the SPC 1A processor frame, encompassing three bays, required extreme care. The optimum location of circuit packages, correct assignment of logic gates, examination of critical leads and, in some cases, recommendations for circuit changes to reduce lead length or avoid inductive problems were considered.

3.2 *Networks*

The TSPS trunk link and position link network frames make possible the full access of any trunk to any one operator position or service circuit. This full accessibility inherently provides relatively high traffic handling capacity since all operators can be utilized as one large team. It also alleviates the need for facilities for the distribution of junctors such as a junctor grouping frame.

In the trunk and position link networks, 2 wire metallic connections are switched through four stages of ferreed switches. As many as four networks may be provided for maximum capacity. Figure 12 shows the frames, their floor plan relationship and the wiring arrangements for gaining full access of any trunk to any one operator position or service circuit.

3.2.1 *Trunk Link and Trunk Switch Buildout Frame*

The three bay trunk link frame contains the ferreed switches and trunk path selection unit, ferreed junctor switches, junctor selection path unit and other control units for 256 trunks and 512 each of "A" and "B" links. Up to three additional trunk switch buildout frames may supplement this frame depending on traffic calling rates.

3.2.2 *Junctor Switch Buildout Frame*

A one bay junctor switch buildout frame is used to extend the 512 "B" links to other trunk link networks. A similar frame is used to
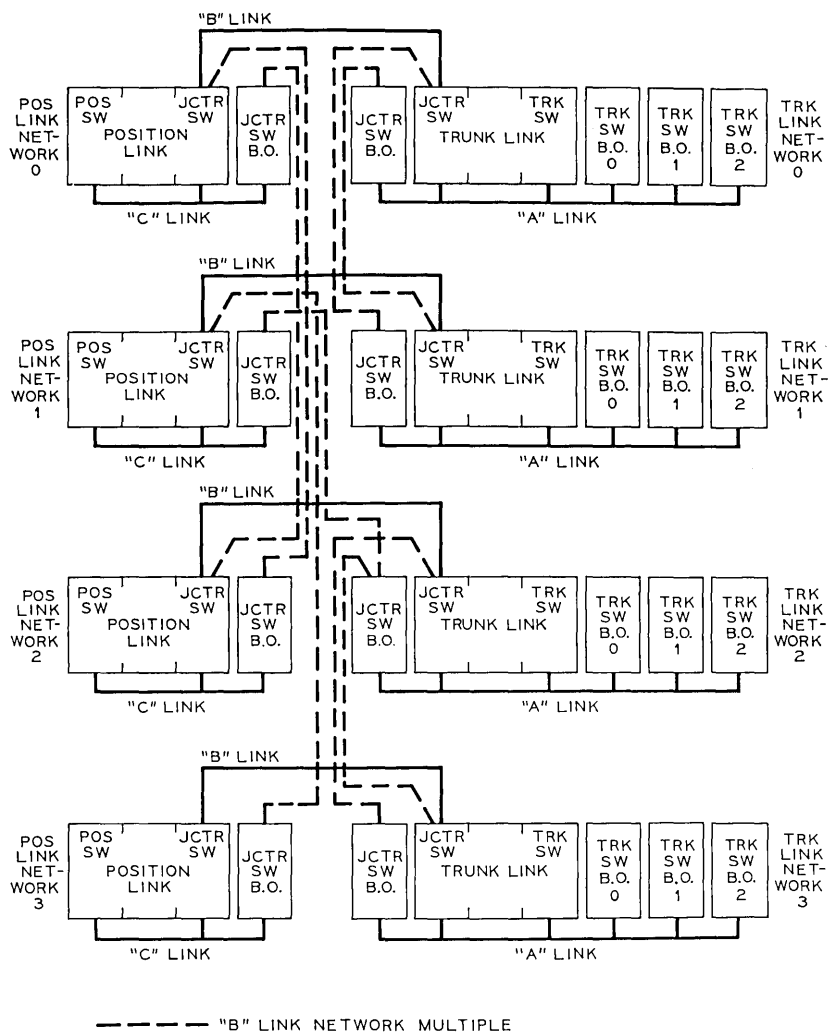
Fig. 12—Trunk and position link frames, floor-plan relationship, and wiring arrangements.

provide a full access pattern for the "B" links associated with the various position link networks.

3.2.3 Position Link Frame

A position link frame contains the position switch and operator

cut-through units, position switch path selection units, junctor switch, junctor switch path selection units and other control units. This frame accommodates as many as 96 operator position via operator cut-through circuits and accommodates 64 service circuits such as multifrequency and dial pulse receivers, multifrequency outpulsers and coin control trunks.

3.3 *Trunk Equipment*

3.3.1 *Universal Trunk Frame*

Universal trunk circuits in TSPS No. 1 are unusually simple and standardized. Their functions have been limited almost entirely to transmission and supervision. These circuits are arranged as small compact pluggable units as shown in Fig. 13. The universal trunk frame is shown in Fig. 14. The left and right bays are each arranged to mount 64 individual trunk units for a frame total of 128 units or 256 trunks. The two bays of trunks flank a control bay which contains
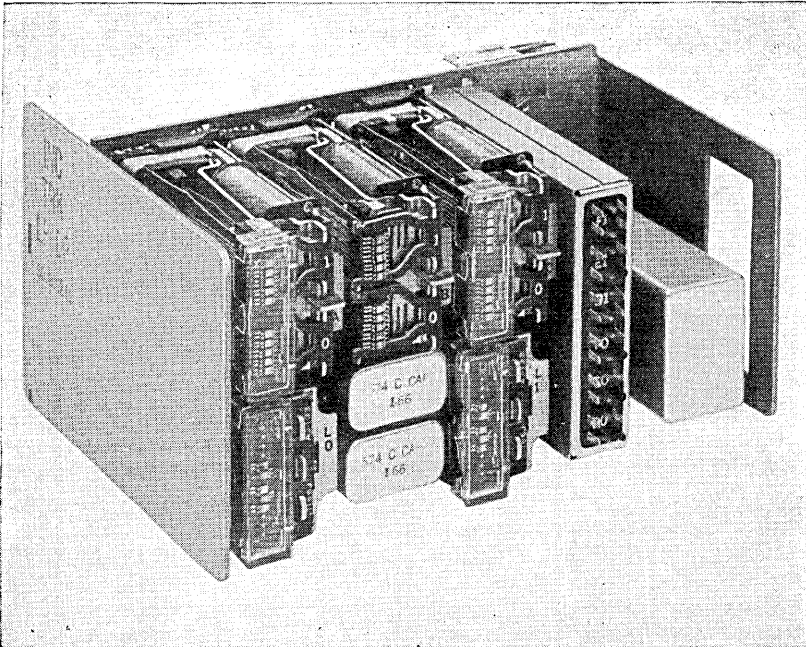
Fig. 13—Pluggable universal trunk circuit.
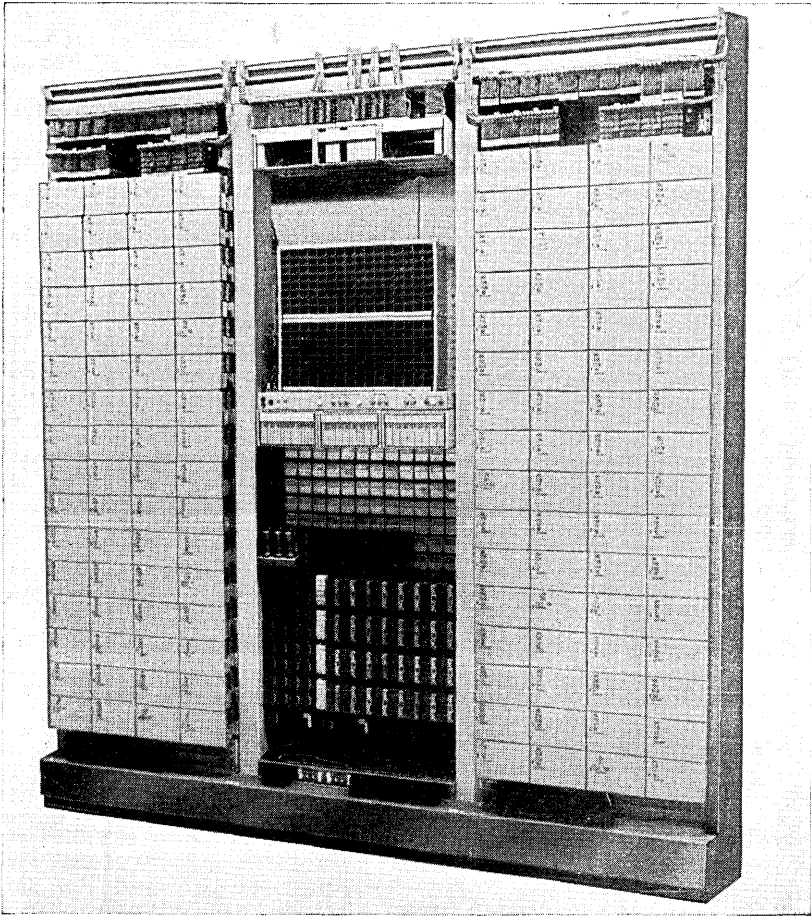
Fig. 14—Universal trunk frame.

the 512 point scanner matrix, signal distributor, and signal distributor controls for both trunk bays. A minimum of four universal trunk frames are required in each office to provide sufficient signal distributor points for service circuits. The minimum of four trunk frames also provides sufficient spread of the individual trunk groups over the network frames to insure maximum service assurance and load balancing.

All universal trunk frames are furnished with connector ended cables which mate with like designated plugs furnished as part of the trunk

link network frames. A predetermined cabling pattern between universal trunk frames and network frames eliminates the need for a trunk distributing frame. This fixed and predetermined pattern greatly simplifies the stored program data table as the trunk equipment location and its network appearance have a fixed relationship for all trunks in all offices. In this manner, bulky data tables dependent upon job conditions are not necessary. Trunk maintenance busy indicators are furnished with the universal trunk frames to aid the maintenance personnel in removing the pluggable trunk units from the frame.

3.3.2 *Miscellaneous Trunk Frames*

Trunks providing audible signals, digit reception, ringing, coin control, etc. are provided by small specialized groups of service circuits which the SPC No. 1A processor connects under program control to the trunks via the network only at times when their functions are needed. These service circuits (including outpulsers, receivers, reorder tone and announcement trunks, coin control and ringback trunks, etc.) are designed on conventional mounting plates and are mounted on miscellaneous trunk frames.

3.4 *Position Frames and Operator Console*

3.4.1 *Position Subsystem Description*

Position frames are those frames used in the flow of information from the SPC No. 1A processor to the 100B console operator and in the collection of information from the console.

There are three different modes of communicating with 100B consoles in TSPS. Their designation is indicative of the location of the position frames and the chief operator unit in relationship to the base unit. The modes of operation are:

3.4.1.1 *Local.* Position frames and chief operator unit located in the same building as the base unit.

3.4.1.2 *Semiremote.* Position frames and chief operator unit located in a distant building up to 4 cable miles (1200-ohm loop) from the base unit. Cable pairs are provided for transmission and for data exchange.

3.4.1.3 *Remote.* Position frames and chief operator unit may be located in a different office up to 50 cable miles from the base unit. Remote

operation employs the use of a T1 carrier system to carry voice and data transmission between offices.

3.4.2 *Position Group Gate Frame*

The position group gate frame contains the equipment for one group gate and the teletypewriter buffer for the traffic teletypewriter and for the remote position maintenance teletypewriter. One frame is required in the base unit for each operator unit. When positions are remote, a maximum cable length of 150 feet is allowed between the D1C bank bay of the T1 carrier system and the position group gate frame.

The position group gate is the one unit in the position subsystem which connects to the peripheral address bus, central pulse distributor and master scanner. Thus it is the one unit communicating intimately with the SPC No. 1A and all subsystem orders must first pass through it. In the local case it acts primarily as a translator and time buffer to accept high-speed binary information from the address bus and deliver expanded information (in the form of 1 out of 8 codes) at slower speed to the next subsystem unit, the position signal distributor. In the remote case a T1 carrier system and a special checking circuit (Data Assembly and Check Circuit) are interposed between the group gate and the signal distributor. In this case the group gate does not modify the binary data received from the bus but still acts as a time buffer and additionally provides certain control signals needed by the T1 carrier.

The teletypewriter buffer, which is mounted on the position group gate frame receives high-speed data from the processor and transmits time buffered signals to control the teletypewriters associated with the chief operator units. Traffic TTY machines are furnished for all chief operator units regardless of location while the remote position maintenance TTY machine is optionally furnished only at remote or semiremote chief operator units.

3.4.3 *Position Signal Distributor Frame*

A position signal distributor frame is arranged to mount the equipment for the positional signal distributor circuit. One frame per chief operator unit is provided.

The position signal distributor receives data from the position group gate at the local, or semiremote installation. Information from the data assembly and checking circuit is received at a remote instal-

lation. The data is used to select a particular set of cut-through relays on the position buffer frames and also select a particular output path through these cut-through relays. The outputs, in turn, operate or release the relays of the position buffer or the position signal distributor applique. The outputs of other cut-through relays control the digital displays at the operator positions.

3.4.4 *Position Buffer Frame*

The position buffer frame is arranged for 16 operator positions. A minimum of two and a maximum of four frames may be equipped in each chief operator unit. The first buffer frame accommodates the equipment for the first 14 position buffer circuits, the position signal distributor applique circuit, and other equipment common to the chief operator unit. Other buffer frames contain 16 position buffer circuits. The position buffer frames are located adjacent to their associated signal distributor frame.

Associated with each position are miniature wire-spring relays mounted on pluggable printed circuit wiring boards as shown in Fig. 15. These boards, mounted on the position buffer frame, provide the means to light lamps on a particular operator position or control other circuits associated with the position. The relays are either magnetic latching or nonmagnetic latching.

Ten of the A-type circuit packs containing a total of 68 relays are furnished for each position. For a large office of 280 positions, 2800 circuit packs containing a total of more than 19,000 relays are provided. This application is the first large-scale use of miniature wire spring relays.

3.4.5 *Position Scan Frame*

The position scan frame is arranged to mount the following circuits: (*i*) position scanner and gate, (*ii*) alarm sender, (*iii*) data assembly and checking (for remote operator units).

The main function of the position scanner and gate circuit is to transmit signals to the master scanner and in turn to the SPC whenever a key at any operator position is depressed. The transmitted signal identifies both the key depressed and the position at which the key is located. The equipment for this circuit occupies about 35 percent of the 2-bay framework.

The alarm sender circuit provides a means by which alarm and maintenance indications from circuits which comprise a chief operator
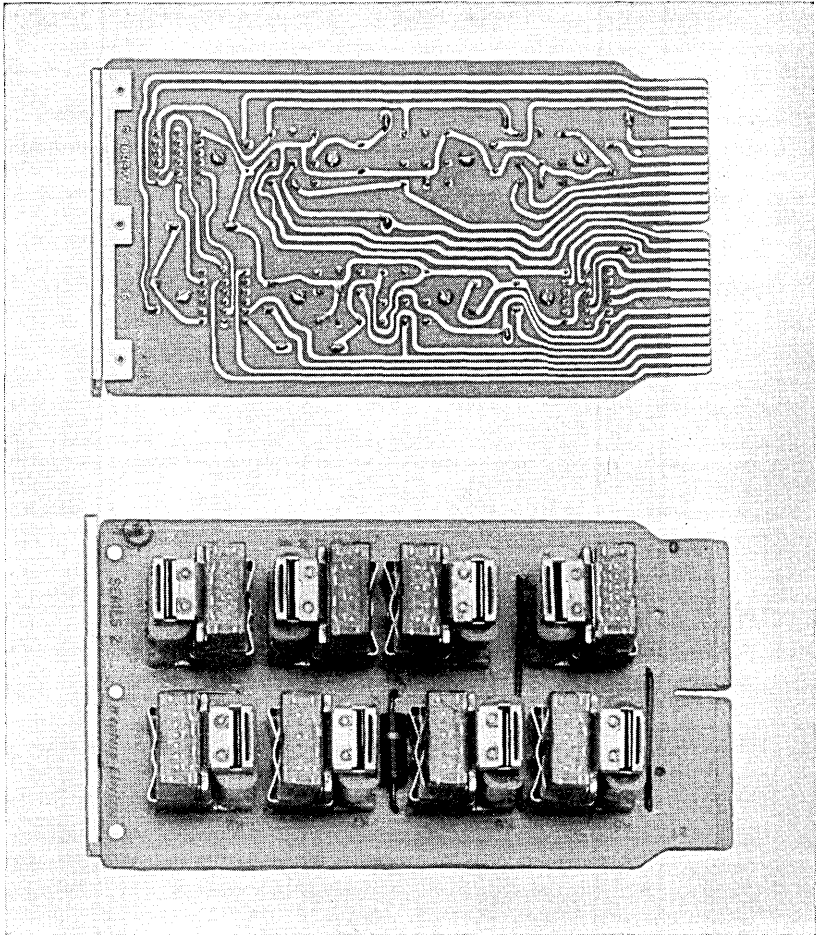
Fig. 15—Miniature wire-spring relays mounted on pluggable printed circuit wiring boards.

unit can be transmitted to the stored program control. It also provides input circuitry connected to a 10-position, 5-module thumbwheel switch for the maintenance person to send maintenance diagnostic requests to the SPC. The equipment for this circuit occupies about 30 percent of the 2-bay framework.

The purpose of the data assembly and check circuit, and T1 carrier facilities is to provide a data path from the group gate to the remote position signal distributor frame. The data assembly and check circuit

receives the information from the T1 carrier, checks for a valid transmission, and translates the word to 1 out of 8 codes before passing it on to the position signal distributor.

3.4.6 *100B TSP Console*

The 100B operator console, as shown in Fig. 16, was modeled after the 100A Traffic Service Position used in Crossbar Tandem. The section (2 positions) framework is designed specifically as a support for the keyshelves, cable rack, and equipment mountings required at each position. The basic section is an all-welded steel frame finished with textured vinyl. A completely welded assembly is composed of a rear panel, two-end frames, a lockrail assembly and a cable rack. The major items fastened to the welded frame are a keyshelf assembly, end covers, floor supports and foot rests. The keyshelf assembly is the most important and intricate portion of the overall framework as-
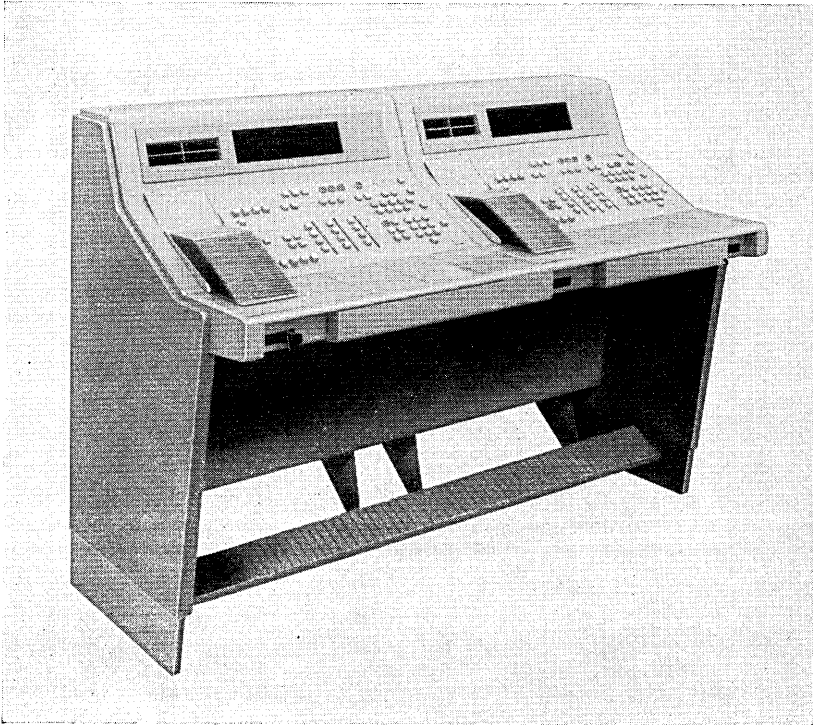
Fig. 16—The 100B operator console.

sembly and consists of a die casting with an aluminum mask that is also finished with textured vinyl. A total of 116 round universal mounting holes for keys or lamp sockets are cast into the shelf as well as other rectangular holes for mounting ticket boxes and a digital display housing. The keyshelf mask conforms precisely to the top surface of the die casting. Square holes are located at only positions equipped with keys or lamp sockets.

A numerical display unit assembly is used for various number displays, i.e., "calling number, called number, coin calls time and charges, time-of-day and special billing number." The assembly consists of a printed wiring board, a plastic shield and a plastic housing. Figures 17 and 18 show the board and the shield mounted on the rear of the board.

3.5 *Master Control Center*

3.5.1 *Maintenance Center* (see Fig. 19)

Inasmuch as the TSPS system uses the SPC No. 1A as a component part of the system, provisions have been made to supplement the SPC-1A CD-PT-TTY frame with three frames of the TSPS design to provide a unified master control center. At this single location these frames allow for the monitoring of system status, for communicating with the system and for the performance of routine test functions.
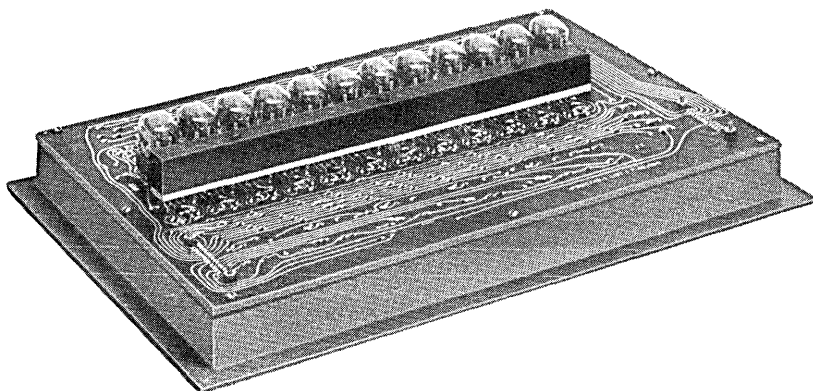

Fig. 17—Printed wiring board of the numerical display unit assembly.
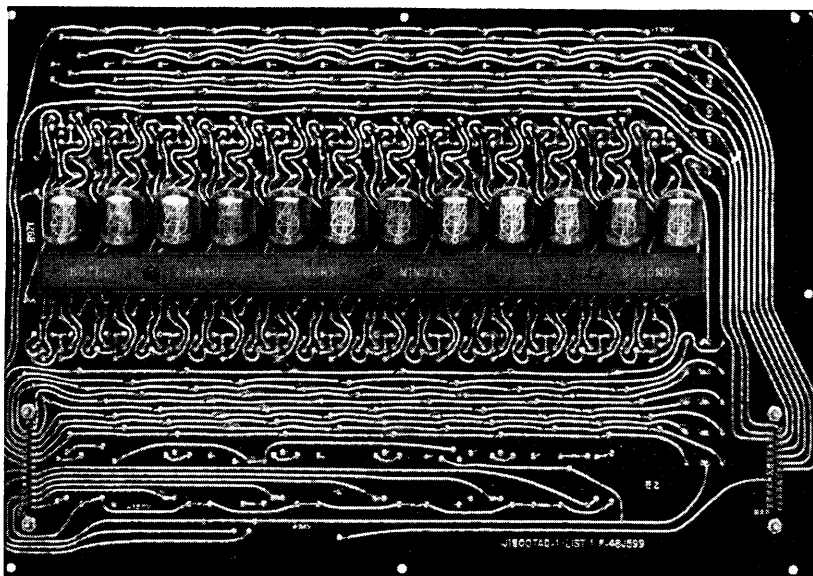
Fig. 18—Shield mounted on the rear of the numerical display unit assembly.

3.5.2 *Control Display-Program Tape-Teletypewriter Frame*

The control display-program tape-teletypewriter frame accommodates the equipment for the program tape unit used in loading the programs, the maintenance teletypewriter used for communicating and manually influencing the program, and a display panel showing the status of equipment units. Keys for emergency actions form a part of the control-display panel.

3.5.3 *Control, Display, and Test Frame*

The major functions of the control, display, and test frame are to test trunks and positions and to display system status. Trouble conditions are displayed for the peripheral frames which are common to the office and for those which are specific to a chief operator unit. Transmission noise and voltmeter tests are made on individual trunks and traffic service positions that are in trouble. Other frame component units are mounted below the writing shelf on the right hand bay. A removable cover is located in front of these units to protect them from damage. A telephone set unit mounted below the writing shelf on the left-hand bay provides equipment for connecting a telephone set which
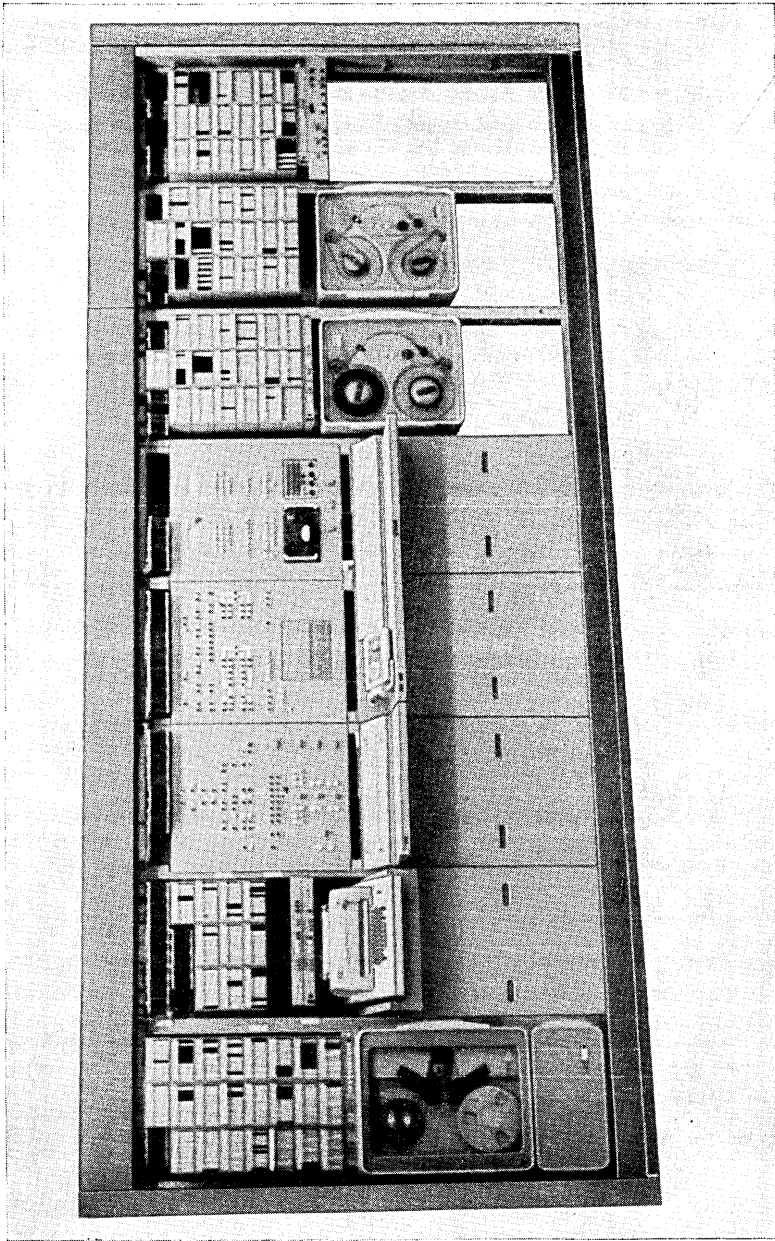
Fig. 19—Maintenance center.

is located on the writing shelf. The control display and test frame is arranged to connect this telephone set to a 1A2 key telephone system.

3.5.4 Automatic Message Accounting Frame

The automatic message accounting frame is a component frame of the master control center. This frame includes the 9-track AMA circuit and 2-tape transports, one in each bay.

The tape transport reel can accept up to 2400 feet of 1/2 inch magnetic computer tape and continuous billing information for one day for all but the most heavily loaded offices. The transport has 9-track write and read heads and a write-read tachometer head for measuring tape speed. An erase head removes all previously recorded data to ensure a clean record. A photocell detects the end of a tape reel and three microswitches monitor tape tension.

3.5.5 Time-of-Day Frame

The time-of-day frame derives tenths of seconds, seconds and minutes from the 60-cycle 120-volt ac line. This information is transmitted in binary form via the communication answer bus to provide time-of-day information for the AMA recorders, for displays in the 100B traffic service positions and to the SPC processor.

The time-of-day frame contains redundant clock circuits and a control panel. The latter contains a clock control and readout unit consisting of switches, keys, lamps and a visual readout that indicates the time-of-day when activated by a readout switch.

3.6 Test and Evaluation Equipments

3.6.1 General

To facilitate the introduction of TSPS as a new system, a new family of SPC No. 1A and TSPS program debugging and troubleshooting test sets were devised. Through the extensive use of integrated circuits it was possible to make the test sets portable.

3.6.2 Transfer Trace Test Set

A transfer trace test set as shown in Fig. 20 makes possible recording up to 600 program transfers prior to a significant event. This set can be arranged to follow individual instructions as they are executed by the processor for detailed analysis of program flow. It can also be used to count the frequency of particular effects as they occur in the system.
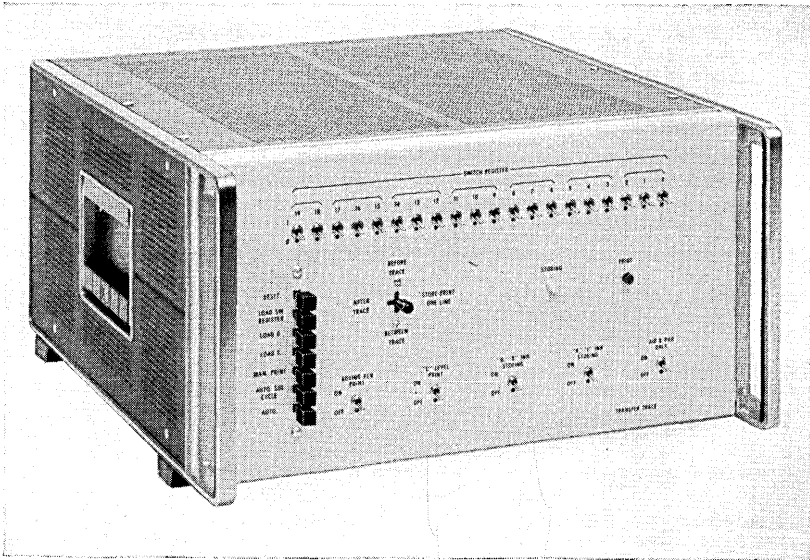
Fig. 20—Transfer trace test set.

3.6.3 *Portable Matcher and an Auxiliary Logic Box*

A portable matcher provides a wide range of matching functions for use in triggering the dump store freeze[3] and/or the transfer trace printout.

The auxiliary logic box supplements the portable matcher logic functions when necessary.

IV. POWER PLANT, DISTRIBUTION AND GROUNDING

4.1 *Power Plant and Distribution*

The −48 volt and +24 volt power from dedicated plants is delivered to two or more power distributing frames (PDF) in the switchroom. There, each battery supply is filtered and the battery and ground pairs are branched through banks of fuses that feed the various frames. At each switch frame served, the battery lead from the PDF is connected to the frame fuse panels through a filter unit.

In remote or semiremote offices similar −48 and +24 volt plants are usually provided although the −48 volt power may be supplied from an existing power plant in a remote building. A remote power distributing frame (RPDF) is provided and has a 135 ampere peak

capacity for each voltage and serves a maximum of two chief operator units.

A $+130$ and -130 volt dc supply converted from a -48 volt power supply is used for coin control, NIXIE® lamp displays, etc. Power from this source is delivered to fuse panels on a miscellaneous power distributing frame (MPDF) or an RPDF and distributed from there to all frames which require this power.

4.2 *Grounding*

Special frame insulating practices are followed in TSPS to avoid electrical interference from stray ground potentials. Switch frames are bonded to the building grounding system at a single point. All frames and cable racks are insulated from the building at other points. The frames are connected together by ground feeders between the frames and the PDFs. In addition, a number 6 copper wire is run in the cable rack to provide a bonding network to all frames. A ground lead is run between the TSPS ground system and the central grounding point for central office building ground.

V. ENVIRONMENT

The TSPS System is designed to operate in the standard central office environment for electronic equipment. This dictates that the normal ambient operating temperature of the office should remain between 40° and 100°F.

Determination that a complete system would operate under emergency conditions was accomplished by subjecting a complete office to a heat test. The main objective of the heat test was to make certain that the TSPS and associated equipment would operate in an ambient temperature of 115°F without major failures.

In cooperation with New Jersey Bell Telephone Company, the first complete TSPS office was operated in ambient temperatures of 100°F and then raised to 115°F. This office is located on the first floor of a multistory building in Morristown, N. J. Two sides of the office were enclosed by outside walls and the other two sides separated, by floor to ceiling asbestos walls, from a crossbar tandem machine. This office had two air duct systems available for use in the heat tests. The one system had access to outdoor air so a heat source was placed at this location and heat was supplied by this duct system. The other duct system with its air-conditioning equipment was used to modulate the temperature of the TSPS equipment. The air distribution within the

room was redistributed to reduce the temperature gradient across the office. In this manner of controlling temperature the associated equipment frames were also tested at the higher temperature.

Activation of the TSPS office was accomplished by using load boxes, and records were kept of system performance. Equipment component and room air temperature were automatically measured with 40 thermocouples attached to recorders. Thirty-six (36) thermometers were also used to visually measure other temperatures. Figure 21 depicts the schedule and resultant temperature readings.

Component temperatures indicated that one type of capacitor was operating near its critical temperature. Several hardware problems not previously found by other means of analysis were quickly brought to light by heat testing the complete office. Germanium transistors in circuit packs in the position scan frame were found to fail at about 105°F. These packs were subsequently redesigned with silicon transistors. However, upon reducing the temperature these cards returned to normal operation. Component temperatures ranged from 130°F to as high as 257°F under maximum temperature operations with very few failures during the test.
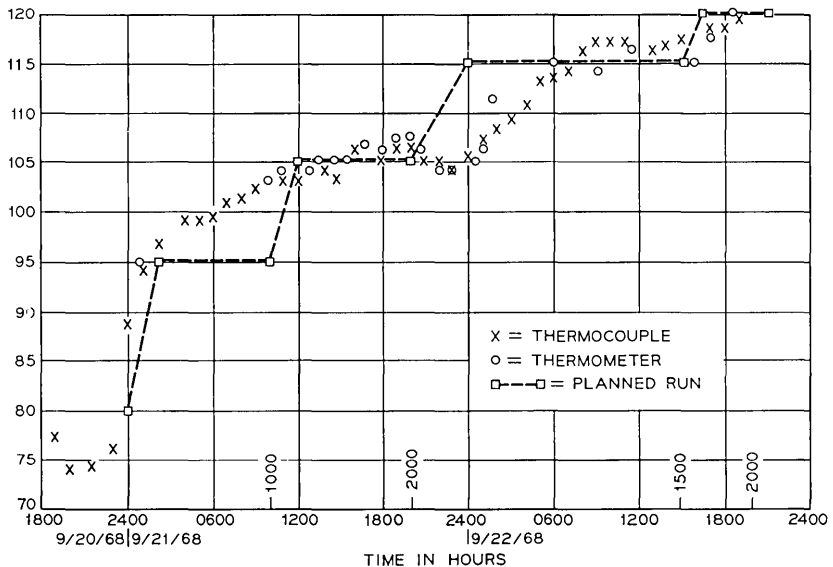

Fig. 21—Schedule and reading for temperature tests.

VI. SUMMARY

The Traffic Service Position System No. 1 physical design has made a sizeable contribution toward the achievement of high performance and low maintenance. TSPS equipment and operating costs are reasonable and therefore will reach a wide market. Many additional, attractive features and improvements, deferred for schedule reasons, are to be added as quickly as possible to further enhance this system.

TSPS No. 1, to an extent, was based on No. 1 ESS designs and this resulted in economies in development time and effort as well as achieving a high degree of standardization in manufactured product. Innovations in the form of new frameworks and connectorization offer ease of installation and facilitate growth in a working environment. These physical designs and new color applications result in harmonious environment which reflects system design intent.

REFERENCES

1. Ferguson, J. G., Grutzner, W. E., Koehler, D. C., Skinner, R. S., Skubiak, M. T., Wetherell, D. H., "No. 1 ESS Apparatus and Equipment," B.S.T.J., *43*, No. 5 (September 1964), pp. 2355–2439.
2. Funck, R. H., "Designing the Traffic Service Position Console," Bell Laboratories Record *42*, No. 6 (June 1964), pp. 210–212.
3. Heller, K. A., Schmitt, D. A., and Taylor, R. M., "TSPS No. 1: System Testing," B.S.T.J., this issue, pp. 2711–2728.

TSPS No. 1:

System Testing

By K. A. HELLER, D. A. SCHMITT and R. M. TAYLOR

The results of system testing against a set of performance criteria indicate the readiness of a new installation for service. In the first installation of TSPS No. 1 at Morristown, N. J., system testing also had to accomplish a design shakedown concurrent with final design and program debugging efforts at the Holmdel laboratory test model. This article describes the overall efforts and test facilities involved in the initial shakedown.

I. INTRODUCTION

In a typical TSPS No. 1 installation, the Western Electric Company performs a series of installation tests which check out the equipment as received from the factory and the equipment interconnection cabling as installed in the field. Extensive use is made of manual buzz through testing and X-ray programs.[1] Then system tests are conducted to demonstrate that the system is complete and operable. Initially basic system functions related to handling a small call load are shaken down. Then all system features are tested in the presence of the call load background. Finally, continuous operation is imposed wherein the system is allowed to run under typical environmental conditions with call load boxes being used to simulate traffic. System performance during continuous operation must satisfy a set of performance criteria (described in Section III) to be acceptable for turnover to the Bell System Operating Company. After turnover, operating company craftsmen tie the new system into the existing Bell System plant, determine that connecting offices function smoothly with it, run it continuously in accordance with telephone company discipline, and for training purposes perform all routine functions. On completion, the system is cut over into service.

The system tests applied at the first TSPS No. 1 installation at Morristown, N. J., were more exhaustive than would be used for sub-

sequent installations since, in addition to demonstrating a complete and operable installation, they also had to demonstrate an adequate design. Also the performance criteria for turnover to be used in subsequent installations had to be established on the basis of experience gained with the system at Morristown. As a result, considerable engineering judgment had to be exercised in deciding when to cut over the first installation. The adequacy of the system's performance to give a commercial grade of service in spite of residual design and installation bugs had to be balanced against economic and service demand pressures for an early cutover.

System testing in the first installation turns out to be an engineering feat. The planned tests reveal design bugs which necessitate software and/or hardware changes. In addition, the effectiveness of the tests devised for the newly designed system is not always predictable, with the result that the tests themselves have to be revised. The time and activity required to design and install corrective changes interferes with further testing. Almost constant re-evaluation of test plans, schedules, and status are necessary.

In the year prior to the Morristown TSPS No. 1 cutover, a total of 177 necessary hardware changes were installed. Most of these were minor, involving only a few wire changes. Some, however, were major, such as complete replacement of store frames, bus connectorization, and several significant changes in the position subsystem equipment. In the same year over 2000 trouble reports were issued that resulted in software changes.

Software debugging and utility facilities (see Section 2.2.1) were left in the Morristown system until three months before cutover. In addition to providing debugging aids, these facilitated timely insertion of program changes. After removal of the utility programs, special generic recent change programs were used to implement program changes. These had been provided to facilitate emergency changes in working offices and proved useful for pre-cutover changes at Morristown. As program overwrites or changes accumulated in the program memory, it became necessary to reassemble the programs to permanently incorporate the changes and produce new issues of program load tapes and listings. New program issues were loaded in the test model at the Holmdel, N. J., location of Bell Laboratories, and after initial shakedown were loaded at Morristown. Program reload and recovery of the system to the operational state that existed prior to the reload frequently took several days, particularly in the early stages of system testing.

1.1 *Program Administration*

Because of the volume of program change activity, a program change committee was formed about a year before the Morristown cutover. As programs became debugged, they were "frozen," and thereafter changes could be made only with the change committee's approval. To insure tight control, the program "source" card decks for all frozen programs were turned over to a program support group which had the responsibility thereafter for reassembling frozen programs to incorporate only approved changes and for producing new program issues.

When the need for a program change became known, the programmer would originate a correction or program overwrite, which he would prepare in the form of a punched card deck. By means of the utility system he would insert the overwrite into the system's memory on a temporary basis for testing. If good, he would submit the overwrite deck, a correction report, and an alter deck to the change committee for approval. The alter deck was required for incorporating the change in the next program reassembly. If approved, the overwrite would be inserted into the system memory via the utility system on a permanent basis. When a sufficient number of overwrites had accumulated, the new program issue was produced and loaded.

To minimize programmer effort and human error, the utility and program assembly systems were designed so that overwrite and alter decks could be identical in content and in addition could be in symbolic format except for certain items that must be in absolute code format for the overwrite deck.

Another useful programmer aid that was built into the assembler/loader system is the ability to generate program listings that are consistent with the state of the system's memory as modified by overwrites. Specifically, the instructions are listed as corrected by the overwrites and in the sequence executed, but the address that appears for each instruction is the absolute location of the instruction in memory. This listing is particularly useful to the programmer while overwritten programs reside in the system memory. Later, when the programs are reassembled for a new issue, the instructions are resequenced so that successive instructions occupy successive memory locations.

II. PROGRAM DEBUGGING

Debugging of the Stored Program Control-TSPS software package

occurred in three stages, each of which required a different set of debugging tools and techniques:

Stage 1—Strip or T-cart debugging
Stage 2—Utility debugging
Stage 3—Functional testing

Figure 1 shows when these three stages were employed and the fact that they overlapped in time. At most times during the entire debugging interval, the tools and techniques of all three stages were in use. The transition from one stage to the next was not prearranged but rather represented a natural evolution. Actually the three stages were followed for each program. The fact that programs became available for debugging at various times and underwent extensive changes at times is the primary cause of the extensive overlap.

2.1 Strip or T-Cart Debugging

Strip debugging is the process of checking basic "strips" of program instructions without testing the interconnections between the strips. Such a test is normally conducted by setting up the initial conditions required by the strip being tested, executing the strip, and then observing the terminal conditions.
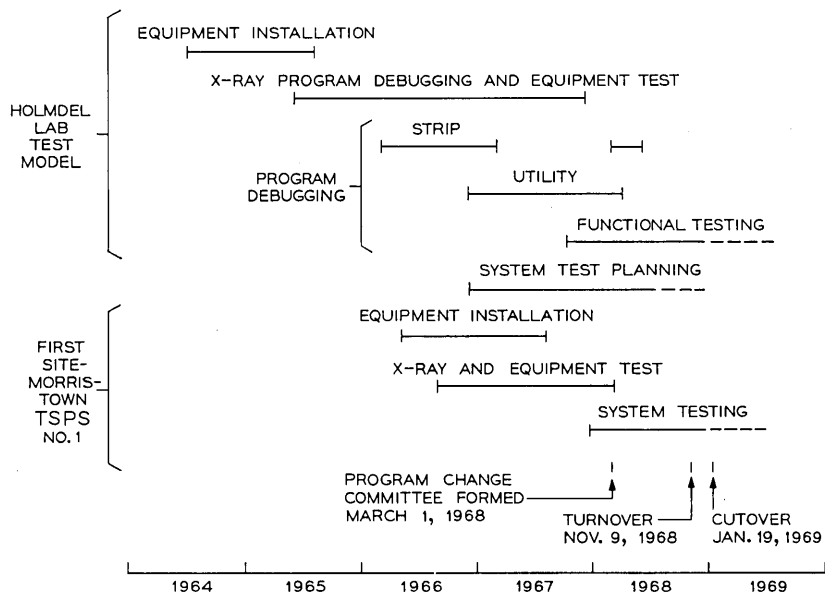

Fig. 1—Test model and first site installation and test schedule.

Strip debugging was used to check the basic SPC-TSPS control programs and the utility package which was later used for utility and batch-mode debugging operations and functional testing. The only debugging tool required for strip debugging is the test console (T-cart) shown in Fig. 2. All initial conditions are established through the T-cart instruction and data insertion keys. The instructions being tested are then executed one at a time by successively operating the cycle key. The results are observed on the various display lamps. It is also possible to execute the instructions at normal machine speed by operating the run key. For this case, the T-cart is equipped with four hardware matchers which can be set up to stop the machine when:

(*i*) a specified address is accessed for reading, for writing, or for program execution;

(*ii*) an address within or without a specified range is accessed for reading, for writing, or for program execution; or

(*iii*) 20 bits of data being written or read from temporary memory match pre-specified data.

When the machine is stopped by one of the matchers, the results of the program operation can be examined by reviewing the T-cart display lamps.

Strip debugging enjoyed heavy use only during the very early part of the debugging interval. As more programmers desired access to the laboratory model, the obvious inefficiencies of the manual strip method began to seriously retard progress. However, at later times during the debugging interval programmers sometimes found it necessary to fall back on the strip method. This was particularly true during the testing of the processor and store fault recognition programs[2] which were not as amenable to utility debugging.

2.2 *Utility Debugging*

In the utility debugging mode, the laboratory model is under control of the "utility program package." This is a collection of programs that coordinate the execution of each debugging job. All commands to the utility package are entered through a card reader which is connected to the laboratory model through a special interface. All output is directed to a high-speed printer, which is also specially interfaced to the laboratory model.

To use the utility debugging system, the programmer prepares "debugging jobs," in the form of punched-card decks. The card decks
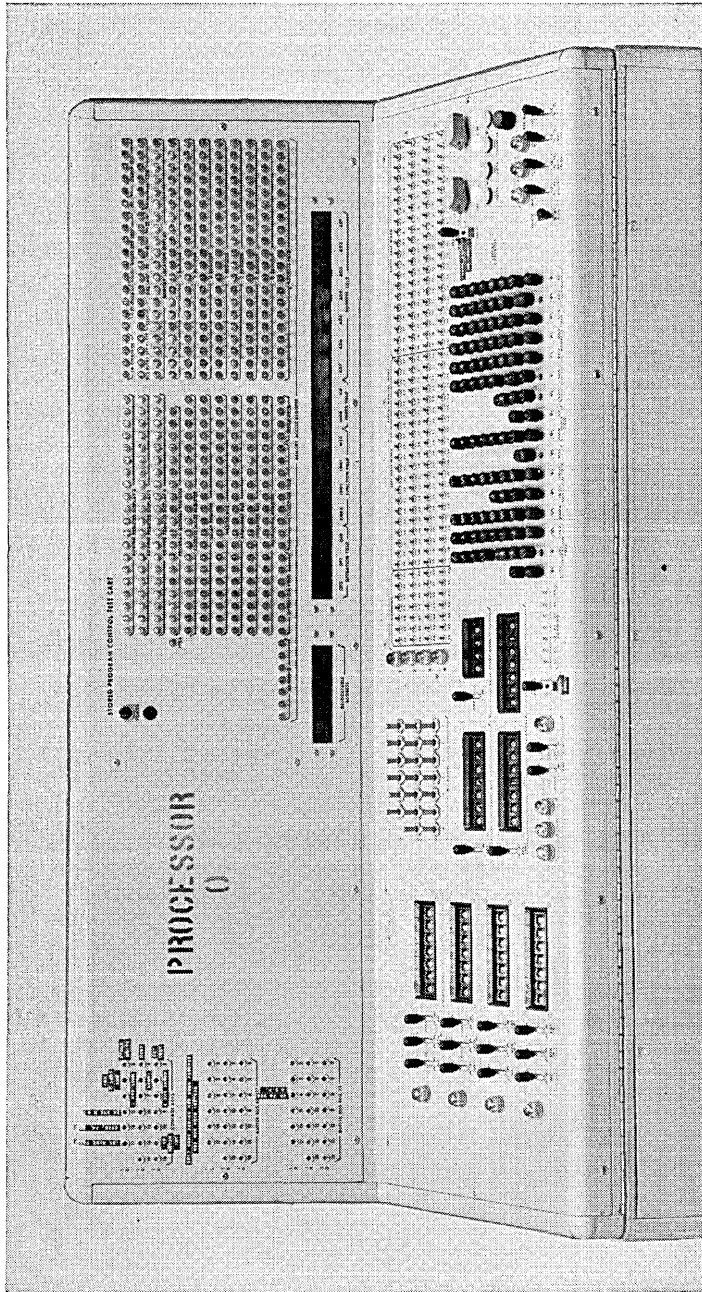
Fig. 2—Test cart.

contain control or command cards which are read and executed by the "utility program." These commands establish initial conditions, generate printouts of specific registers and memory areas, and control the execution of the instruction strip being tested. In other words, all of the manual functions required for manual strip debugging are prespecified on punched cards and executed automatically at high speed.

Of prime importance to the utility mode are the matchers which trigger execution of the various commands as prespecified in the punched card decks. The T-cart matchers can be used for this purpose. To increase the number of available matchers and to avoid the manual setup time for the T-cart matchers, twelve pseudo matchers were provided as part of the utility system. The pseudo matchers permit the programmer to specify the program address to match in his debugging deck. The utility program changes the instruction at that address to one which causes the processor to interrupt the normal sequence of program instructions at A-level or the highest level interrupt.[2] Whenever the program under test reaches that address an interrupt occurs, control is transferred to the utility program, and the required utility functions are executed. Then the original instruction which had been saved by the utility program is executed and control is transferred back to the system program.

Because the utility system enabled the programmers to completely plan and specify their debugging jobs on punched cards, a batch mode type of debugging operation was made possible.[3] In the batch mode the programmers submitted their relatively short debugging jobs to the laboratory support group in lieu of requesting machine time to run their own jobs. The support group batched the submitted jobs and ran them in rapid sequence, returning the results to the programmers for analysis. This technique makes more efficient use of machine time than if each individual programmer conducted his own debugging runs and further permits a programmer to get several runs a day, which might not otherwise be possible.

The batch mode technique was first tried on the No. 1 ESS ADF (Arranged with Data Features) test model in New York City. Its success there prompted its use in TSPS No. 1 where it also proved to be successful.

For the more complex debugging runs requiring more time per run or programmer presence at run time, the programmers would sign up for machine time, which would be scheduled by the laboratory support group. Even for those cases where programmers signed up for

machine time they still prepared their debugging runs and punched their utility control cards in advance to take full advantage of the utility system and maximize machine time utilization.

In the final analysis, maximum debugging efficiency (orders debugged per unit time) is achieved only if the programmer uses the machine to get the results of pre-planned debugging runs and does his analyzation of results at his desk. Machine time is wasted when a programmer signs up for a large block of time and then spends a large part of the time analyzing results to plan the next step. The success of the batch mode operation is largely attributed to the fact that it overcomes loss of machine time due to analyzation activity by the programmer while he is assigned exclusive use of the machine.

2.2.1 *Utility Package*

This section gives a detailed description of the functions provided by the utility program package. The utility programs are loaded into spare SPC memory areas and, where necessary, temporary linkages are provided to the generic program software.

The utility package is divided into four main parts: (a) initialization, (b) command processor, (c) administration, and (d) overwrite control. Throughout the system debugging interval, these programs underwent continual modification to improve their speed and power. Many improvements were suggested and implemented as a direct result of debugging experience so that the final versions bore little resemblance to the originals. The following paragraphs describe the final versions.

2.2.1.1 *Initialization Program.* The utility initialization program acts as a "system reset." That is, whenever the operator actuates the manual interrupt (MINT) key on the T-cart, the initialization program restores all hardware to service, clears all call memory, and removes all temporary overwrites from the SPC-TSPS package. At the end of initialization the system is left in the idle mode (under utility control) awaiting input from the card reader. In the batch-mode, the system is initialized between each job.

Before the generic recovery programs were debugged, the utility initialization program was also used to initialize many generic tables and software registers used by the generic programs. This feature was provided so that programmers would not have to include an excessive number of standard initialization commands in their debugging decks. As the generic program package became more reliable, these special

initialization routines were deleted from the utilities in deference to the generic initialization or restart routines.[4]

2.2.1.2 *Command Processing.* The command processing routines interpret the commands in the debugging deck and perform the specified functions at match time. The major functions provided are: (*i*) memory dump, (*ii*) transfer trace, (*iii*) initialization, (*iv*) jumps, and (*v*) conditional statements.

The dump function provides octal, binary, or decimal printouts of specific memory areas when a match occurs. The addresses to be dumped can be stated explicitly in the debugging deck, or the deck can identify the location that contains the start address for the dump, a feature known as indirect dumping. The trace function enables the programmer to obtain a printout of the "from" and "to" addresses and index register contents on each transfer instruction. Initialization functions enable the programmer to initialize index registers and memory contents to specific values before entering the test strip or to change register and memory contents during the test. The jump function is used primarily to enter the test strip initially and to re-enter it (with different input data) when the end of the strip is reached. Conditional functions provide selective control over the debugging run by allowing the programmer to specify the conditions under which other functions should be executed. For example, the programmer can specify that when matcher 6 fires the tenth time, the X register should be initialized to a value of 3 if the Y register is not equal to zero.

2.2.1.3 *Administration.* Part of the utility package consists of administrative routines which are used only by the Laboratory Support Group. These routines provide for making tape copies of memory and matching or reloading these tapes. Also, the administration routines provide printouts of the size and location of patch and spare memory areas. The Laboratory Support Group required these facilities in order to maintain rigid control over the contents of memory in the laboratory model.

2.2.1.4 *Overwrite Control.* The overwrite control routines allow program changes to be inserted in a symbolic format. Thus the programmer can concentrate on solving his problem rather than on manually translating his symbolic instructions into machine code. The overwrite program also automatically allocates all spare memory required when additional instructions are added to a program. Changes can be inserted either temporarily or permanently. Temporary overwrites are

automatically removed by the initialization program, and are always used on batch debugging jobs. Permanent overwrites are inserted only by the Laboratory Support Group.

2.3 *Functional Testing*

The normal method of debugging is to first check several basic instruction strips and then to combine these into a larger strip and check it, and so on. Eventually the strips become large enough that they constitute a set of "functional packages," such as the call connections program. At this point utility debugging diminishes and the functional test period begins.

A "functional" test differs from a "strip" test in that the initial and terminal conditions for the functional test cannot be handled by the utility package or the T-cart. For example, a functional test of the call connections program involves the actual seizure of a trunk and observance that the relays and network operated under control of the program. This need to have "actual," or realistic, conditions at the time of the test meant that the programmer had to be present for the test and that the test would take much longer than the few minutes normally allowed for a utility debugging run. Extensive use was still made of the utility system, however, in conducting functional tests.

Figure 3 summarizes the program debugging progress in orders debugged, using the three stages of debugging technology described. During the functional test period, the number of orders debugged per unit time was less because more sophisticated and time-consuming tests were needed to isolate the remaining problems.

III. SYSTEM TESTING

Many individual tests are performed on the various units and programs that constitute a complete system such as TSPS No. 1, but system testing properly refers only to those tests that treat the system as an entity and where the stimuli and measurements are made at the system ports. The principle results from this type of testing are the measurements made to ensure that an acceptable system performance objective is met.

Planning for this testing sequence began during late 1966, as shown in Fig. 1, and continued throughout the testing interval until the machine was turned over to the operating company in late 1968. During the early stages of the test planning for TSPS No. 1 it was decided to shakedown first only those features required to allow the system
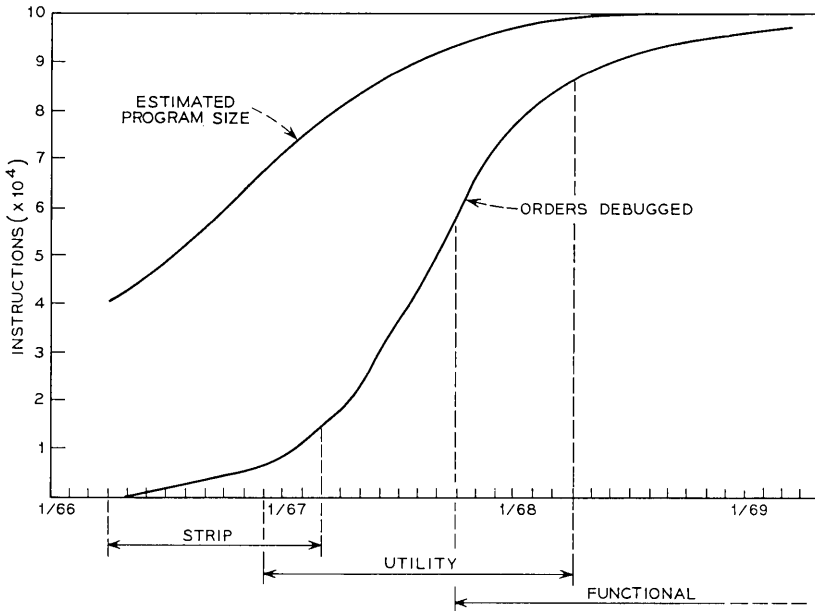
Fig. 3—Debugging progress curves.

to handle a simple call load as generated by load boxes. All subsequent testing could then take place with the system continuing to process traffic, and any interference with the systems ability to process calls would immediately point out design conflicts so that corrective action could be taken.

The call load background environment was the closest possible practical approach to the machine's normal real world environment. It was not intended that the testing done during this interval be done in the presence of an overload or even a full load condition, but rather that the system be continually exercised at a reasonable level of call load activity. As the testing progressed, the type of calls generated for the background were expanded to include as many variations as the available test equipment would allow.

In order to implement this test plan, two hardware test sets were developed:

The Automatic Local Toll Simulator Set (Fig. 4) was developed to interconnect with the TSPS incoming trunks at the main cross-connection frame terminations. This "load box" provides for the
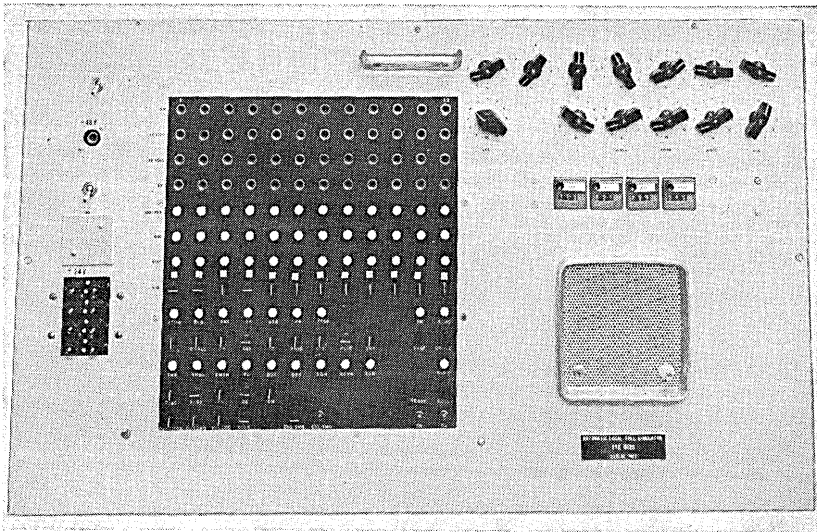
Fig. 4—Automatic local-toll simulator set.

origination and termination of calls[5] through the system. It generates any type of call that the system can process and verifies each system response to the external stimuli offered. Sufficient sets were provided for Morristown to allow the generation of approximately 4000 calls per hour of all types.

The Automatic Operator Simulator Set (Fig. 5) was developed to interconnect with the 100B Traffic Service Position. It electrically detects the various lamp displays and generates an appropriate operator's keying response, allowing the system to complete processing of the call. This set processes any type of call that can be connected to a position.

As shown in Fig. 6, these sets were connected external to the TSPS No. 1 system—i.e., they did not become a part of the system or change its operating characteristics in any way.

The test sets made possible the continual call loading of the system over the relatively long period of time during which the other operational, maintenance, and administrative tests were performed. They also provided data for one of the key system performance measurements; the Call Failure Rate.

During the test planning phase the call processing and administrative tests were written in detailed step-by-step form, and the main-
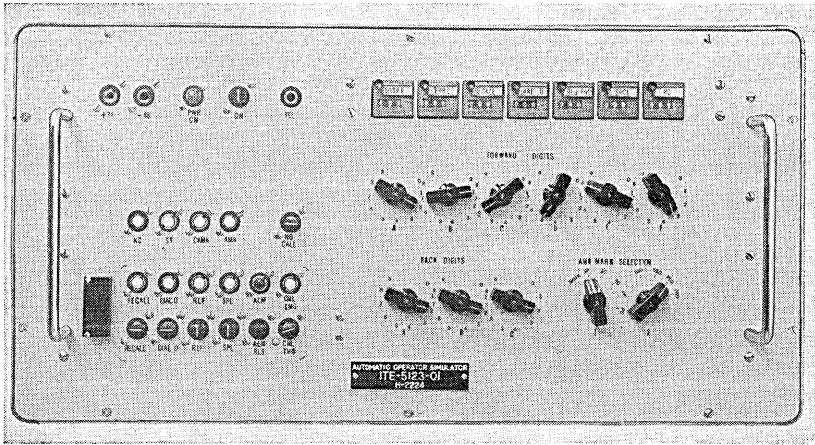
Fig. 5—Automatic operator simulator set.

tenance tests were adapted from the individual programmer's functional tests. In total, there were 345 individual tests or test items that formed the system test specification for the Morristown office and a selected sub-set has been produced in the form of Western Electric Company Installation Handbook Sections for the system testing of later offices. Since the installation at Morristown was used to prove-in the system design, the overall system testing performed was consid-
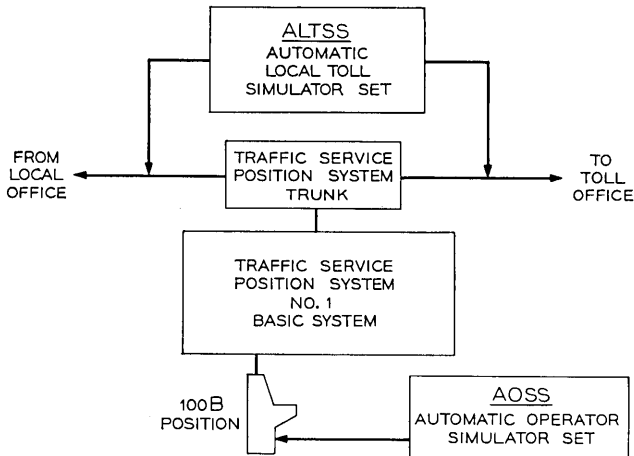

Fig. 6—Connection of ALTSS and AOSS to system.

erably more detailed and intensive than is required for later installations.

As the testing progressed, the machine was brought under call background load in late March of 1968, and this background was increased to include all call types by April. It should not be assumed that the machine was under continual call loading from this time until cutover. The call background was provided to enable the detection of program and hardware interaction problems, and as these were detected, various hardware and software techniques (utilities, etc.) were used to isolate and assist in the required solutions. Continual hardware and software changes were implemented during this interval, and entirely new programs were added to the system as they became available. When large changes were made, application of a sub-set of the detailed call processing and administrative tests, observation of the call failure rate and maintenance TTY output were instrumental in determining the goodness of the system following the change.

Throughout most of this interval the primary tool for locating and clearing software problems was the utility program package, described in Section 2.2.1. It quickly became apparent that the use of the utilities was incompatible with the call background due to the consumption of real time by the utility program. The output of the utility program is normally via the high-speed printer, and this work is done on an A-level interrupt basis. While in A-level and printing the utility results, so much time is used that the system is blinded to
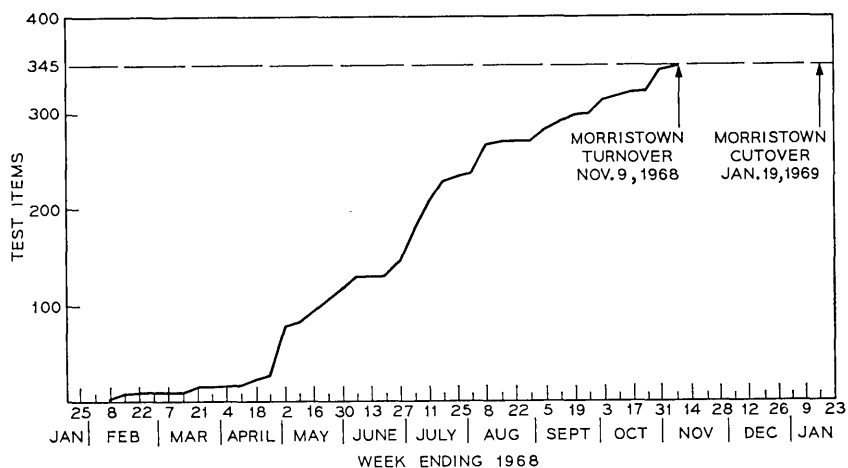

Fig. 7—System test items completed.

external stimuli and the call processing function is affected. Early in the testing interval, this interference could be tolerated, but later in many cases the call background was required to develop the problem, and because of this conflict, the data dump store and hardware transfer trace were provided in place of the utility package (see Section IV).

All during the testing interval, progress was marked by various indicators, including the Call Failure Rate, Test Items Completed (see Fig. 7), hours of simplex or duplex operation, hours of call loading, various maintenance printouts from the teletypewriter in the Maintenance Center, and the number of outstanding trouble reports. As turnover approached, the important measurements were the call failure rate, and the numbers of maintenance interrupts,[2] audit messages,[4] peripheral unit failures,[2] position subsystem failures,[2] and outstanding trouble reports. As these items approached a reasonable level, turnover to the Telephone Company took place and a new stage of testing was begun.

Following turnover, the New Jersey Bell Telephone Company performed tests on the machine, including the handling of live (non-simulated) test call traffic through the system while performing many maintenance and administrative routine operations. As was to be expected, this testing turned up problems that had not or could not have been detected previously. An intensive effort was required to clear these problems before cutover.

In spite of a few known residual design problems, the Morristown TSPS No. 1 system was cut over into service on January 19, 1969, and

TABLE I—TSPS No. 1—PERFORMANCE CRITERIA MEASURED OVER A 48-HOUR PERIOD

	Requirements for Morristown	Requirements after 5/1/70
Call Non-Completion Rate*	$\leq 0.5\%$	0.3%
Audit Errors	$\leq 250/\text{day}$	50
Maintenance Interrupts		
Software	$\leq 3/\text{day}$	1
Hardware	$\leq 10/\text{day}$	4
System Recovery Phases		
MNA, MJA	$\leq 2/\text{day}$	0
SIA, SIB	≤ 0	0
Store Errors	$\leq 2/\text{day}$	2
SPC Faults	$\leq 10/\text{day}$	5
Peripheral Unit Faults	$\leq 20/\text{day}$	10
Position Subsystem Faults	$\leq 10/\text{day}/\text{group}$	10

* Includes an allowance for test equipment irregularities.

has provided a satisfactory grade of service ever since. As expected, customer usage of the system revealed design problems that had not been detected by earlier testing. A concerted effort had to be applied in the first few weeks after cutover to clear these problems.

Based upon Morristown system experience, the performance criteria were established for all new systems. These are summarized in Table I. Basically the performance criteria establish an upper limit for the number of unexplainable trouble or fault indicating events. If a particular threshold is exceeded but the excess can be attributed to a source external to the TSPS No. 1 system or can be attributed to an explainable procedural error, the system performance is considered acceptable.

IV. FIELD SUPPORT AND HARDWARE MONITORING FACILITIES

As one might expect, no reasonable amount of design shakedown or system testing can guarantee a trouble-free design for a system as complex as TSPS No. 1. The combinations of key actions which the TSPS operator can generate individually and in concert with the other operators served by the system are practically unlimited. As a result, a field support operation that continues until design problems no longer interfere with field operation must be provided. To effectively support the field, tools for obtaining the right kind of data to allow design trouble analysis must be provided, and the tools must not interfere with basic system operation.

4.1 Dump Store and Transfer Trace

The two most vital sets of data required in a stored program control system to isolate design problems are the state of the system's memory at the time of the interesting event—the time the problem or trouble manifests itself—and the program sequence leading to the interesting event. To satisfy this need, the data dump store* and hardware transfer trace were developed. The dump store is an autonomous facility that continuously monitors the state of the call or transient

---

* The data dump store concept originated with the No. 1 ESS development. For TSPS No. 1 a specially modified PBT store or store pair is connected to one of the store buses to monitor all processor writes to the unprotected memory in the system stores. Since the unprotected memory area is assigned to various 16ths of the systems stores,[6] each 16th of the dump store memory is equipped with special switches that permit specifying the system store name code and 16th that is to be monitored by each 16th of the dump store memory. Also, in a write-only mode the dump store can be made to accept writes at a rate faster than a system store in a normal read/write mode. As a result, the dump store can accept all store writes at the system write rate.

data stored in memory. The transfer trace continuously records in its autonomous memory the FROM and TO addresses of the last 80 transfer instructions executed by the program system. At the time of an interesting event, both the dump store and transfer trace are "frozen" with their present contents. Later the stored information is extracted for offline analysis and both devices are reset to record data for the next interesting event. An offline program was developed to organize and format the data to ease the job of interpretation and manual analysis.

The interesting event is detected by setting program address matchers to fire, thereby freezing the dump store and transfer trace. As a result the state of the system with the trouble condition present is preserved in the monitoring devices without in any way interfering with system operation or recovery.

Information gathered from the dump store and transfer trace has been instrumental in solving most post cutover design problems in the TSPS No. 1 system. In the first 20 weeks following the Morristown cutover about 100 dump store tapes and transfer traces were processed. In most cases they contained sufficient information to permit complete problem solution or they narrowed the scope of the problem considerably. In the latter case, the program address matchers were reset to fire at program locations which could provide more data in the narrowed problem area. This is the significant advantage of using program address matchers as interesting event detectors.

Because the original 80 transfer capacity transfer trace was so effective, an improved version has been developed which provides for storing the last 600 FROM/TO transfer addresses or the last 100 transfer addresses plus the contents of the processor index registers at the time of each transfer.

V. EPILOGUE

System testing requires a thorough understanding of all the functional and maintenance requirements of the system and requires early and careful planning. The people who are going to be the system testers should be identified early to allow them to broaden their knowledge to be effective as system testers. Test plans should be made early enough to insure that test facilities like the load boxes and dump stores are available when needed. System testers should have a reasonable understanding of existing systems of time-proven design that are going to interconnect with the newly designed system. There is

always the tendency to place fault on the new system when interconnection problems develop. Experience has shown that the interconnecting system is often at fault and the troubles are revealed because the new system has more sophisticated maintenance features or is less tolerant of equipment that only marginally complies with specified operating criteria. Everyone concerned with system testing must maintain an unbiased position.

Finally it should be recognized that exhaustive system testing for design shakedown does not end with the first installation. Whenever a new system feature is added or system capability is exploited for the first time in the field, design shakedown testing must be performed. For example, the second TSPS No. 1 site at Miami was the first to employ multiple remote position subsystems and the third site at Houston was the first that is large enough to permit testing to determine if the system has sufficient real-time capacity to handle design limit traffic loads. Also shakedown testing must be performed to debug growth procedures for adding equipment to a working office whenever equipment of a particular type is added for the first time. Design shakedown testing ends only after all system features and capabilities have been fully exploited in the field and the system is capable of smooth field operation requiring a minimum of manual intervention and maintenance. This is achieved only through continued diligent field support.

REFERENCES

1. Haugh, G., Tsiang, S. H., and Zimmerman, L., "System Testing of the No. 1 Electronic Switching System," B.S.T.J., *43*, No. 5 (September 1964), pp. 2575–2592.
2. Durney, G. R., Kettler, H. W., Prell, E. M., Riddell, G., and Rohn, W. B., "TSPS No. 1: Stored Program Control No. 1A," B.S.T.J., this issue, pp. 2445–2507.
3. Barney, D. R., Giloth, P. K., and Kienzle, H. G., "No. 1 ESS ADF: System Testing and Early Field Operation Experience," B.S.T.J., this issue, pp. 2975–3004.
4. Kettley, A. W., Pasternak, E. J., and Sikorsky, M. F., "TSPS No. 1: Operational Programs," B.S.T.J., this issue, pp. 2625–2683.
5. Jaeger, R. J., Jr., and Joel, A. E., Jr., "TSPS No. 1: System Organization and Objectives," B.S.T.J., this issue, pp. 2417–2443.
6. Baker, W. A., Culp, G. A., Kinder, G. W., and Myers, F. H., "TSPS No. 1: Stored Program Control No. 1A Store," B.S.T.J., this issue, pp. 2509–2560.

Index to TSPS No. 1 Articles

No. 1 ESS ADF:

System Organization and Objectives

By J. C. EWIN and P. K. GILOTH

This article describes the system organization and objectives of the No. 1 Electronic Switching System Arranged with Data Features. The new hardware and software designs required to adapt a No. 1 ESS to a store and forward system are outlined with emphasis on the techniques used to get large system capacity, continuous reliable operation, and flexible service features. System operation, traffic capacity, and use of the new system are also discussed in this introduction to seven detailed articles.

I. INTRODUCTION

A new store and forward message switching system has been designed and is now serving a nationwide network for the Long Lines Department of the American Telephone and Telegraph Co. It handles administrative traffic, time and payroll reports, circuit order layout records, and plant service results records. The new system is called ADNet, for Administrative and Data Network.

This is the first of eight articles reporting on the new system, which uses single and multistation lines for transmitting teletypewriter and data messages. The lines are supervised by an electronic message switcher which polls the stations, receives their messages and queues them for delivery to one or more destinations. The switcher is designed around the No. 1 ESS processor[1] supplemented by new peripheral units for assembling, storing, and transferring data characters. It is called No. 1 ESS — Arranged with Data Features: No. 1 ESS ADF. Its capacity and reliability exceed that of other known electronic message switchers.

In today's business world there is a growing need for rapid and economical delivery of data and printed copy, for simultaneous transmittal to several destinations, and assurance of delivery. Store and forward techniques allow messages for a given destination to be queued

and lines to be engineered on the basis of traffic delay. For customers who can tolerate the delay in delivery of information, this method allows more efficient use of lines than line switching arrangements. Thus, economies are realized by use of narrow-band facilities, by serving several stations on a single line, and by queueing messages for contiguous delivery.

Although electromechanical systems of this type[2] have been provided by the Bell System since 1940, the advent of electronic processing and electronic station controllers has opened new opportunities for accommodating service features, speeding delivery and expanding traffic capacity. The system described here is currently used to provide internal Bell System communications.

II. SYSTEM OBJECTIVES

The prime objective of the new system is to provide a more efficient and economical means for handling teletypewriter and other data traffic on a private line basis by using the large switching capacity of No. 1 ESS.

2.1 Service Features

The new system has been designed to provide:

(i) Compatibility between computer and teletypewriter terminals.

(ii) Speed change: 60, 75, 100, and 150 words per minute.

(iii) Code change: Baudot and American Standard Code for Information Interchange (ASCII).

(iv) Parity error detection.

(v) Mnemonic addresses and group codes for routing.

(vi) Multiple address message delivery.

(vii) Time and date insertion in message heading.

(viii) Sequential message numbering.

(ix) Message retrieval by identification of message number, time, and date.

(x) Privacy.

The Appendix lists and describes these and other service features.

2.2 Maintenance and Administration

The system has been designed to give 24-hour service with reliability comparable to that of telephone switching systems. Individual station and loop problems are quickly revealed by regular polling and

troubleshooting. Discrete call-in codes alert only specific stations on a line and, with proper answerback, guarantee delivery to the proper station. Mutilated messages are recognized by character parity and the message retrieval function will often permit recovery of lost or mutilated messages.

By following No. 1 ESS maintenance philosophy, maintenance effort and the hazard of an outage are greatly reduced. This involves dual processors with automatic identification and location of faults.

Administrative effort has also been minimized. Traffic reports are available by line and by station to head off temporary congestion and allow long term traffic balance. Upon instructions from a control location, traffic destined to specific stations can be diverted. Other instructions permit suspension of message pickup from specified stations. Simple means are provided for adding group codes, changing addresses, or adding new stations.

2.3 *Flexibility*

The system can grow in a modular fashion, allowing economy for either small or large offices. The basic repertoire of service features allows a wide range of choice for individual stations or communities of users. The addition of new service features may require hardware as well as program changes. In either case such changes are possible without disrupting office operation.

2.4 *Capacity*

The new message switching system is designed to handle at least 1,000 lines. The system can efficiently accommodate a wide variety of message lengths up to many pages. It will handle messages occurring concurrently on many lines, initiating delivery to an idle line within 10 seconds after receipt of a full message. Polling codes can deal with 20 stations on a line. Multiaddress codes allow a single message to be directed to 300 or more stations.

III. NETWORK CONFIGURATION

The network which No. 1 ESS ADF controls is indicated in Fig. 1, which shows the relationship among stations, controllers, lines, and No. 1 ESS ADF. Each station has an associated electronic controller.[3] The stations may be computer ports or teletypewriters of the send only, receive only, or automatic send and receive type. The controller responds to polling signals from the No. 1 ESS ADF and registers
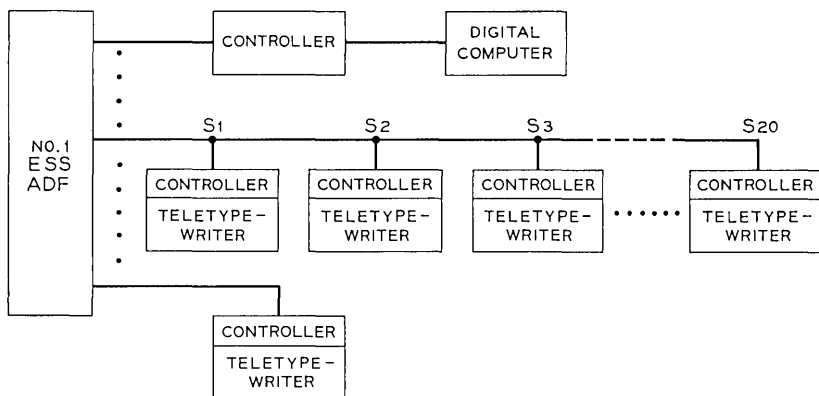
Fig. 1—Connection of terminals to No. 1 ESS ADF.

service requests when the customer has a message for delivery. Figure 1 shows several stations on one line. Contention for service is avoided by No. 1 ESS ADF which determines which station may receive a message or deliver a message at any time.

On half duplex lines, messages may be originated and terminated, but not simultaneously. Full duplex lines operate on a more complicated algorithm which allows supervisory interruptions without interference with message delivery. Full duplex lines may handle originating and terminating messages simultaneously. Mixing full and half duplex stations on one line is not permitted. Neither may different speeds or codes be mixed.

The traffic load on each line is the sum of the loads for the individual stations. The load is totalled separately for each direction of transmission. Traffic tables have been prepared which show the load which can be carried for given queue delays for half and full duplex lines. The physical routing of each line is carefully configured to interconnect the greatest number of stations not exceeding the prescribed load with minimal cost facilities linking them.

The stations are connected by private-line facilities operating up to 150 words per minute. The facilities are derived in a conventional manner using a full cable pair per station where distances are short and narrowband frequency division carrier channels where distances are long. These facilities are maintained through existing private-line-serving test centers which are capable of sectionalizing facilities and making measurements on them.

No. 1 ESS ADF has been programmed to quickly detect a difficulty in a controller or on a line by failure to receive proper acknowledgment from messages or polling. Reports are automatically printed on a teletypewriter at the serving test center closest to the No. 1 ESS ADF. This enables many problems to be identified and corrected before personnel at the affected stations are aware that they exist.

Stations transmit binary signals by means of frequency shift keying at 1175 ± 100 Hz. They receive at 2125 ± 100 Hz. An ac hub arrangement is used to join facilities at branch points in a line.

In addition to reports on station and line failures, the No. 1 ESS ADF provides reports periodically, or on request, on queue lengths and stations out of service to a number of specially designated teletypewriters at network management centers.

IV. SWITCHING CENTER DESCRIPTION

4.1 General Design Plan

The design of No. 1 ESS ADF is based on No. 1 ESS technology to minimize design effort and to take advantage of the lower costs of apparatus in large production.[1] (Sixty per cent of the frames in No. 1 ESS ADF are No. 1 ESS production items.) As Fig. 2 shows, the program-controlled central processor from No. 1 ESS has been used without change; additional special units have been designed to gain access to data lines and to provide the mass storage needed for store and forward operation. The hardware and software design principles developed for No. 1 ESS have been used and extended to store and forward data. Bell System switching reliability has been incorporated in all new designs.

The bus structure shown in Fig. 2 is designed to make modular growth possible by adding line terminal units and memory units as required for increased message handling capability.

Early in the development of No. 1 ESS ADF, simulation and analytical studies were used to aid in the development of hardware and software designs to provide maximum traffic handling capacity. These studies led to a system organization which provides traffic handling capacity for at least 1,000 low-speed lines by means of overlapping operation within the 5.5 microsecond central processor and concurrent processing within a new wired-logic arrangement called buffer control. Repetitive operations such as serial-to-parallel and parallel-to-serial conversion of data characters, special character recognition, assembly
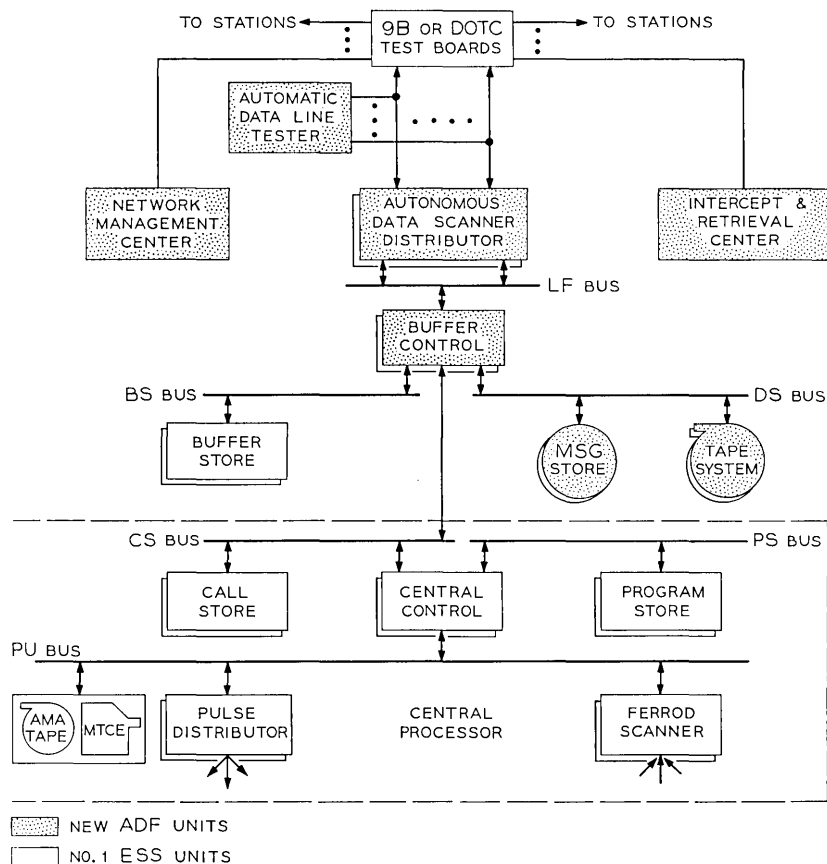
Fig. 2—No. 1 ESS ADF switching center block diagram.

of characters into computer words, error detection, and so on, are accomplished by wired logic in the buffer control and in other peripheral units.

The less repetitive and more complex functions such as station polling, heading analysis, address translation, routing, and traffic statistics, are handled by the stored program processor. This system organization minimizes the load on the central processor and bus structure so that very high traffic handling capacity can be obtained with economical computing units operating at a 5.5 microsecond cycle time.

The switching system hardware, as shown in Fig. 2, can be divided

into five main subsystems: autonomous data scanner distributor, buffer control, message store, tape system, and the central processor.

4.2 Autonomous Data Scanner Distributor

As shown in Fig. 2, the low-speed lines (up to 150 words per minute) are directly connected to an autonomous data scanner distributor.[4] The unit can terminate 512 half or full duplex lines operating with the Baudot code at 60, 75, and 100 words per minute and the American Standard Code for Information Interchange (ASCII) at 100 and 150 words per minute. This unit is a time-division multiplex system with 1,024 time slots—an input and an output slot for each of 512 lines. The common control of the data scanner distributor samples each line 1,650 times a second and stores the sample taken at the center of each data bit. This sampling frequency permits sending characters with a maximum of 1 percent distortion and accepting characters with up to 45 percent telegraphic distortion at 150 words per minute.

The data scanner distributor uses the memory capability of aluminum strip ultrasonic delay lines to address lines, to sample the data being received in serial form from those lines, to store each bit in a memory time slot until the entire data character is received, and to pass on the character bits in parallel to the buffer control. The data scanner distributor also uses delay lines to store control information that assists in servicing user's lines. This unit works as an independent subsystem and can handle the input and output from 512 lines in real time. A maximum of five data scanner distributors can be connected to the system.

The duplicated common control of the data scanner distributor operates in a matched mode. Matching and internal parity checks are used to protect data during the assembly and disassembly process. The processing system monitors the data scanner distributor; and when fault sensing circuits indicate trouble, the system calls in automatic fault recognition and diagnostic programs. For example, if one common control fails, wired logic fault sensing circuits switch out the faulty unit at microsecond speed. Maintenance programs then diagnose the failure when spare processing time is available.

4.3 Buffer Control

The purpose of the buffer control is to relieve the No. 1 ESS central processor of a large number of repetitive tasks and to provide buffering and timing compatibility between the central processor normal cycle

time and various rates of other peripheral units. The buffer control performs the following major functions.

(i) Receives data characters from the data scanner distributor and assembles characters into computer words.

(ii) Recognizes and flags special control characters.

(iii) Performs error control operations on the data.

(iv) Acts as a buffer in controlling the transfer of messages to and from the message store and the tape system.

(v) Interleaves the operation of the data scanner distributor, message store, tape, and central processor subsystems to permit concurrent operation of these asynchronous units.

The buffer control is a wired-logic, fully duplicated unit operating in a matched mode. All data transfers are matched and checked for parity. Buffer control administers the transfer of data between the buffer store and peripheral units (message store, tape, data scanner distributor), as well as the central processor, by providing queued access to the buffer store. Access to the buffer store is sufficiently often to meet the requirements of each peripheral unit.

The buffer control also operates as a maintenance coordination center and monitors all bus transmissions to and from connecting units. The buffer control, in conjunction with maintenance programs, uses match and parity circuits as the primary means for detecting and diagnosing troubles.

4.4 Message Store

The message store is a duplicated, sequential access (block oriented) memory unit which provides in-transit storage for all messages passing through the system.[5] Storage is accomplished in a disk file consisting of four double-faced rotating disks with a capacity of approximately 60 million bits. Duplicated disk files are synchronized by a digital servo system which controls the frequency of the motor drives so that identical data can be transferred to or from both files in one bus cycle time. The message store performs the following main functions:

(i) Retains each originated message until satisfactory delivery has been made to all addresses.

(ii) Stores a cross-reference file consisting of tape search numbers as a function of message numbers for retrieval of messages from the tape system.

(iii) Stores various types of registers, queues, and data blocks associated with the operational program.

Data are stored in fixed record length blocks on the disk with each block containing thirty-two 24-bit words. Data are stored in a block interlaced format at approximately 1,000 bits per inch. The disks are each divided into 16 sectors; queues are set up in buffer control call store so that in each revolution sixteen blocks of data can be written on or read from the disk system. Since the rotation interval of the disk system is 40 milliseconds, the average access time for a block of data can approach 2.5 milliseconds when a large number of blocks are being handled.

The switching system is designed to handle two duplicated disk systems providing a total of 120,000,000 bits of storage. The maximum traffic handling capacity of the disk system is approximately 38,400 characters per second.

4.5 *Tape System*

The tape system consists of two tape unit controls and up to 16 tape units per switching center.[6] Each tape unit control translates the tape instructions received from the buffer control into detailed logic sequences necessary to execute the tape operation. The tape unit control also assembles, disassembles, and buffers data transferred to or from the tape unit. Tape unit controls operate concurrently and can work with any tape unit for journal file, permanent file, and message retrieval functions.

The tape transports are nine-track units recording 800 bits per inch at a speed of 56.8 inches per second. The tape transport can read both forward and reverse, and data integrity is obtained by the use of parity and read-after-write check circuitry. The operation of the tape units has been designed to be almost entirely controlled by computer. Traffic attendants are required only to replace tape reels when indicated by the computer.

4.6 *Central Processor*

The central control, call stores, program stores, control circuits, and maintenance center are all standard No. 1 ESS units.[1] Both the call store and the program store are modular and can grow very large, as required.

The call store (ferrite sheet) units can be added in increments of 4,096 words. A maximum ADF system can have 192,472 words of call store. The program store is the semi-permanent twistor memory with removable aluminum cards containing magnetized bit patterns. Program stores can be added in increments of 65,536 forty-four-bit words

for the storage of the call processing programs, maintenance programs, and for translation tables for customer lines, stations, features, group codes, and directory numbers. A maximum ADF system can have 393,216 words of program store.

4.7 *Summary of System Design*

The system organization and detailed design of subsystems provide:

(*i*) High processing power by concurrent operation of four asynchronous subsystems and use of wired logic for repetitive functions.

(*ii*) Modular growth with the flexibility to add high-speed lines and trunks as required for multiple switching center applications.

(*iii*) Economies associated with standardization of circuit design based on present No. 1 ESS production.

V. STATION AND TRANSMISSION DESIGN

A new family of electronic controllers and associated data sets have been developed for Bell System 4-row ASCII teletypewriters stations.[3] These electronic controllers are designed to provide improved message control, lower maintenance, and silent teletypewriter operation when not in use. The following features are built in to improve the integrity of message transmission:

(*i*) Roll call after transmission of each message.

(*ii*) Teletypewriter motor control.

(*iii*) Error detection.

(*iv*) Station power failure detection.

(*v*) Out-of-paper and paper jam alarm.

(*vi*) Regeneration of incoming and outgoing signals.

The performance of data transmission lines has also been improved by the use of new data sets. Maintenance has been improved by distortion monitoring provided by test boards and by the automatic distortion measurement capability of the No. 1 ESS switching center.

VI. OPERATIONAL AND MAINTENANCE PROGRAMS

6.1 *General*

The instruction format used in No. 1 ESS ADF is the same as that used in No. 1 ESS.[1] The program organization is similar to No. 1 ESS but has been influenced by the real-time requirements of handling many messages simultaneously by store and forward. Defensive techniques have been employed to guarantee the integrity of each message accepted by the system, since unlike conventional voice switching systems, the originator does not directly contact the terminator.

Therefore, the system is designed to guarantee the delivery of all messages accepted.

The program has been organized to respond quickly to errors detected by the internal hardware and software trouble detectors to assure dependable operation continuously. The program has been designed so that the system can recover from hardware and software troubles without losing or mutilating messages.

6.2 *Service Features*

A comprehensive set of service features has been incorporated into No. 1 ESS ADF to meet new requirements. The complete set of features includes those commonly found in single-user private line computer switchers. Other features have arisen from the ability of the machine to handle a number of different users with varying requirements. In addition to those features mentioned in the introduction, the system provides flexible heading formats, four levels of precedence, directory number addressing, interception of undeliverable traffic, journal file of messages, traffic statistics and status printouts to improve system administration and maintenance. All service features are listed and described in the Appendix.

Emphasis has been placed on developing message formats that are easy to read and easy to use. The system is designed so that all control characters are deleted; clear English copy appears on all printers. A message consists of two functional parts, heading and text. The heading contains message number, time, date, precedence, address (or group code), and personal address information. The text is the actual information conveyed by the originator and is delivered unaltered by the No. 1 ESS ADF. Although the heading is flexible in content, the format must be strictly adhered to. Format violations are detected and a service message is sent advising the originator of the type of error made. A typical message format is:

ORIGINATOR'S COPY
s102 04/26 1340 EST   (ADF inserts message number, date,
CHGO123 BOS12 [Mr. H. Jones]   time) (terminal addresses supplied by
"TEXT"   originator)

DELIVERED COPY
CHGO123   (Station address)
R506 04/26 1345 EST   (ADF inserts terminal message num-
CHGO123 BOS12 [Mr. H. Jones]   ber, date, time)
"TEXT"

Three features—privacy, action requests, and service messages—deserve specific emphasis because of their importance to overall system operation.

6.2.1 *Privacy*

Special steps have been taken to prevent the unauthorized delivery of messages. System programs have been designed so that messages cannot be delivered between stations of different users unless specified in the No. 1 ESS ADF translation tables for the originating and receiving stations.

6.2.2 *Action Requests*

Stations may originate various action requests which notify the switching center of certain desired changes. These changes may involve putting a station out of service (skip) or obtaining certain specific information, such as retrieving a message. Some 40 types of requests may be originated by the user and the maintenance center.

6.2.3 *Service Messages*

In response either to an action request or to some reportable occurrence, No. 1 ESS ADF originates certain messages to user or maintenance stations. There are 65 different messages which cover a variety of requests and conditions.

The entire system program has been designed so that user features are flexible and can be selected for each user or each station. Lines, stations, and features can be added or deleted from a user's set by a "recent change" procedure which can be done rapidly on-line without recompiling programs.

6.3 *Operational Programs*

The operational programs have been designed to handle a large variety of teletypewriter station terminals using the Baudot and ASCII codes at a variety of data rates.

Features mentioned earlier are implemented in 75 programs consisting of 100,000 computer words of code. The details of the operational programs can be found in Ref. 7.

The program system handles a typical message from a data terminal in the following manner. A message is prepared on a teletypewriter and then is placed on the machine with a "bid" for message pickup. All stations in the system are periodically polled by the switching center. When the service request is recognized by the processor, the transla-

tion tables are consulted to obtain a description of the data machine and the features selected by the user for that machine. Message number, time, and date are then sent to the originator, and transmission of the message into the system is started.

The message passes through the test board where line distortion and performance can be monitored. The data scanner distributor samples the incoming bit stream and converts the serial character stream to parallel characters. The parallel, multiplexed characters per line are received by the buffer control where they are assembled into 24-bit computer words and stored in the buffer store. Also in this process, the buffer control recognizes and flags special characters and detects and flags parity errors.

The computer length words associated with each active line are transferred to the processor and assembled into 32-word blocks in the call store. The processor extracts enough data from the heading of each message to put an entry into a queue for each terminator. Blocks containing heading and text are transferred via the buffer store to the message store.

As each message is delivered, the message is retrieved from the message store and is disassembled to the outgoing line by reversing the input process. After messages have been sent to all terminals, the data identifying each transaction, the heading and the text, are transferred from the message store to the tape system for permanent file and journal file recording.

6.4 *Maintenance Programs*

In a message switching system, operation without interruption is very important because there is a continuous flow of data through the entire system for the duration of each message. It is imperative that messages not be garbled, lost, or misdirected.

Reliability in the No. 1 ESS ADF system is accomplished by:

(*i*) Duplication of all units and buses.

(*ii*) Matching and parity checks on all data transfers.

(*iii*) Fault recognition programs which control micro-second reconfiguration of units to maintain a working system.

(*iv*) Comprehensive set of automatic diagnostic programs.

(*v*) Comprehensive safeguards against a variety of anticipated user errors.

(*vi*) Automatic overload control.

(*vii*) Automatic recovery program in case of memory multilation.

(*viii*) Comprehensive set of user and maintenance service messages.

The system is designed to operate without interruption for long periods of time. Experience with No. 1 ESS switching centers over the last three years has shown that a well debugged system can operate for years without a single interruption of service.[8] The programs for ADF are designed so that if a duplex failure does occur, all messages stored on disks can be recovered and delivered automatically after the system restarts. A record will also be found on disks of all messages being originated. After restart, originators of incomplete messages will automatically be sent a service message requesting that the message be originated again.

The maintenance programs for fault recognition, automatic diagnosis, and exercise are covered by 100 programs amounting to 150,000 words of code. The programs consist of modified No. 1 ESS programs for No. 1 ESS units and new programs for the new No. 1 ESS ADF units. New hardware and software techniques have been developed for handling autonomous wired logic and memory units such as the data scanner distributor, message store, and tape system. These programs are described in detail in Ref. 9.

Maintenance dictionaries and raw data printouts have also been developed for the new No. 1 ESS ADF equipment to permit rapid repair of hardware failures. This involved the insertion of 200,000 faults to develop the necessary data for dictionary preparation.

VII. CAPACITY

A store and forward system of this type can best be characterized by its thruput, that is, the maximum number of ten-bit characters transferred per unit of time at the interface between the data lines and the data scanner distributor. The thruput of a store and forward system is a function of message length, the complexity of the message heading, the multiple address factor, and the amount of intraline traffic.

The thruput of No. 1 ESS ADF was measured by using a programmed computer load test facility. The results are given in detail in Ref. 10. The results show that No. 1 ESS ADF can handle 6,800 characters per second at a message length of 1,200 characters. This is equivalent to 19,000 messages in the busy hour. For this configuration 600 simultaneous message transmissions (input plus output) can be handled. If the multistation teletypewriter lines are loaded to 0.6 erlang the system can handle 1,150 lines.

By using the load testing facility, the system was driven into real-time overload and it was demonstrated that the system could continue to operate without aborting, mutilating, or losing messages.

VIII. OPERATIONAL EXPERIENCE

The No. 1 ESS ADF system replaces an existing multicenter electromechanical system used for coordination of the Long Lines Department work operation. The revised nationwide network called Administrative-Data Network (ADNet) which went into service February 3, 1969, connects some 720 Long Lines, Telephone Operating Company, and Western Electric Company locations to the switching center with 400 circuits and 1,250 four-row ASCII teletypewriter machines. Connection is also made to computers at two Long Lines data processing centers so that computer-generated data can be sent or field data can be received and processed.

The system is being used to send administrative messages, traffic orders, commercial service orders, payroll, plant service results, circuit layout information and budget analysis reports. Daily originated plus terminated traffic is now averaging 38,000 messages consisting of 50,000,000 characters.

The system has given very satisfactory operation since February 3, 1969. System down time is averaging two minutes per month without loss of messages. A more detailed account of operational experience is given in Ref. 10.

IX. ACKNOWLEDGMENTS

Many people contributed to the development of this system from many areas of Bell Telephone Laboratories, American Telephone and Telegraph Company, and Western Electric Company. Significant contributions were made by Bell Laboratories people from the Data Systems Engineering Center and the Device, Telegraph Station, Electronic Switching, and Data Switching Development Laboratories. Overall guidance on the application aspects was given by the Engineering, Marketing, and Long Lines Departments of the American Telephone and Telegraph Company.

APPENDIX

Service Features

A.1 *Mnemonic Addresses*

Mnemonic address codes (a combination of letters and numbers such as an abbreviation or contraction of the destination name) are used to route messages to the proper destinations. The user may select a number of different codes, each of which may include up to seven

characters. In addition, a given destination may be assigned more than one code.

A.2 *Group Code Addresses*

Group codes are mnemonic codes which address a specific combination of stations. A group code is selected by the user and may consist of up to seven characters.

A.3 *Multiple Addressing*

The ADF system can handle originated messages with up to 379 destination addresses. The destination address may be one or more mnemonic address codes, group codes, or call-directing codes (for five-level half-duplex stations only).

A.4 *Precedence*

The ADF system queues messages for delivery to terminating stations and will rank messages for delivery according to the following descending levels of precedence:

Precedence Level	Code
URGENT	1
RUSH	2
NORMAL	3
DEFERRED	4

The precedence assigned by the message originator to each mnemonic code of the message heading affects only the delivery of the message to that address by the terminating ADF office.

A.5 *Personal Address Information*

For originators using the ASCII format, up to 31 characters of personal address information may be used with each mnemonic address code or group code in each message heading.

A.6 *Message Identification*

The user may elect to identify each message with a sequence of from one to seven characters within the message heading. The use of message identification is available only for five-level full duplex stations.

A.7 *Message Numbering*

A message numbering service is available at the originating machine and at the terminating machine. Originating message numbering is

optional for each station for both eight-level and five-level stations. Terminating message numbering is required for all stations.

A.8 *Date and Time Services*

Originating and terminating date and time service is optional. When used, the date and time are provided as a group following the message number. Time is indicated on a 24-hour clock with the user specifying the time zone.

A.9 *Delivery of Originator Nontext Information*

The delivery (or optional selective deletion) of message originator nontext information to terminating stations is governed by both originating and terminating options. Normally, the information units that may be delivered (as nontext information) will consist of the originating message heading, the relevant address (mnemonic or group code, precedence designation and personal address information), and the user message identification. Options vary for five- and eight-level stations and, also, for half-duplex and full-duplex per-line or per-station configurations.

A.10 *Tabbing*

An appropriate time interval after transmitting tabbing or form-feed characters allows the receiving terminal to perform the mechanical functions with a minimum delay. This option applies only to 8-level stations.

A.11 *Automatic Station Operation*

A message prepared for transmission is automatically picked up by the system without occupying the attendant at the sending station; message reception is automatic at unattended receiving stations.

A.12 *Multiline Hunting Groups*

The switching center can distribute message deliveries over a group of stations at a location with a rotary terminal hunting process. Messages addressed to the group are delivered to an idle station within the group. Messages may also be addressed and delivered to a specific station of the group.

A.13 *Station Service Arrangements*

The user may select half-duplex or full-duplex stations for originate only, terminate only, or automatic send and receive service. Three-row

Baudot and four-row ASCII teletypewriters can be specified. Arrangements are also provided for IBM 360 computers.

A.14 *Network Management Arrangements*

A user may select stations (within the user set) to perform control functions, to receive control and status information, and to receive messages undeliverable to their destination. A station may be assigned more than one of the above functions; only the supervisory position function requires transmission capability.

A.15 *Status Printouts and Traffic Statistics*

The No. 1 ESS ADF system compiles and sends status reports and traffic statistics to the status printout station. Status reports may be delivered periodically (as a summary), on request, or on occurrence. Traffic statistics are delivered daily and monthly.

A.16 *Treatment of Undeliverable Traffic*

A message becomes "undeliverable" when the message has been accepted by the system (origination was valid), but cannot be delivered to the intended destination. When a message becomes undeliverable, it will be rerouted to a suitable alternate station and a service message will be appended which indicates (if known) the addressee to whom the message could not be delivered.

A.17 *Skip in Polling Sequence*

A station is "on skip" when the ADF system is not picking up traffic according to the polling list. The No. 1 ESS ADF system will place any normally polled station on skip and will also take the station off skip and resume normal operation upon request from user or maintenance control position.

A.18 *Hold*

A station will be placed on or off hold at the request of the affected station user control station or maintenance station. When, temporarily, no attempt is being made to send messages to a given station and instead these messages are being stored, the station is on "hold." For high precedence message, an action copy will also be sent to a suitable station such as a control station.

A.19 *Alternate Delivery*

At the request of the affected station, all messages addressed to one station will be rerouted to a different specified user station. The station

precluded by alternate delivery will be returned to normal message reception at the request of the user.

A.20 *Permanent Message File*

All message traffic is stored on a magnetic tape file for specified periods of time to accommodate message retrieval.

A.21 *Message Retrieval*

Retrieval of messages (or groups of messages) sent or received by users within a specified time interval is entirely automatic. Messages not on active tapes can be retrieved by manually inserting an off-line tape on a tape drive. A service message accompanies each retrieved message. The service message identifies the copy as a retrieval and includes the message number used in the retrieval request and the pertinent station identity.

A.22 *Directory Numbers*

Every station has a ten-digit directory number unique within the system plan. The No. 1 ESS ADF system accepts messages addressed with directory numbers from maintenance stations and routes these messages to proper stations.

A.23 *Intercept of Undeliverable Traffic*

When a message is undeliverable to a given address, it will be delivered to a designated interception position which may be the originator, user control, or maintenance position.

A.24 *Charge Recording*

Service and feature usage (by the user) is automatically recorded for charging purposes.

REFERENCES

1. "No. 1 Electronic Switching System," B.S.T.J., *43*, No. 5, Pts. I and II (September 1964), pp. 1831–2609.
2. Bacon, W. M. and Locke, G. A., "A Full Automatic Private Line Switching System," AIEE Trans., *70*, Pt. 1, (1951), pp. 473–480.
3. Carney, A. C., Fitch, S. M., and Parker, G., "No. 1 ESS ADF: Teletypewriter Stations and Transmission Facilities," B.S.T.J., this issue, pp. 2941–2973.
4. Corbin, J. E., Lehman, H., and Townley, R. C., "No. 1 ESS ADF: Autonomous Data Scanner and Distributor," B.S.T.J., this issue, pp. 2857–2885.
5. Aitcheson, E. J., Ault, C. F., and Spencer, R. G., "No. 1 ESS ADF: Message Store—A Disk Memory System," B.S.T.J., this issue, pp. 2887–2913.
6. Potter, J. L., Strebendt, Mrs. F. E., and Williams, J. R., "No. 1 ESS ADF: Magnetic Tape Subsystem," B.S.T.J., this issue, pp. 2915–2940.

7. Kienzle, H. G., Nicodemus, K. L., Smith, M. T., Jr., Weber, E. W., and Zydney, H. M., "No. 1 ESS ADF: Message Processing Program Organization," B.S.T.J., this issue, pp. 2753–2830.
8. Haugk, G., "Early No. 7 ESS Field Experience I—Two-Wire Systems for Commercial Applications," IEEE Trans. Commun. Technology, *Com-15* No. 6 (December 1967), pp. 744–750.
9. Aitcheson, E. J., and Cook, R. F., "No. 1 ESS ADF: Maintenance Plan," B.S.T.J., this issue, pp. 2831–2856.
10. Barney, D. R., Giloth, P. K., and Kienzle, H. G., "No. 1 ESS ADF: System Testing and Early Field Operation Experience," B.S.T.J., this issue, pp. 2975–3004.

No. 1 ESS ADF:

Message Processing Program Organization

By H. G. KIENZLE, K. L. NICODEMUS, M. T. SMITH JR.,
E. W. WEBER and H. M. ZYDNEY

*The No. 1 Electronic Switching System, Arranged with Data Features
(No. 1 ESS ADF), uses the basic program organization of No. 1 ESS
with additional task programs, input-output equipment, and mass storage
to provide a reliable switching arrangement for store-and-forward data
service. This paper describes the operational program organization used
to control the reception of a message into the system from an originating
data station, its placement in the message store, and its delivery to the
terminating stations. In addition, it describes the service features and the
unique characteristics of the system's translation and message-integrity
features.*

I. INTRODUCTION

The No. 1 Electronic Switching System, Arranged with Data Features (No. 1 ESS ADF), is a stored-program system which uses the basic No. 1 ESS program organization with additional task programs, input-output equipment, and mass storage to provide a reliable switching arrangement for store-and-forward data service.

To meet the needs of data switching, No. 1 ESS ADF must meet the stringent operating requirements of a telephone switching office[1] in addition to meeting requirements that are unique to a data-switching office. The system must (*i*) respond rapidly in real time to the demands for service, (*ii*) have the versatility to provide the large variety of service features demanded of a modern data-communication system, (*iii*) be reliable, (*iv*) be capable of meeting the needs of different installations without modification of the basic program, and (*v*) be economically balanced in its use of real-time and storage facilities. In addition, the No. 1 ESS ADF must meet requirements beyond those imposed on telephone switching systems.[2] Unlike line-switched systems, a message-switched system provides no end-to-end verifica-

tion of the communication path or delivery of the message. The originating user is completely reliant on the system to properly deliver accepted messages to the appropriate terminators. This imposes stringent requirements on the message-switching system to provide message protection, assurance of delivery, and privacy.

In a message-switched system, communication between incompatible station arrangements is possible since the originating and terminating stations are never directly connected. Only the switching center need be compatible with all station arrangements. Thus, the service offering includes many station arrangements: (*i*) send only, receive only, and send-receive; (*ii*) single and multistation lines; (*iii*) multiline hunting groups; (*iv*) a variety of transmission rates; (*v*) a variety of information codes; and (*vi*) a variety of signaling sequences.

Other important features which are desired by users are multiple-address operation, the use of mnemonic codes and group codes for addressing, message numbering, time and date insertion, and message retrieval. A complete list of features is given in Ref. 2. The extensive set of service features offered by this system is made economically possible by employing the stored-program concept of system control.

A data message passes through three basic stages of operational processing: (*i*) detection of a request for service, (*ii*) processing of the input transmission along with storage of the message and the formation of a delivery queue for each addressee, and (*iii*) servicing of the delivery queues and transmission of each message to all terminators.

In the following sections of this paper a description of the operational program is given through both a general and a detailed account of the processing of a typical data message. While the system is capable of handling both five-level and eight-level stations, the scope of this paper will be restricted to eight-level operation since this represents the more modern station arrangements.

II. SYSTEM OPERATION

To utilize economically the transmission facilities, each line may service up to twenty eight-level stations as indicated in Fig. 1. The use of multistation lines requires that the switching center periodically interrogate or poll all stations to determine their desire to originate traffic. The polling is accomplished over the transmission facilities by an interchange of a sequence of characters within the code set of the data machines. Each station on a given line is assigned a specific station poll code to which it alone responds. The station responses
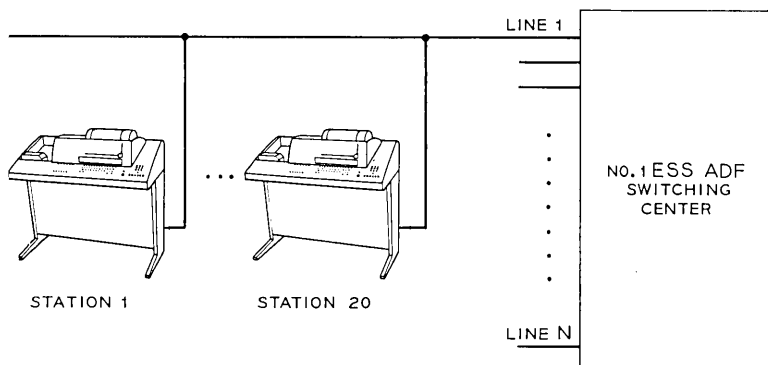
Fig. 1—The No. 1 ESS ADF system provides service to multistation lines.

which are tabulated in Table I contain information in addition to an indication of the desire to originate traffic. First, the importance of the originating traffic is indicated as priority or regular traffic. This allows for a per line priority option which involves the selective pickup of originating traffic on a per message basis. Priority options in prior switching systems have been limited to selectivity on a station basis. On any line which exercises this option, the switching center will pick up all priority traffic before accepting any regular traffic. Second, the status of the receive portion of the station is indicated. If the station receiver is out of service or made busy, the polling response indicates a not-ready-to-receive condition. This allows the switching center to hold traffic for the given station and concentrate on deliveries to other stations on the line.

On half-duplex (HDX) lines, the line can be polled continuously whenever it is neither transmitting nor receiving traffic. However, on full-duplex (FDX) lines, two independent communication channels are available for concurrent transmission and reception of traffic. Polling of FDX lines is conducted by temporarily halting any outgoing traffic on the line and sending the appropriate poll code for each station. Continuous polling of the FDX lines would interfere with message delivery. Thus, when all stations on the line have no traffic to send, polling is suspended and all stations are placed in a "suspend poll" state. If a user mounts a message tape on the station and bids for service, a polling request code is generated by the station and is recognized by the switching center which initiates a round of polling for the line. The suspend-poll mode is backed up by a

TABLE I—EIGHT-LEVEL STATION POLLING RESPONSES

Station Response	Input Traffic Available	Status of Receiver
C A N	None	Ready
N A K	None	Not Ready
P A C K	Priority	Ready
P N A K	Priority	Not Ready
R A C K	Regular	Ready
R N A K	Regular	Not Ready

time-out mechanism which insures the polling of all FDX lines at least once every 15 minutes.

A user who wishes to originate a data message prepares a paper tape in a standard format consisting of two parts, heading and text. These parts are delineated by special characters for start-of-heading, start-of-text, and end-of-text. Several messages may be cut on a single tape and the last message is followed by a end-of-transmission character. The heading is subdivided into fields for routing information, delivery priority, and personal address information and must follow specific format rules. If the format rules are violated, the originated message will not be accepted by the processor. A service message is sent to the originator indicating a control-code error or a heading format error with the erroneous heading field identified.

Having prepared the message, the user loads the message into his machine and conditions the machine to indicate that it has traffic. When the switching center detects the desire to originate traffic as a result of the polling response, it consults its translation information to obtain a description of the data machine, code and speed, and the service features subscribed to by the user for this machine. Based

on the service features chosen by the user, the switching center generates and transmits to the station a printed record which typically consists of the time and date of message pickup and a serial number assigned to the message which the user can use in referring to this message. The last part of the printed record is a sequence of characters which initiates transmission of the message to the switching center. The switching center assembles the characters into blocks of memory and builds a single copy of the message on magnetic disk. As soon as the complete heading is received and properly stored, the switching center proceeds to analyze it. In so doing, it generates a centralized descriptive record of the message called the message-processing block. This record contains the identity of the originator, message descriptive information, a list of the recipients, and an indication of where the message has been stored on magnetic disk. Upon completion of the analysis of the heading, an entry consisting of the location of the message-processing block is made in each of the terminators' message-delivery queues. Each line has a separate queue in which messages are ordered by priority and time of arrival within a priority class. Deliveries to the recipients may now be initiated simultaneously with, yet independent of, the input transmission. When the complete message has been received from the input line, the switching center continues to monitor the line for receipt of either a start-of-message character indicating a multimessage transmission or an end-of-transmission character which restores the station to the idle condition.

Delivery to the stations is initiated as a result of scanning the message-delivery queues. If the line associated with a given queue entry is idle and the station is ready to receive as indicated by the polling responses, a call-in sequence of characters is transmitted to the line and conditions the desired station for reception of the message. One of these characters, the call enquiry code, is unique to each station on the line. To insure that the proper station receives the message, the station responds with a unique station identification code which is verified with translation data. In addition, the station also responds with an indication of its readiness to receive. The switching center then generates and transmits a heading to the station based on the options chosen by the user. This typically consists of the mnemonic used in the heading to address the station (this helps the station attendant to identify the message recipient at a communication center which may be addressed by many mnemonics), the time and date of delivery, a message number assigned by the switching center

to the transmission, and the originating time, date and message number. The originator's heading, at the option of the user, is transmitted to the station, followed by the text of the message as derived from the copy stored on magnetic disk. The heading and text are code converted if the originating code differs from that of the terminating machine. Upon completion of the transmission, the switching center and the station engage in an exchange of signaling called roll-call to verify the proper delivery of the message. The switching center again transmits the call-enquiry code to which the station responds with its station identification code and the present status of its readiness to receive. If the station fails the roll-call checks, the message is requeued for future delivery. During the call-in and roll-call signaling sequences, input traffic on FDX lines must be halted so that the input channel can be used for responses from the station. The above discussion is directed to delivery to a single station; however, if the message is addressed to more than one station on a given line, the message is directed to all such stations in a single transmission. The call-in and roll-call sequences are expanded and the generated heading is directed to each station individually as appropriate.

On HDX lines intraline delivery capability of messages is provided. For this type of operation, the originating station halts its transmission upon completion of the heading. The switching center then analyzes the heading and restarts the station if no intraline deliveries are involved. If the message being received is addressed to stations on the same line as the originator, the addressed stations are called in, receive their generated heading and receive optionally the originator's heading. The switching center then restarts the originator's station, and the text is simultaneously transmitted to both the switching center and the receiving stations. The receiving stations are roll-called upon completion of the message.

The control logic provided by the switching center is further complicated by the need and desire of user groups to control and manage the traffic on their networks. The user may designate a particular station or stations as a control location. This control location has the capability of placing stations on skip, denying them origination capability, and on hold, denying delivery capability, or traffic may be rerouted to an alternate station. This requires that the polling process examine the skip status of all stations and send out polling codes only to those stations which are not on skip. It also requires the delivery process to examine the hold state of each station and skip over message queue entries for stations in a hold state.

The somewhat complex dialog required between the switching center and the stations it serves provides a high degree of integrity in handling messages. Any anomalies detected by the almost continuous monitoring of service are reported to transmission-plant craftsmen through service messages. This allows for rapid maintenance of the transmission plant which does not depend on user complaints to detect troubles as prior systems have done.

III. SOFTWARE STRUCTURE

3.1 *Task Schedule*

The organization of the No. 1 ESS ADF program was strongly influenced by the desire to maintain compatibility with the No. 1 ESS program to permit the possibility of future merger of these programs in a combined voice-and-data system. In designing the executive control program, the scheduling algorithm described by J. A. Harr, et al.,[1] has been used with only minor modifications.

The real-time tasks or stimuli processed by the switcher vary largely in their requirements for response time. The rapid response time necessary for the more critical tasks is achieved through an interrupt mechanism. The central control clock interrupts the base-level programs periodically and transfers control of the system momentarily to the input-output main program (J-level) which systematically searches for high-priority work. The majority of tasks, including all operational input-output tasks, can be handled with a 10-ms interrupt. Only a few infrequent maintenance tasks require a 5-ms scanning period during the interval that the maintenance task is active. The interrupt mechanism imposes a real-time overhead on the processor. First, the central control registers must be stored before performing the input-output tasks and then restored before reentry to the interrupted base-level program. Second, the process of scanning for work involves an administrative overhead, especially if done at a frequency greater than required. To minimize this overhead associated with the interrupt work, the standard 5-ms clock interrupts are subdivided by program means into odd and even interrupts. Only on odd interrupts, a 10-ms period, is control transferred to the input-output main program. Even interrupts return control to the base-level program that was interrupted immediately without the necessity of storing and restoring the central control registers.

The base-level main program scheduling algorithm is basically cyclic and nonsynchronous. All processing tasks are assigned to one

of five main-program levels. These levels are executed in a prear-
ranged order so that each level is scheduled twice as frequently as
the next lower level. Each task is either permanently active or acti-
vated on request and is assigned to a given main-program level based
on the real-time requirements of the task. One additional level exists,
called interject, which allows for the insertion and execution of a
high-priority task between any main program task. This algorithm
tends to favor the real-time tasks of higher priority while insuring
that no task is deferred indefinitely.

3.2 *Processing Memory*

The processing associated with a message transmission consists of a
number of disjoint tasks or processing segments. The action taken
by the program in response to any given stimulus is a function of
the history of all preceding actions. The program uses temporarily
assigned memory areas, called processing registers, to retain the
state of the transmission and associated data during the intervals
between processing segments. A register is temporarily assigned to
a data line when a transmission is set up, remains associated with the
line while the line is active, and is released soon after the transmis-
sion is terminated. The register assigned to a message origination from
a data machine is called the input-processing register, and the register
used for message delivery to a data machine is called the output-
processing register. To provide for the simultaneous delivery of the
same message to more than one station on a given line, additional
units of memory must be assigned to the line to store the data asso-
ciated with each station. These memory areas, called processing-regis-
ter annexes, are linked to the output-processing register as needed.
The formats of the input-processing register and output-processing
register are shown in Fig. 2.

The first word of each register contains the program tag defining
the state of the processing. This tag is a relative address defining
the program routine which is expected to process the next stimulus.
If abnormal responses occur or a time-out of the expected response
occurs, the program tag is indexed to produce the address of the
particular routines designed to process these conditions. At the con-
clusion of the processing segment, the program tag is updated to
reflect the next expected response.

A single central record of each message must be maintained by the
switching center. This record contains the identities of the originator,

OUTPUT-PROCESSING REGISTER

| PROGRAM TAG |
| QUEUE & TIMING LINK ADDRESSES |
| MESSAGE LINK ADDRESS |
| LINE ASSOCIATED DATA |
| INPUT REGISTER ADDRESS |
| FIRST STATION ASSOCIATED DATA |
| ANNEX ADDRESS |

INPUT-PROCESSING REGISTER

| PROGRAM TAG |
| QUEUE & TIMING LINK ADDRESSES |
| MESSAGE LINK ADDRESS |
| STATION DATA |
| OUTPUT REGISTER ADDRESS |
| PROCESSING INFORMATION |

PROCESSING REGISTER ANNEX

| SECOND STATION ASSOCIATED DATA |
| ANNEX ADDRESS |

Fig. 2—Input- and output-processing register formats.

the message, and each terminator as well as the final disposition of each delivery. Since the number of addresses in a multiple-address message may be very large, perhaps several hundred, this central record is composed of assignable blocks of memory called message-processing blocks which are linked together in the required number. During transmission of the message, the associated processing registers are linked to the message-processing block as required. During periods of time when no active transmission is occurring for a given message, i.e., the message is merely queued for delivery, the message-processing blocks are stored on the disk memory to minimize the use of temporary memory. The format of the message-processing block is shown in Fig. 3.

During intervals of time when no active transmission is being processed for a given line, the line is constantly monitored by the polling process. A small, one-word register is used during these monitoring intervals. These registers are organized in a dedicated table, called

FIRST
MESSAGE-PROCESSING BLOCK

DISK ADDRESS
LINKING ADDRESS
GENERAL PROCESSING DATA
ORIGINATOR DATA
FIRST TERMINATOR DATA
DISK ADDRESS

LINKED
MESSAGE - PROCESSING BLOCK

DISK ADDRESS
LINKING ADDRESS
N_{TH} TERMINATOR DATA
N + 1 TERMINATOR DATA
N + 2 TERMINATOR DATA
DISK ADDRESS
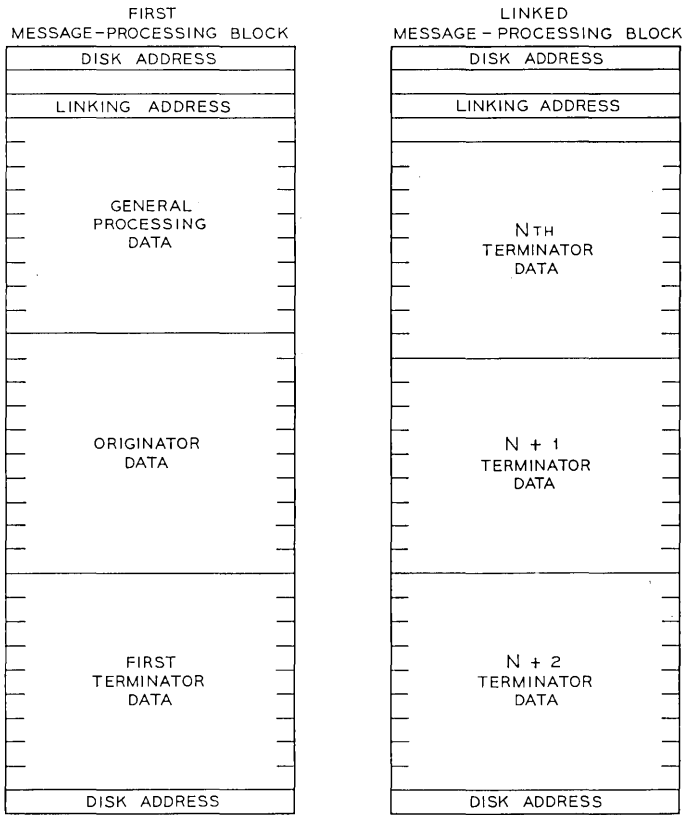
Fig. 3—Message-processing block format.

the line-status table, and indexed by the data-line number (DLN). Each word contains a line-status tag, a message waiting field which indicates the presence and highest precedence of messages awaiting output, and a polling field which is utilized by the polling process. The line-status tag is used in a manner similar to the program tag in the processing registers as a relative address to the program routine expected to process the next response from the line. The line-status table also provides the means for systematically associating the processing registers with a given line. In this case, the polling field is used to store the processing register address. The state of the line-status tag indicates the presence of a processing register so that the proper routine for processing a response can be determined from the

program tag rather than from the line-status tag. The linking of the various areas of processing memory is illustrated in Fig. 4.

3.3 *Program Classification*

The programs associated with message processing may be classified in three broad categories. First, the programs which interface with the hardware or the outside world are called input-output programs. Second, the processing programs are the high-level programs for state control which advance a given process from stage to stage. To minimize the interfaces between the program designers, only three such programs were defined, one each for (*i*) polling, (*ii*) input transmission, and (*iii*) output transmission. However, to maintain programs of a manageable size, a number of second-level processing programs were defined to perform specific functions, which were large in nature but could be characterized with a minimum of interface. Third, service routines perform functions of a general nature for all processing programs and are characterized by the memory areas which they control.
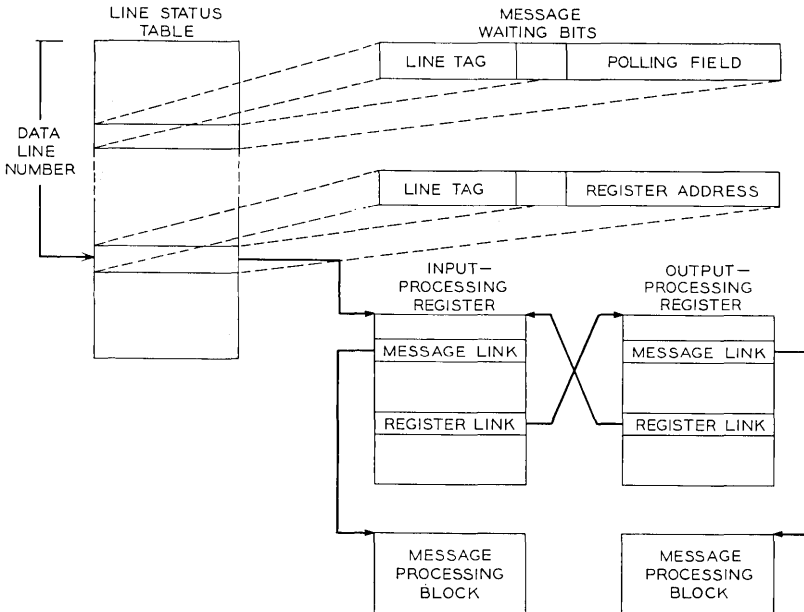
Fig. 4—Linking of processing memory areas.

3.4 *Economy of Resources*

The major resources of an electronic switching system are its storage facilities and its real-time capability. A store-and-forward switching system places greater requirements on temporary memory than a line-switched system since holding time on facilities is much greater and all data must pass through the processor. A unit of temporary memory cannot be viewed as a hardware cost alone because the total available storage affects the system capacity in much the same way as real-time capability. In the design of the No. 1 ESS ADF, each trade off between real-time capability and storage had to be viewed from a need to achieve a balance which resulted in the maximum system capacity.

IV. HARDWARE-SOFTWARE INTERFACE

One of the basic objectives in the design of the No. 1 ESS ADF was to maximize the use of No. 1 ESS equipment. The resulting equipment configuration is shown in the functional block diagram of Fig. 5. The central control, program store, and call store are equipments used in common with the No. 1 ESS. The central control guided by the program stored in the program store directs the operation of the switching center using the call store as a temporary scratchpad.

The buffer control and its associated equipment form an input-output community specifically designed for the No. 1 ESS ADF. The buffer control is a wired-logic subordinate processor which relieves the
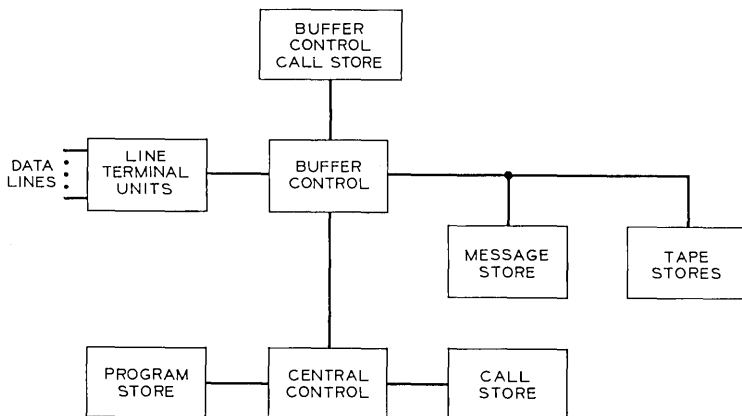
Fig. 5—Functional block diagram of No. 1 ESS ADF switching system.

central processor of many of the time-consuming taks associated with the input-output functions. The input-output community has a variable memory, called the buffer-control call store, which is used as a scratchpad in processing its own tasks and to communicate with the central control. Information destined for the central control is loaded in predetermined areas of the memory which are scanned and unloaded periodically by the central-control program. Information destined for the input-output equipment is loaded by the central-control program in other assigned areas of memory which are consulted and unloaded periodically by the wired-logic sequencers of the buffer control. Thus, this memory serves as the vehicle for the indirect interface between the input-output equipment and the central-processor program.

The program is primarily interested in communicating with three input-output communities: the data lines, the magnetic disk or message store, and the tape stores. The following sections will discuss the interfaces with the first two of these communities. The program interface with the tape system is discussed in another article in this issue.[3]

4.1 *Data Lines*

The memory-interface area for the data lines is shown in Fig. 6. Data entering the system from data lines are loaded into an area in the buffer control call store called the input-character hopper by the buffer control. This is a fixed-length common area used for all data lines. All entries in this area are time ordered and consist of two words. The first word contains the data-line number which identifies the source of the input data and a special action code. To relieve the program of the necessity of examining each character to detect signaling codes such as end-of-message, the buffer control recognizes all characters used for signaling and identifies these characters in the special-action code. The second word of the input-character hopper entry contains the three characters assembled from the line. All signaling characters produce an immediate entry in the input-character hopper. In this case, the first- and second-character positions in the second word may contain a fill character. Two pointers are maintained by the buffer control to assist in the loading and unloading of the input-character hopper. The load pointer indicates the next available entry slot. The unload pointer locates the next data entry to be unloaded by the program. The program addresses the
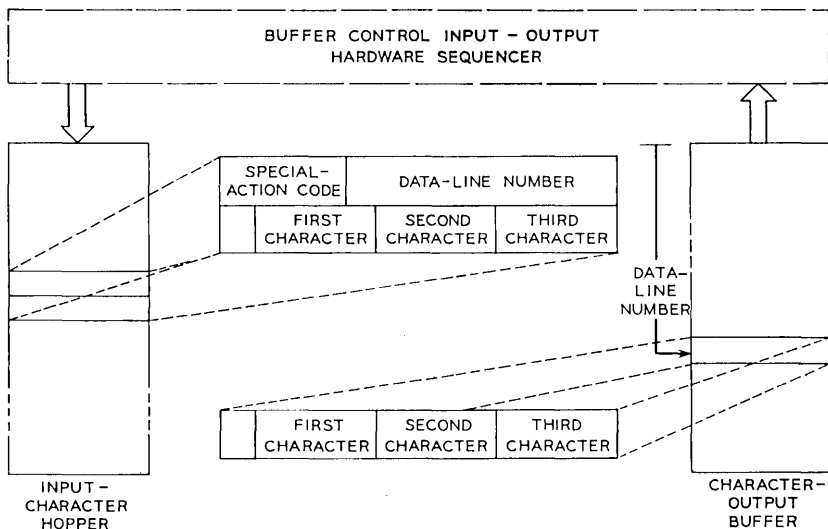
Fig. 6—Memory interface between program and data lines.

input-character hopper with a special entry address and is not cognizant of the absolute address of any given entry. This relieves the program of work associated with updating the pointers.

To transmit data to the lines, a memory area called the character-output buffer is used. This area consists of a list of words ordered by DLN, one word for each line connected to the switching center. Each word, serving as buffer storage for the associated line equipment, stores the next three characters to be transmitted on the data line. When a line equipment unit is ready to accept another set of characters for transmission, the buffer control unloads its associated word in the character-output buffer and inserts an idle code to indicate to the program that the unloading process is completed. The function of the program is to scan and load the character-output buffer at a sufficient rate to maintain the data-transmission rate on all lines that are actively transmitting.

4.2 Message Store

The memory interface area for the message store is illustrated in Fig. 7. To provide for greater throughput capability, the message store is organized into 16 sectors with the capability of transferring one

data block of 32 words per sector per disk revolution. The message-store hardware executes the block transfers as a result of instructions in the instruction queue. In order to assist the program in maintaining the instructions in the proper order for block-transfer requests, the message-store instruction queue consists of 16 dedicated instruction slots, each slot dedicated to a particular sector. The message-store sequencer is designed to sense the physical position of the rotating disks and read the proper instruction slot. The task of the program is to follow the progress of the message-store hardware and keep the message-store instruction queue loaded with new instructions. The hardware maintains an unload pointer to indicate the next instruction to be executed. The program maintains a load pointer to indicate the next instruction slot to be loaded with a new instruction.

The format of an instruction slot is illustrated in Fig. 7. The operation code directs the action of the message-store hardware while the two addresses, the message-store address and the call-store address, specify the source and destination of the data to be transferred. Upon successfully executing an instruction, the message-store hardware overwrites the operation code with an idle code to notify the program.
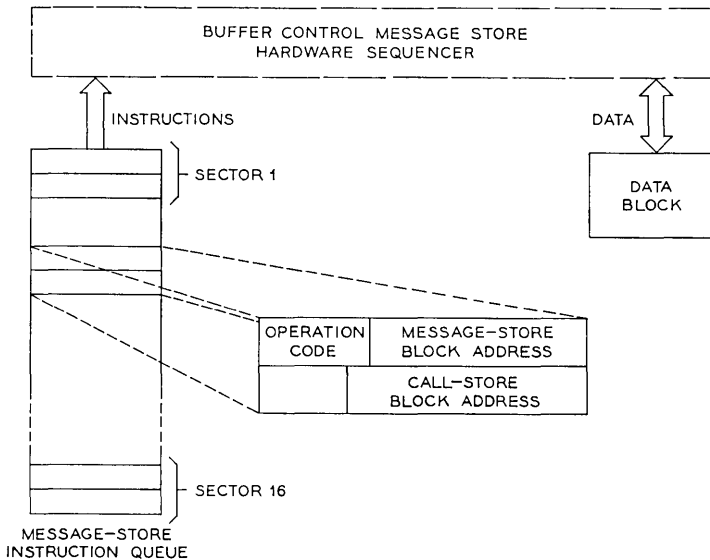
Fig. 7—Memory interface between program and message store.

V. INPUT-OUTPUT PROGRAMS

The programs which interface with the external environment are designed to be relatively simple but very efficient in their use of processor real time because of the highly repetitive nature of the functions. One is willing to sacrifice storage in the interests of maximizing the operating speed of these programs. The major input-output programs associated with the data lines and the message store are described in the following sections. Other programs, which detect carrier signals from data sets, detect maintenance states of hardware within the switching center, control-office status and alarm indicators, transmit messages to maintenance personnel, and interface with the tape stores,[3] are not covered.

5.1 *Line Input*

The input-sequence control program services the input-character hopper and does all the preliminary processing on the input data from lines. The primary function assumed by this program is the assembly of the data characters into blocks so that the message-processing programs can deal with the data in larger entities. In addition, some prefiltering of signaling characters is done before passing this information to the processing programs. The input-character hopper provides adequate buffer storage to allow this program to operate as a base-level program, thus minimizing the overhead. The input-sequence control program does not require communication buffering to the processing programs since all involved programs are operating at base level. However, time order must be maintained between the input and output stimuli handled by the processing programs. Since the output program operates on J-level, buffering of output reports is required. Thus to maintain time order between the reports, both the output program and the input-sequence control program communicate with the processing programs through a buffer called the data-service request hopper. Each entry in this hopper consists of two words, which are similar in format to the input-character hopper. The first word contains the DLN and a report-identification code. This code identifies the signaling characters in the report or processing signals such as the completion of a block of assembled data. The second word may contain the data received from the input-character hopper or a data-block address.

In processing the input data, the input-sequence control program must be cognizant of the state of the line and the next available loca-

tion in the data block for the assembly of data. The word-status table provides the memory for this function. This table has one dedicated word per line indexed by the DLN. The special action indicator in the input-character hopper and the input state from the word-status table are used as a double index to determine the proper routine for processing each entry. The input state is a five-bit code of which 11 states are currently used:

(*i*) Two states are concerned with the assembly of heading and the assembly of text.

(*ii*) Two states are used to monitor the line after a start-up code is transmitted. The first three characters of heading or text are assembled in the data block, an entry is made in the data service request hopper indicating that the station has responded to the start-up code, and the line state is changed to the assemble heading or assemble-text state.

(*iii*) Two timing states are provided for the measurement of the marking interval after a FORM FEED or tab character has been received. The elapsed time is stored in the data block for use by the output program.

(*iv*) Two signal-monitoring states are used during signaling periods on the line. Both states are similar in that all received characters result in a data-service-request hopper entry. However, one state indicates that an input has been interrupted to use the input channel for station responses in which case the true-line state has been temporarily stored in the data block associated with the line.

(*v*) One state, that is peculiar to five-level stations, is provided for the assembly of heading with all information passed to the processing programs for immediate analysis of the heading information.

(*vi*) A loop-test state is provided for a loop-around test of the line-terminal hardware. All data received during this state are stored in a specified buffer area for analysis by a maintenance program.

(*vii*) A "no operation" state is used to temporarily place a line out of service and disregard all inputs. If an invalid character is received for a given state, the signal filtering provided by the input-sequence control program results in the input state being set to the no-operation state and a data-service-request hopper entry is made to initiate abort procedures on the line.

The process of assembling a three-character word into a data block represents the primary work load for the input-sequence control program and is designed as the main line thread of logic with the greatest efficiency. The characters are assembled into 30 words of a 32-word block. The first and last words are reserved for forward and reverse link addresses which link all blocks of a message together. The five least-significant bits of the data-block address indicate into which of the 32 words the assembly is to be made. After assembly, these five bits are tested to determine if the block is filled. If the end of block has not been reached, the data-block address is incremented and reinserted in the word-status table. If the last word of the block is used, a new data block is seized and its address is placed in the word-status table and the last word of the data block just assembled. An entry is made in the data-service-request hopper to notify the processing programs of the completed block. If the flow of input characters should halt, as a result of a stuck station transmitter or broken tape, the lack of stimulus and assembly of further characters would go undetected. To detect this condition, called intercharacter time-out, a timing bit is assigned to the word-status table. An executive control routine sets this timing bit every 32 seconds. The assembly process resets the bit upon updating the data-block address. If the executive control routine, during its periodic visitation, finds the bit set while the line is in an assembly state, an intercharacter time-out is declared and the originating user is informed through a service message.

The input-sequence control program performs a rather important function which tends to reduce the work load on the output program. Input data which requires special action by the output program, such as tabbing functions or end-of-text (ETX) characters, are specifically flagged so that the output program does not have to scan and test each character that is being transmitted in order to detect these functions. The flagging process results in a two-word assembly in the data block. The characters involving the special action are assembled in the first word and an output control code which defines the special action is assembled in the next word. The sign bit of the word containing the output-control code is set to distinguish this as a control word rather than data.

The processing functions performed by the input-sequence control program are illustrated in Fig. 8. Upon reception of an ETX character when only one word in the data block remains to be filled, an additional data block may be seized to assemble the output-control
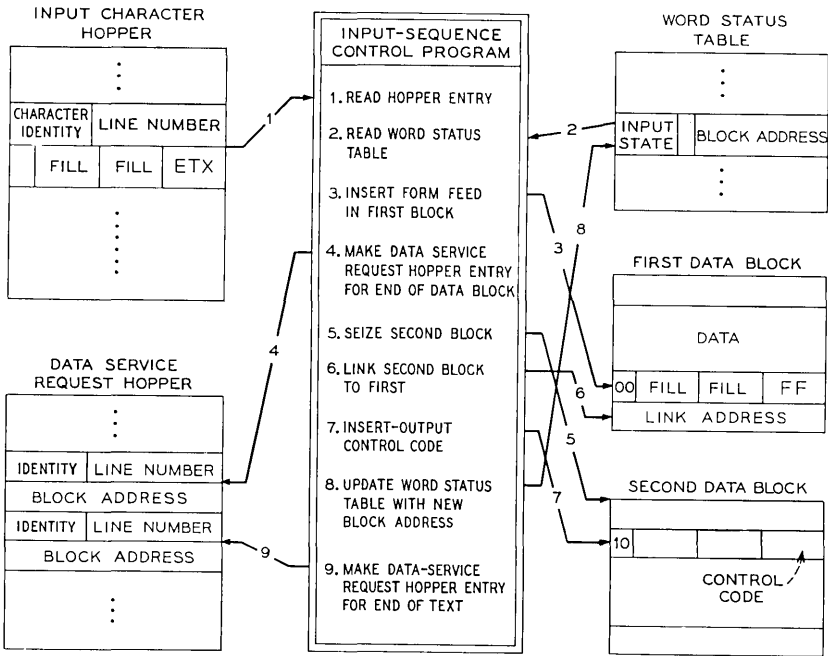
Fig. 8—Illustration of functions performed by input-sequence control program.

code and two data-service-request hopper entries are made, the first indicating the end of assembly in the original data block and the second indicating the completion of the message. The work performed on this unit of input information is unusually large but illustrates the type of functions performed by this program.

5.2 Line Output

It is the output-sequence control program which delivers data to the lines. This involves the scanning of the character-output buffers, mentioned earlier, and the orderly word-by-word disassembly of data blocks as each idle buffer is detected. The data blocks comprise parts of messages received from input lines and retrieved from message store, service messages generated by the processing programs, or blocks of fixed data used in the signaling dialog with the stations. The output program is not cognizant of the source or purpose of the data. It merely goes about its task of processing words of data at the direction of other programs which manipulate the data blocks.

The character-output buffers must be scanned periodically at a rate

fast enough to maintain the character rate of the associated line. For line rates of 150 bits per second, a visitation period of less than 200 ms is required. To achieve the required timing, the output sequence control program is operated as a J-level interrupt task with a 10-ms periodic entry. It would be desirable to scan only active lines to maximize the scanning efficiency, and equalize the work during each entry to minimize the effects of peaking. Both of these objectives are achieved by driving the output-sequence control program from a work list of active lines called the output work table. For all lines of 150 bits per second or less, this table consists of 18 subtables of n words each as shown in Fig. 9 where n is chosen to meet the output capability of the office. On each J-level entry the output-sequence control program services one subtable, so that the character-output buffer for each active line is scanned once every 180 ms. Each entry
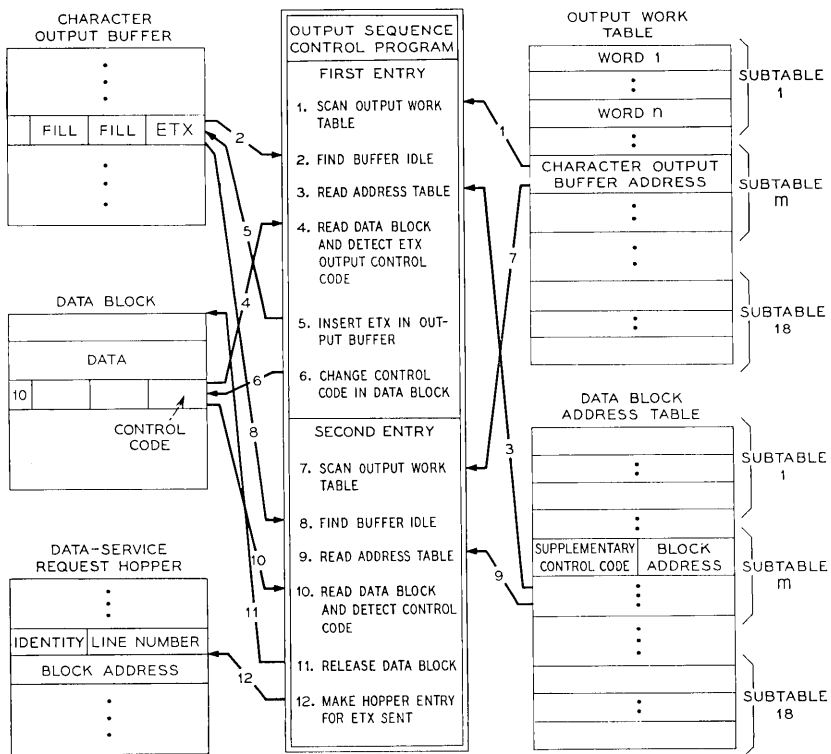
Fig. 9—Illustration of functions performed by output-sequence control program.

contains the character-output buffer address of an active line requiring scanning.

A second table, which complements the output work table, is used to store pointers to the next word to be transmitted to the active lines. For each entry in the output work table, there is a companion entry in the data-block address table in the same relative position. This relative position is defined as the data-block index.

A processing program that wishes to transmit a block or group of blocks of data to a given line must seize an idle location in the output work table. The busy-idle state of each entry is maintained in a matrix in which each column defines a subtable of the output work table and a row defines the word position in each subtable. The seizure process, a systematic search of the matrix from top to bottom, results in a concentration of work in the upper portion of each subtable and equalizes the work in each subtable. Upon seizing an entry, the processing program loads the data-block address table with the address of the data block to be transmitted and the output-work table with the character-output buffer address of the desired line.

A data-block entry which has the sign bit set to one indicates the presence of an output-control code which informs the output program that some special action is required. This allows the output-sequence control program to perform the task of block disassembly in the fastest, most routine, repetitive manner possible. Output control codes are inserted in data blocks by the input-sequence control program when assembling data blocks or by the processing programs when generating data blocks for output transmission. There are some 49 output-control codes defined to perform such diverse actions as the detection of the end of signaling sequences or units of a message, insertion of transmission-free timing intervals on the lines, transmission of break signals, and detection of data-block format errors. While some of the codes result in special action and then the continuation of the disassembly process, the majority of the codes result in a cessation of further disassembly and a report to a processing program so that it can perform some action at that point in the disassembly process. These reports are buffered to the base-level programs through the data-service-request hopper. In making entries in this hopper, the output-sequence control program must algorithmically convert the character-output buffer address to the DLN.

A secondary control mechanism is provided through the use of the supplementary-control code stored in the data-block address table. These codes deal primarily with halting further disassembly for one of

a number of reasons. For example, one encoding of this item stops further output while waiting for the next block to be retrieved from the message store. In addition, if a processing program wishes to halt the output transmission to poll the station or transmit a service message concerning an input transmission, a supplementary-control code is written into the data-block address table. Detection of this code by the output program results in a halt of further disassembly and a report in the data-service-request hopper.

Figure 9 illustrates the processing functions performed by this program when it detects the output-control code for the end-of-text indication assembled by the input-sequence control program. For this particular case, the output-sequence control program transmits the ETX character and changes the output-control code in the block. It is only upon detecting this second code that a data-service-request hopper entry is made. This ensures that the character-output buffer has been unloaded before the processing programs are informed that the message delivery is complete.

5.3 *Message Store Input-Output*

The last of the major input-output programs performs both input and output transfers of data blocks to and from the message store. This program, called the message-store administration program, consists of two functionally independent parts: (*i*) the message-store input-output routine, a J-level program, which services the message-store instruction queue, and (*ii*) the message-store service routines which maintain a busy-idle status for each block of message store and perform common functions required to interface with the input-output routine.

Figure 10 illustrates the memory areas associated with the message-store program. A processing program that wishes to store a block of data on the message store makes a direct transfer to a message-store service routine. There are as many entry points to the service routines as there are services performed by these routines. The first task of the service routine is to assign a destination disk address for the data block residing in call store. This is done by consulting a busy-idle map. Each block is represented by a single bit in the map. In searching the map for an idle block, a pointer is used to keep track of the last map word that was read. In this way, those map words most likely to contain idle blocks are searched. The number of words scanned in a search is limited to prevent real-time abuses when disk usage is near capacity. If an idle block is not found within the limit,
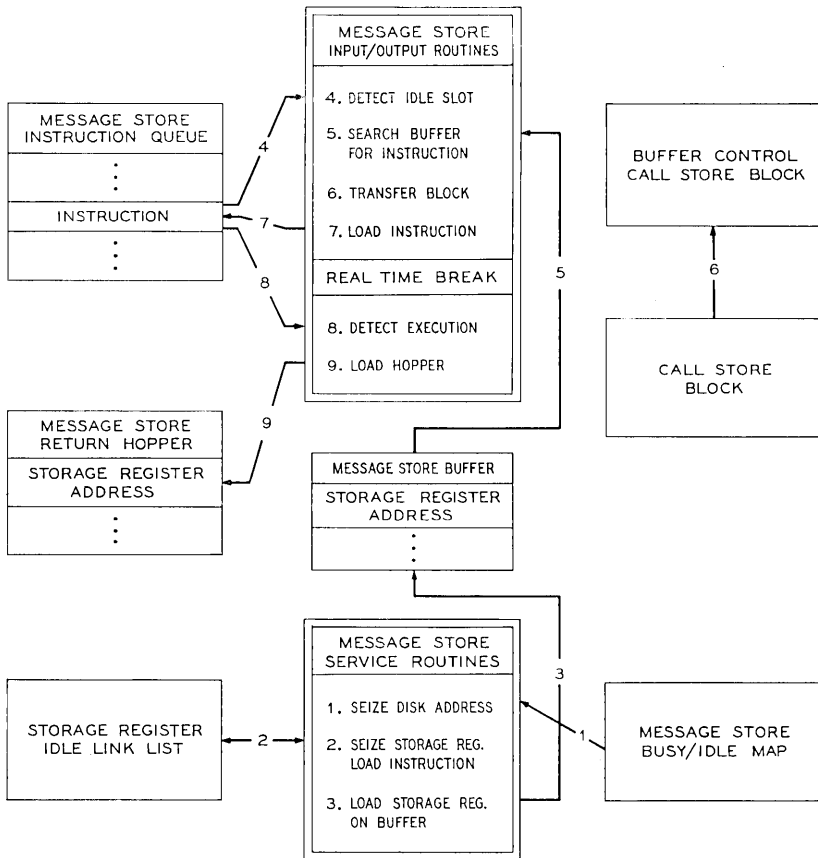
Fig. 10—Illustration of functions performed by message-store administration program.

a failure return is given to the processing program. The resulting address defines the sector and entry location in the message-store instruction queue into which the instruction must be loaded. Because the desired instruction slot may not be presently available, a waiting list of instructions must be maintained. This waiting list is called the message-store buffer. Since the message store is sector oriented, this buffer is also sector oriented and consists of 16 link lists. The linked-memory areas are called storage registers. The second function of the message-store service routines is to seize an idle storage register, load the instruction data, and link the register to the appropriate sector-oriented message-store buffer link list. The instruction data consists

of the instruction, the message-store address and the call-store address. In addition, a processing-register address is needed to associate the transfer request with a particular data call. The notification of the successful block transfer is directed to the processing register so designated.

The message-store input-output program, a J-level program, monitors the action of the message-store hardware in executing the instructions. On finding that a given slot in the message store instruction queue is idle, the corresponding link list in the message-store buffer is consulted to determine the next block to be transferred. Since the message-store hardware can only access the buffer-control call store, the data block must first be transferred from its call-store location to a buffer-control call store. A dedicated area, one block per sector, is defined for this purpose. The instruction is then generated and stored in the message-store instruction queue.

At some later time, the input-output routine will detect that the aforementioned instruction was executed. At that time, the storage register, used initially for the transfer request, is entered into a link list called the message-store-return hopper. This hopper acts as a buffer between the J-level input-output routine and the base-level processing programs for notification that the requested transfer has been successfully completed.

The message store is organized on a block basis to achieve adequate throughput with a mechanically rotating device having large access time. The message-store administration program provides the capability of single-word transfers. In the case of a single-word write instruction, the input-output routine generates two instructions: (i) read the data block in which the one word of data is written, and (ii) write the resultant data block back to the message store.

In storing messages and message-processing blocks on disk, the blocks are linked together so that the data can be referenced with a minimum number of addresses stored in call store. This linking scheme is illustrated in Fig. 11. The first and last words of each block contain reverse and forward link addresses and only the 30 central words contain data characters or processing information. The double linking is provided so that the program can retrieve the data using either the initial data-block address or the final address. In storing a block of data on message store, the address of the next block must be known to insert the proper link address in the data block. Thus the service routines provide separate entries for a first-block transfer, two message-store addresses are seized; for a middle-block transfer, one mes-
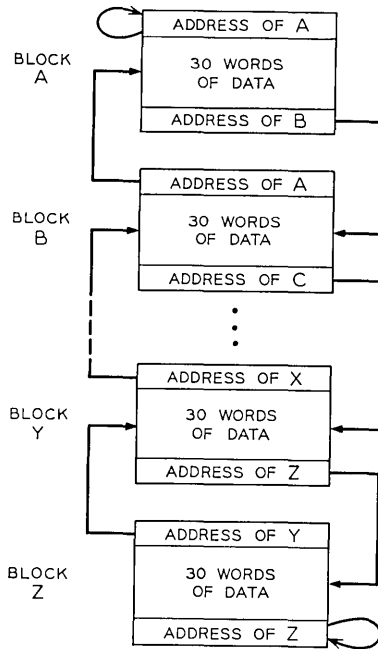
Fig. 11—Illustration of double linking of data blocks.

sage-store address is seized; and for a last block transfer, no seizure is required.

VI. MESSAGE-PROCESSING PROGRAMS

The message-processing programs, each of which handles a given phase of processing, are relatively complex because the design covers many station arrangements, line types, codes, and optional features. The decision to incorporate all station characteristics and options within a single program was made to achieve efficient use of common logic functions. In the following sections the processing of a typical message is used to describe the functions of these programs.

6.1 Polling

The polling program consists of several major functions which provide the segmented control for this cyclic process, as illustrated in Fig. 12. To observe the operation of this program, the cycle will be entered at the point of generating the polling code. The polling pro-
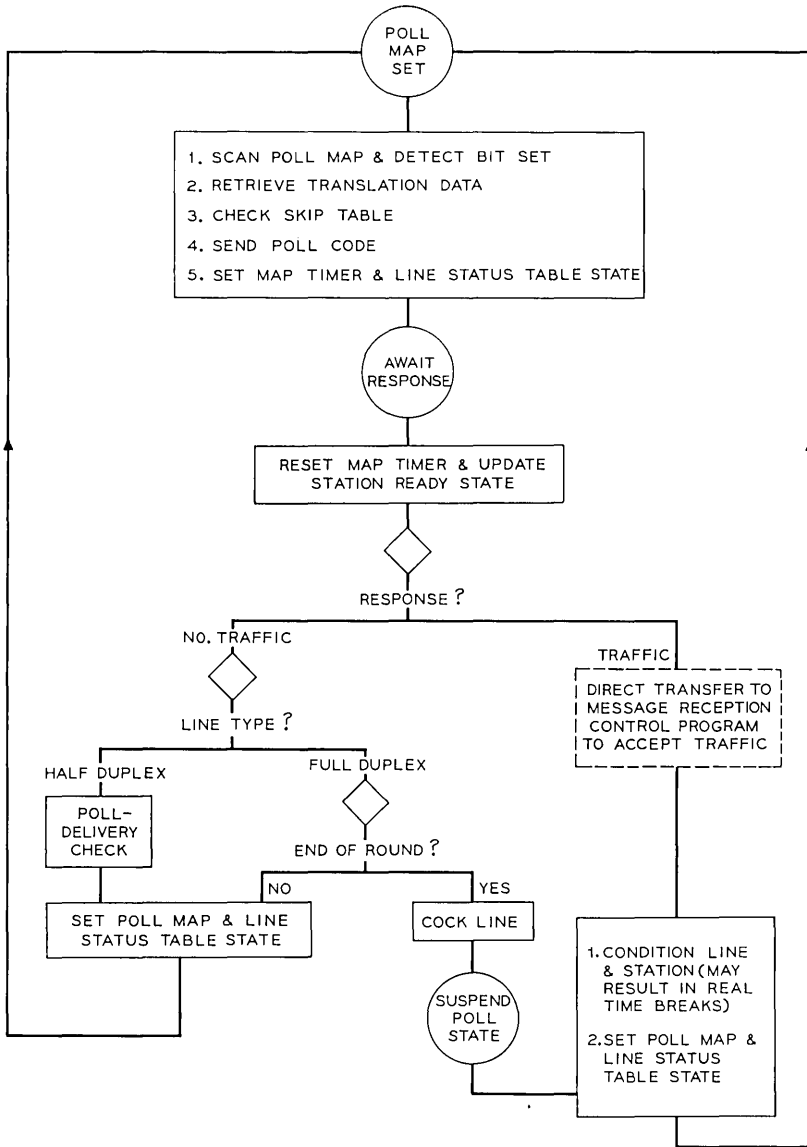
Fig. 12—State diagram of polling program.

gram is entered periodically as a base-level task by the executive control program to poll a given number of lines, the number being an office parameter. The particular lines to be polled are defined in the poll map, a dedicated area of call store in which each line is represented by a single bit. The program scans the poll map; and when it detects a bit set to one, it interprets this as an indication of a line to be polled. The translation service routines are used to retrieve the characteristic data about the line: code, type, and options available to the line. The program then consults the skip table for the line to determine the next station to be polled. The basic function of the skip table is to indicate, with a particular bit pattern, which stations on a line are to be polled or skipped and, with another bit pattern, which stations have indicated in the past a receiver not-ready status. The skip table also contains a counter, called the poll-table index, which indicates which of the stations on the line should be polled next. The skip-table bits are examined for the station referenced by the poll-table index and the station is polled if the station is not on skip or if it is not-ready. Stations on skip, but not-ready, must still be polled to detect the transition from the not-ready to ready state so that traffic may be delivered to the station. The poll-table index defines the specific poll code to be sent to the line. This code is loaded into the character-output buffer directly and a per-line timer is set to make certain that stimulus on the line is not lost if the station should fail to respond. The line-status table state is updated so that the response will be passed to the proper polling routine for analyzing the response. The poll-code generation segment is then complete.

It has been noted that the station response to polling is timed. In fact, all expected signaling responses are timed. This function is administered by an executive control timing routine which makes use of two bits of memory per line organized into two maps similar to the poll map. The timing routine examines each map alternately at the end of three-second timing periods. During odd intervals the first map is set by the processing programs for response timing. During even intervals while the second map is being set, the first map is allowed to time the responses. Any successful response results in the processing-program resetting the corresponding bits of both maps. At the end of the even interval, the timing routine will examine the first map; and any bits still set indicate a response time-out. The line-status table state is indexed to produce the proper entry point in the processing program required to produce the necessary actions as a result of the time-out. The map-timing technique achieves a large economy in

storage allocation in comparison to conventional link list timing facilities and ensures that a timing facility is always available for every line. The penalty is a sacrifice in accuracy, $\pm \frac{1}{3}$ of the average response time-out.

Under normal conditions, the polling response will be received by the line terminal hardware and entered in the input-character hopper. Upon its periodic visitation of the input-character hopper, the input-sequence control program will detect the entry and, based on the input state of the word status table, will make an entry in the data-service-request hopper. This entry will be passed to the polling program for analysis of the response. If the polling response indicates a request to originate traffic and the priority options of the line allow the present request to be serviced, control is relinquished by a direct transfer to the message-reception control program, which services the request after updating the ready status in the skip table. The polling process has thus been interrupted to accept the message origination. The polling cycle for this line will be restarted at the conclusion of the message origination.

If a no-traffic response is received for a full-duplex line, the polling program must decide whether or not to poll the other stations on the line. If each station has not been polled once since the cycle was started, the poll map is set to initiate polling of the next station. If the polling round is complete, the polling program will suspend further polling by loading the character-output buffer directly with a set of characters to "cock" the line. The line-status table state is updated to reflect an idle state and an additional bit is set to indicate that the line is in the suspend-poll state. The polling will restart on the next quarter hour by an executive control task which examines the suspend poll bit in each line-status table entry. An overview of the polling process, illustrating the relationship between the program and the involved memory areas, is shown in Fig. 13.

On HDX lines, traffic can proceed in only one direction at a time and so the input and output of traffic are necessarily in contention for use of the line. On reception of a no-traffic response, the polling program must determine its next action based on a poll-delivery criteria which establishes a precedence order between the four delivery priorities and the two origination priorities as follows:

(i) deliver all urgent and rush traffic,
(ii) pick up priority traffic (lead-round polling),
(iii) deliver all normal traffic,
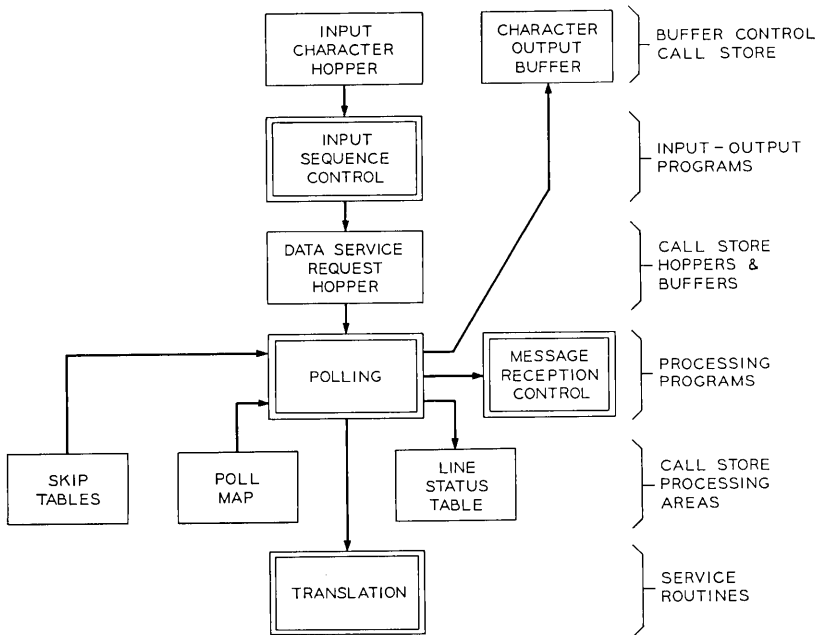
Fig. 13—Relationship between program and memory areas used in polling process.

(iv) pick up of regular traffic (general-round polling),
(v) deliver all deferred traffic, and
(vi) continue polling (subordinate round polling).

To make the proper decision, the polling process for HDX lines is sub-divided into three levels or polling rounds, as indicated in the poll-delivery criteria: lead round, general round, and subordinate round. Memory of current polling round is maintained as an item in the polling field of the line-status table. This item, in conjunction with the message waiting bits, determines the next action to be taken by the polling program. If a delivery of a message is indicated, the polling program relinquishes control of the line and initiates the delivery process by setting a bit in the nomination map, which is discussed in greater detail in Section 6.7. If polling is to continue, the poll map is set to initiate the polling of the next station on the line.

6.2 *Heading Origination*

As noted in the preceding section, the message-reception control program is entered by the polling program when it is found that a

station is ready to originate a message. Using a set of general service routines, the message-reception control program: (*i*) seizes an input-processing register and links it to the line-status table, (*ii*) updates the line-status table state to indicate that the polling field now contains a register address, (*iii*) seizes input capacity—a fictitious facility used to control the input load presented to the system, and (*iv*) seizes a message-processing block and links it to the input-processing register. Any failure to seize a given item results in queuing for that item. The translation program is then entered to retrieve translation data concerning the originating station. The data defining the line characteristics and station options are loaded into the input-processing register, and the message-processing block is loaded with the originator's identity and the user's identity. The user in this context does not refer to an individual station but rather to the community of stations composing a given network. In addition, the program also seizes an output-work table assignment for later use in transmitting the generated heading to the originator. As noted earlier, this assignment is in the form of a data-block index which is stored in the input-processing register. Next the identity of the originating station is verified. The message-reception control program loads the character-output buffer with the originator's call-inquiry code obtained from the translation data and sets the response timer. If the proper station has been identified, the station will respond with a start-of-heading character.

When the station response is received through the data-service-request hopper, the program initiates the construction and delivery of the generated heading to the originator. A data block is seized and a direct transfer is made to the heading-generation program. Based on the station options specified in the translation data stored in the input-processing register, the heading-generation program assembles the desired heading information in the data block. This information will generally consist of a message number, time and date, and a start-of-text (STX) control character which restarts the originator's transmitter, allowing the heading portion of the message to be transmitted to the switching center. In addition, the heading-generation program inserts the message number, time and date into the message-processing block independent of the station options. Upon return to the processing program, the message-processing block is stored on the message store. The service routines provided by the message-store administration program are used to seize a block address in a specified area of the message store and load a storage register on the message-store buffer to transfer the message-processing block to disk.

The message-processing block now contains an identification of the originator and message to be originated. The purpose of this operation is message protection. If the message is lost in the system for any reason, a record of the message can be obtained from the message store. Next the data block address is loaded into the data-block address table, and the DLN is algorithmically converted to a character-output buffer address and loaded into the output work table to initiate the delivery of the generated heading to the originator.

The output-sequence control program systematically transmits the data to the station. After transmitting the restart code to the station, the program detects an ouput-control code inserted in the data block by the heading-generation program. This code directs the program to terminate the transmission by inserting an idle state in the supplementary control code of the data-block address table. The data-block address is removed from this table and inserted in the word-status table so that the block can be used for the assembly of the input heading. A data-service-request hopper entry is made to inform the processing program that the station transmitter has been started. In response to this entry, the message-reception control program sets the response timer to monitor the line for the receipt of the first three characters of heading as an indication that the station has responded to the restart code. The program tag in the input-processing register is updated for this monitoring state.

Upon receiving the first three characters of heading, the input-sequence control program assembles the characters into the data block and makes a data-service-request hopper entry. In response to this entry, the message-reception control program resets the response timer, releases the output-work table assignment, and updates the program tag to the receive-heading state. The input-sequence control program continues to assemble the heading into data blocks. As each block is completed, the message-reception control program is informed through a data-service-request hopper entry. Each block is ordered to the message store. Upon acknowledgment from the message-store administration program that a given block has been transferred to disk, the data blocks are relinked and retained in call store for the analysis of the heading. When the end-of-heading indication is received, the last block of heading is ordered to disk and the program tag is updated to a receive-text state. For FDX lines, the assembly of the text will continue concurrently with the analysis of heading. When all heading blocks have been acknowledged, the message-reception control program initiates the analysis of the heading indirectly

by loading the input-processing register address on the heading-analysis hopper.

6.3 Heading Analysis

The heading-analysis program services the heading-analysis hopper. The analysis consists of an examination of each character of heading assembled in the data blocks, formatting of certain of the characters for translation of the data, and the storing of information in the message-processing blocks in a form appropriate for future action by other programs in the processing of the message. When the analysis is complete, no other program will need to examine the data characters on an individual basis.

The heading format for a data message is quite rigid and may consist of the following units of information separated by delimiters:

(i) a 10-digit directory number,
(ii) a mnemonic code, which is selected by the user to address a single station or group of stations in the user's realm of interest,
(iii) a precedence designator, which defines the priority order of delivery to each terminator, and
(iv) personal address information, which can be used to define an individual or department being served by a given station.

The delimiters and maximum number of characters per unit of information are shown in Table II. For action requests which ask that the switching center perform given functions rather than the delivery of a message, additional units of information are defined as shown in Table II. The input processing of an action request is indistinguishable from message processing, with the exception of the heading analysis. At the completion of the transmission of an action request, second-level processing programs are called upon to execute the request based on information loaded in the message-processing block by the heading-analysis program.

The analysis of the heading is a table-driven process. Each character is used as an index into a 128-word table, one word for each character in the ASCII code set. If the word in the table is not flagged (sign bit = 0), the word contains a six-bit stripped ASCII character which merely compresses the subset of alphanumeric characters to conserve storage. If the word in the table is flagged (sign bit = 1), the word contains a set of indexes used with other tables. The particular choice of index and second table depends on the unit

TABLE II—THE UNITS OF INFORMATION, THE DELIMITERS, AND THE CHARACTERS PER UNIT THAT MAY BE INCLUDED IN A HEADING

Unit of Information	Starting Delimiter	Ending Delimiter	Maximum Count of Characters
For Message or Action-Request Headings			
Mnemonic	Alphanumeric or Hyphen	CR, LF, Space or STX	7
Directory Number	<	>	10
Precedence	Alphanumeric or Hyphen	CR, LF, Space or STX	7
Personal Address Information	[	]	63
For Action Requests Only			
Action-Request Order	/	/	7
Data Line Number	Numeric	CR, LF, or Space	4
Circuit Number	Alphanumeric or Hyphen	CR, LF, Space or STX	14

of information being processed. These second-level tables contain addresses of the particular routine necessary to process the given character.

When a given unit of information has been assembled and identified, it is usually necessary to translate and store the information in the message-processing block in a form more useful for the other programs. For example, the result of a mnemonic translation is a normalized directory number, which is a compressed form of the directory number used for routing traffic within the switching center. Since mnemonics are chosen completely at the discretion of the user, a set of mnemonic translation tables are required per user set of stations. To translate a mnemonic, the heading-analysis program provides the translation program with the mnemonic and the user identity which was stored in the message-processing block by the message-reception control program.

As each addressee in the message is determined, a terminator slot in the message-processing block is loaded. As additional storage is required, the heading-analysis program seizes additional message-processing blocks and links them to the original block. A mnemonic may translate to either a single terminator or a group of terminators. For each terminator the normalized-directory number is loaded into the terminator slot. Before loading a terminator slot, certain screening to eliminate duplicate addresses is done. The originator is always screened from a group code; and if a terminator has a single-copy option, the resulting normalized-directory number is screened against all previously established terminator slots. Also loaded into the terminator slot is the relative address of the first character of the mnemonic and the number of characters. This information is required by the heading generation program to retrieve the relevant address in constructing the terminator's generated heading. The location of the first character of the mnemonic is a relative address consisting of the heading block number, word position, and character position in the word. The absolute address cannot be inserted in a terminator slot because each time a program needs the heading block the call-store address may be different, having been written from message store to call store. If a given addressee has been eliminated because of the single-copy option, a terminator slot is still established to store the relative address of the mnemonic. This allows the heading generation program to retrieve all the relevant address information used in the heading to address a single station. When a personal-addresss information unit is identified, its relative address is also inserted in the appropriate terminator slots.

At the completion of heading analysis, the state of processing the incoming message is unknown since the assembly of text occurs concurrently with the analysis. The heading-analysis program therefore transfers indirectly to the message-reception control program using the program tag in the input-processing register. If a format error or nontranslatable mnemonic is detected by the heading-analysis program, an indication of this condition is given to the message-reception control program. In this way the input message may be aborted with a service message to the originator indicating the field or unit of information that was found to be in error. As a result of this transfer, the message-reception control program updates the program tag to indicate the completion of heading analysis and releases the call-store copy of the heading.

6.4 *End of Originated Message*

The message-reception control program continues the processing of the input message. A data-service-request hopper entry indicating the completion of text assembly in a given data block results in the data block being ordered to disk. As each block transfer to disk is acknowledged by the message-store administration program, the message-reception control program releases the call-store copy of the data block. When the first block of text is stored on disk, and if the heading has been analyzed, the message-reception control program calls upon the message-queue insertion program as a service routine to generate a delivery-queue entry for each terminator whose line speed is less than or equal to that of the orignator. This allows for rapid speed of delivery, i.e., the delivery of the message can be started while the same message is still being originated. The process of generating the delivery queues are discussed in greater detail in the next section.

When an end-of-text entry is received by the message-reception control program, the complete message has been received from the station. Continued processing of the message can occur without further association with the line. The input-processing register is disassociated from the line by removing the register address from the line status table and setting the line status tag to an await-End of Transmission (EOT) state. The last text block is ordered to disk and the program tag in the input-processing register is updated to an end-of-message state. The message-reception control program continues to control both the line and the message independently through the line-status table and the input-processing register, respectively.

The await-EOT state is used to monitor the line for multimessage transmission. A bit in the line status table is set to initiate intermessage timing on the line. If no further stimulus is received from the line, an intermessage time-out will occur in approximately 48 seconds. The message-reception control program initiates a line-abort procedure which transmits an emergency-stop sequence of characters on the line which in turn normalizes the stations on the line. If a start-of-heading character is received indicating that the station wishes to transmit an additional message, the message-reception control logic for premessage initialization is entered. If an end-of-transmission character is received, the intermessage-timing bit is reset and input capacity which has been held during the duration of the transmission is released. For FDX lines, the polling program is entered to condi-

tion the line for polling and to set the poll map which initiates poll-
ing on the line. For HDX lines, the choice between delivering a
message to the line or polling the line for additional input traffic is
based on the poll-delivery criteria discussed earlier. Based on the
message-waiting bits in the line status table and the current polling
round, either the poll map is set to initiate polling or the nomination
map is set to initiate the delivery of a message.

The state word in the input-processing register is used to control
further processing of the message. The entries that are handled dur-
ing the end-of-message state consist of message-store returns which
acknowledge the successful transfer of data blocks to disk and, in the
case of messages with very short text length, heading-analysis re-
turns indicating the successful completion of the processing of the
heading information. After all data blocks have been acknowledged
and the heading successfully analyzed, the message-reception control
program relinquishes control of the message to three programs which
complete the input processing. First, the permanent file program is
entered to update the retrieval tables stored on the message store
for each station, which aids in the rapid retrieval of messages re-
quested by the users. This program is one of a group of programs
which control the tape subsystem.[3] Second, the message queue in-
sertion program is entered to established a delivery request, i.e., a
message queue entry for each addressee that had not been processed
when the first block of text was received from the originating station.
Third, the message-termination program is entered to complete the
processing of the input message.

The primary functions performed by the message-termination pro-
gram are billing, traffic counts, and the final disposition of all call-
store facilities associated with the message. The program transfers
control to the automatic-message-accounting (AMA) program as a
service routine which updates customer summary-billing registers asso-
ciated with the originator. This data is transferred to an AMA tape
once per day. The message-termination program then uses several sub-
routines to update system and user-traffic statistic counters. The user-
traffic statistic data concerning all stations in the user set are periodi-
cally delivered to the user-control stations so designated by the user.
When these tasks are completed, the message-termination program de-
termines from an item of storage in the message-processing block if
the call-store copy of the message processing block is being used by
any other program; e.g., the message is being delivered to one or
more of the addressees. If it is not being used by another program,

the message-termination program transfers the message-processing block from call store to disk, using the message-store administration-service routines. When this transfer is completed, the input-processing register and the call-store copy of the message-processing block are released. If another program is using the message-processing block, then only the input-processing register is released. The transfer of the message-processing block to disk is thus left to the last program using the call-store copy.

6.5 *Generation of Delivery Queues*

The heart and substance of a store-and-forward switching system is the message queues, whereby the system may hold many messages for a particular terminator to insure maximum utilization of the transmission facilities. The message-queue insertion program generates the message queues and administers other special functions which affect the message queues. For example, one such special function is the shifting of entries from one queue to another when the original terminator is placed on alternate delivery. The message-nomination program, which is discussed in a later section, selects entries from the queues and initiates the actions required to deliver messages to the terminating stations.

The requirements and objectives imposed on the message queues lead directly to the memory organization. First, the entries in the message queue must be time ordered within four precedence categories: urgent, rush, normal, and deferred. Second, the holding time on the call-store facilities which provide the message-queue function is very large, so that the efficient use of such facilities is highly desirable. In addition, the different functional areas should be administered from a single pool of call-store facilities to provide the greatest efficiency under different traffic mixes. Third, the structure of the message queue must lead to a single call-store copy of the message-processing block when processing concurrent deliveries to several lines. Besides the economic reasons, this is necessary to insure that the results of one transmission do not overwrite and destroy the results of another transmission when attempting to update the disposition information on the disk copy of the message-processing block.

Each data line or multiline hunting group is assigned a unique queue index which for convenience is identical to the poll index. This index identifies the memory in two fixed tables associated with each line or multiline hunting group. The first of these tables is the nomination map which has a format similar to the poll map. It contains

one bit for each queue index, i.e., for each message queue, and provides the stimulus to service a given queue. A given bit which is set to the "1" state indicates that the associated queue contains one or more messages and that the line is available for transmitting traffic. The second table is the message-queue table which contains two words for each queue index, as shown in Fig. 14. The first word contains a pointer to the queue-location register if a queue currently exists; otherwise the word is zero. The second word contains a count of the number of messages currently in the queue, which is used in making queue reports to the user-control location. The queue-location register is a three-word call-store facility which contains pointers to the subqueues which are organized by precedence. The first word points to the high-priority subqueue which contains both urgent and rush messages; the second word points to a normal-priority subqueue;
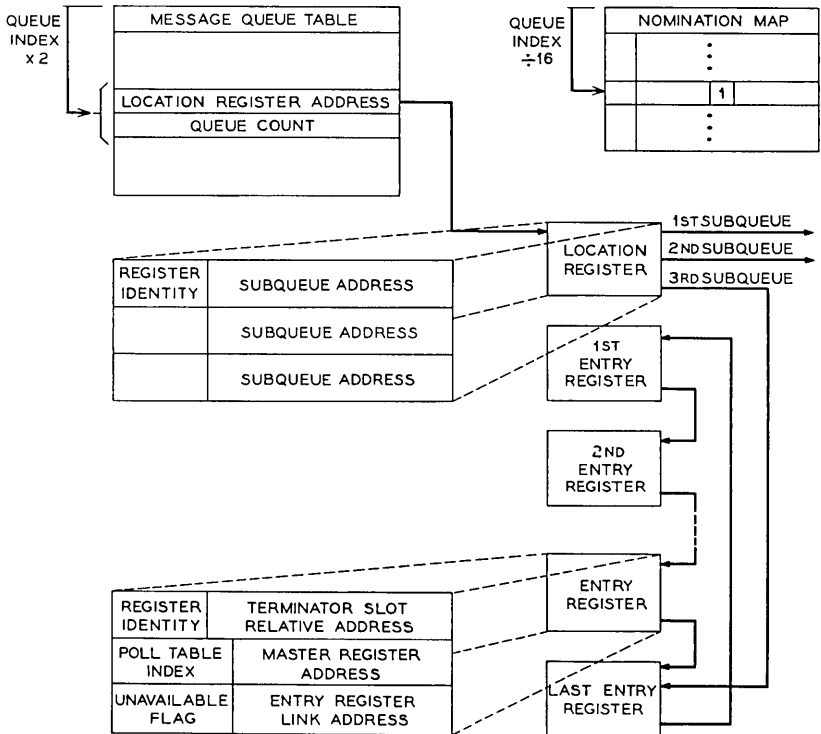
Fig. 14—Organization of message queue memory areas showing the circular linking of entry registers.

the third word points to a deferred-priority subqueue. The urgent
and rush messages are combined in a single subqueue so that the
queue-location registers can be assigned from a common pool of three-
word registers used for all message-queue functions. The percentage
of messages using the two higher-precedence categories is very low,
so that combining these precedences in a single subqueue does not add
an appreciable work load for the processing associated with this sub-
queue.

Each subqueue is made up of three-word facilities called entry regis-
ters which are linked together to form a one-way circular link list as
shown in Fig. 14. One entry register is used to refer to each addressee
in a given message. The subqueue pointer in the queue-location register
points to the last-entry register in the subqueue. The circular nature
of the link list makes it possible to get to the last entry for adding
additional entries to the queue or to the first entry for servicing the
queue in a very few logical steps with a minimum of storage.
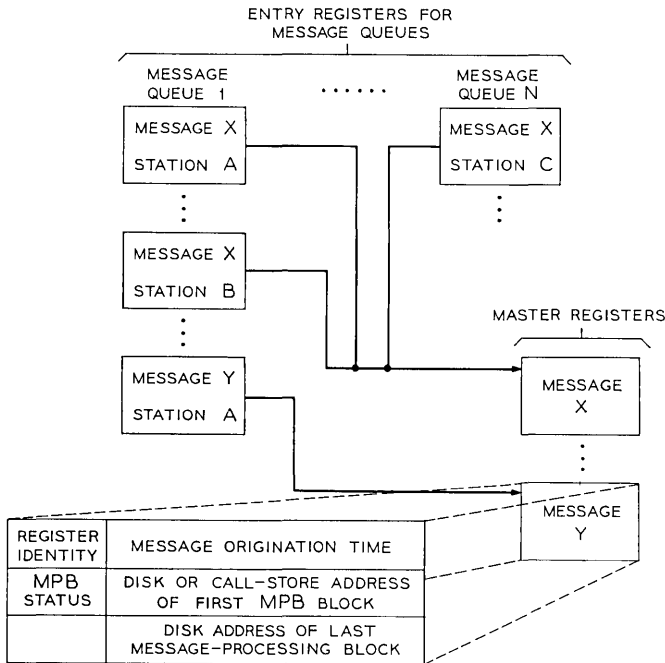
Fig. 15—The master register provides a single common linkage to the message-
processing block for all message-queue entries associated with a given message.

Associated with each message is a three-word call-store facility called a master register. The master register indicates whether the message-processing block is located in call store, message store, or in the process of transfer to or from call store and the address of the first block of the message-processing block. The entry registers for each addressee in the message point to the master register, as shown in Fig. 15. The master register provides a common linkage to the message-processing block for all message queue entries. This allows deliveries to all terminators to be initiated independently while maintaining at most a single copy of the message-processing block in call store. As with queue-location registers and entry registers, the master registers are seized from a common idle-link list of three-word call-store facilities called message-queue registers. An item in the register is used to identify the function assigned to a given facility.

In the normal processing of an input message, the message-queue insertion program is entered by the message-reception control program for the purpose of forming the message queues. This occurs after the first block of text is assembled and also at the completion of the input transmission. To perform its functions, this program requires an available processing register which is linked to the message-processing block. If a master register does not exist, the message-queue insertion program seizes one and initializes it with the appropriate data. The program then proceeds to process sequentially each terminator slot in the message-processing block that was established by the heading-analysis program. When an unprocessed-terminator slot is found, the addressee's directory number is used as an input to the translation subroutines to obtain all of the station related information required for queuing. A privacy check is made, based on the translation data, to determine if the addressee will accept messages from the originator. If the privacy check fails, the message is sent to the originator's user-control location with a service message indicating why the message was undeliverable.

The next operation performed by the program is to check the status of the addressee. If the addressee is on alternate delivery to another station, the disposition item in the terminator slot is marked to indicate the alternate delivery and a new terminator slot is formed for the alternate addressee. If the message to be alternate delivered has already been alternate delivered from another station, an illegal condition exists which may result in "ring-around-the-rosy." In this case, the message is sent to the terminator's user-control location. If the status of the addressee indicates that the station is in a hold

state or in a not-ready state, the entry register for this addressee is marked unavailable for delivery. This will cause the message-nomination program to skip over this entry when processing the queue for the given line.

After the station-status checks are made and it has been determined that the current terminator slot should be queued, the message-queue table is examined to determine whether or not a message queue exists for the terminator's queue index. If a message queue does not exist, then a queue-location register is seized and linked to the message-queue table. An entry register is then seized and linked to the appropriate subqueue through the queue-location register. The entry register is linked to the master register and to the terminator slot in the message-processing block using the relative slot address. The station identity consisting of the station's poll-table index is also inserted in the entry register. The queue counter in the second word of the message-queue table is incremented and, if the station status is such that immediate delivery can take place, the message-waiting bits in the line-status table are set to indicate the appropriate precedence, and the nomination map may be set.

After the entry register is linked in the message queue, the associated terminator slot is marked to indicate a queued disposition. The program then moves on to the processing of the next terminator slot. If the message-processing block should contain many addressees, the message-queue insertion program must segment its work and relinquish control so that other tasks may be performed. This is done by placing the processing register on a timing list. When the time-out occurs, the program will continue the processing until all terminator slots have been examined and a message-queue entry has been formed for each addressee, or until another time break is necessary.

6.6 *Summary of Input Processing*

Figure 16 summarizes the input-message processing sequence. The poll-administration program generated the station-polling codes and analyzed the station responses to determine the desire to originate traffic. On a positive response, the poll-administration program transferred control of the line to the message-reception control program which proceeded to seize and initialize the necessary temporary memory areas for the incoming message. The heading-generation program was used as a service routine to assemble a generated heading which typically included time, date, and message number. At the conclusion of the delivery of this information to the originating station, a
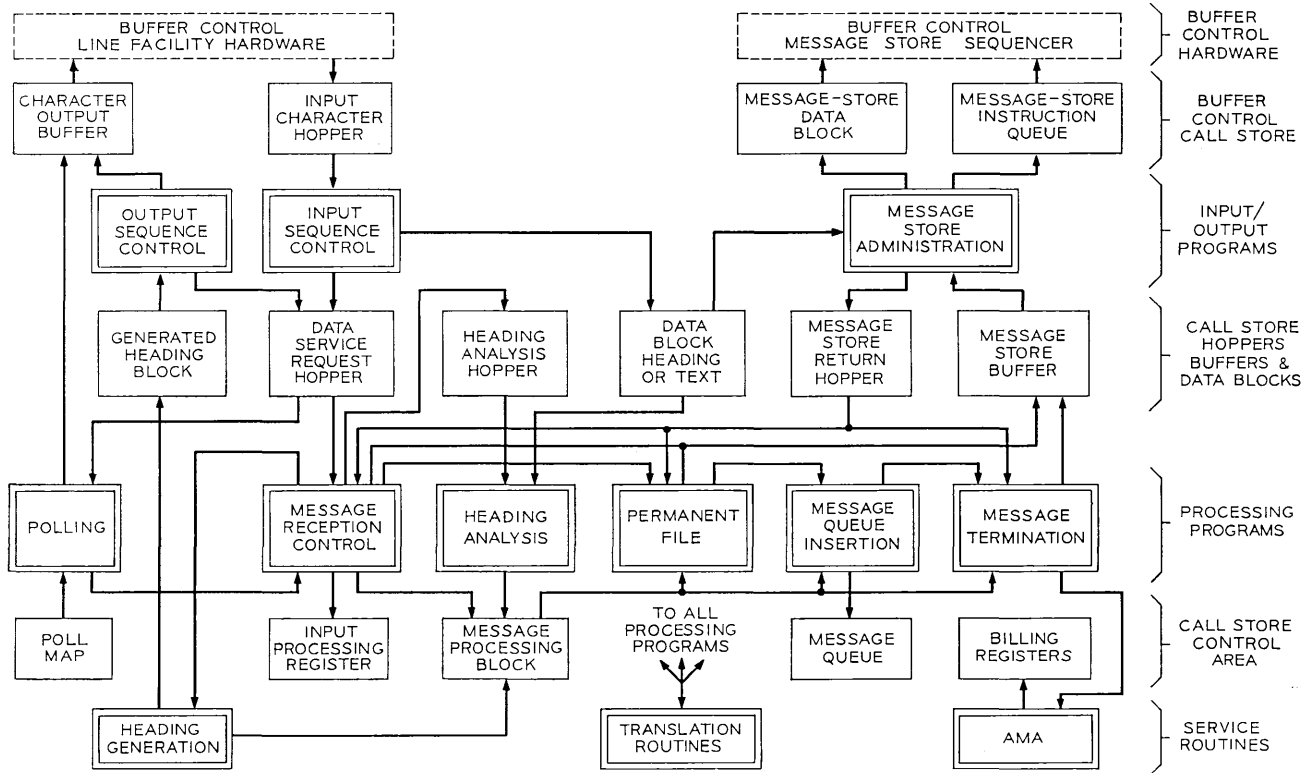
Fig. 16—Relationship between programs and memory areas concerned with the input-message processing.

sequence of control characters triggered the station to transmit its message to the switching center. The heading-analysis program was later entered when the complete heading information was received and stored on the disk. It proceeded to analyze the heading and build the message-processing block, establishing a terminator slot for each addressee in the heading. The message-reception control then guided the progress of the transmission until the end-of-message sequence was received. At that time, the message-reception control program divorced the message from the line, initiated further processing on the line, and relinquished further control of the message. The message-queue insertion program constructed a delivery request for each addressee listed in the message-processing block and added these requests to the appropriate message queues. When this task was completed, the message-termination program proceeded to update billing information and traffic-statistic counters, to transfer the call-store copy of the message-processing block to disk, and to release the call-store facilities.

6.7 *Message Nomination*

The message-nomination program is responsible for servicing the nomination map and initiating the transmissions to the terminating stations. This program is entered periodically by the executive control program and scans the nomination map. As mentioned earlier, a bit in this map, when set, corresponds to a data line that is available to receive traffic and in addition has messages waiting for delivery in its queue. The position of the bit in the nomination map uniquely defines the message queue by its corresponding queue index. With this unique identification, the linked list message queue is located through the message-queue table. The most eligible entry register is selected based on the priority and time ordered structure of the message queue. Entry registers which are marked as unavailable are passed over, i.e., the station intended for delivery is currently in a hold or a not-ready state. The poll-table index stored in the entry register uniquely identifies the station to which the message is to be delivered.

At this point the nomination program seizes an output-processing register which will follow the output processing through the remainder of the transmission. Using the system subroutines, output capacity to send the message is reserved and a data-block index is seized which reserves an output-work table entry for the transmission. The queue index and poll-table index are used as input data to the translation subroutines to retrieve the station related data which is loaded into

the output-processing register. As a result of the translation process, the DLN of the desired station is identified. The DLN is used to inspect the line-status table to see that the line is in a state which is capable of receiving traffic. At this point the program knows that a delivery may be made to the given line; the selected entry register is removed from the message queue and released to the idle-link list. The nomination program then scans the entry registers in the message queue to determine if the given message is destined for more than one station on the given line. If an entry register in the message queue contains the address of the master register associated with the message currently being processed, the program will initiate delivery of the message to the recipient stations simultaneously. To facilitate this, a processing register annex is seized for each additional station and linked to the output-processing register. The poll-table index stored in each entry register is used to retrieve the station translation data (station call-in code, heading-format number, and station options), which are stored in the processing-register annex.

Once the output-processing register and all associated processing-register annexes have been loaded with the station translation data and the line-state checks have been made, the message-processing block for the message must be located. The associated master register indicates the location of the message-processing block: (i) in call store, (ii) in message store, (iii) in transit from call store to message store, or (iv) in transit from message store to call store. The message-processing block must be in call store before processing may begin. If the message-processing block is in transit to message store, a call-store copy of it still exists and the nomination program requests the transfer to stop by making a disconnect entry to the program making the transfer through the processing register whose address is stored in the master register. If the message-processing block is in transit to call store, it is because some processing register had requested the transfer. The nomination program uses the general purpose call-register timing routines to time until the transfer is complete. When the message-processing-block state indicates that it is residing in message store and not in a transit state, the nomination program makes use of the message-store-administration routines to transfer it to call store.

When the message-processing block is finally located in call store, the count of stations waiting delivery is decremented by a count equal to the number of stations which will receive the delivery; i.e., the number of processing register annexes plus the output processing register.

In addition, an item in the message-processing block is incremented to indicate that another program is currently using the message-processing block. As noted previously, this item is used by the message-termination program in its determination of the final disposition of the message-processing block. The message-processing block is checked to see if the delivery is a service message rather than a data message. If so, the service-message program is entered as a service routine to generate the appropriate service message based on data in the message-processing block.

The next task in the output-processing sequence is the generation of the personalized header information to be delivered to each of the stations involved in the delivery. This is the function of the heading-generation program which is discussed in greater detail in Section 7.2. In summary, the heading-generation program seizes a data block and loads it with the appropriate time, date, and message number for each station. This information is also loaded into the message-processing block. If the user has elected the proper heading options, the heading-generation program retrieves the originator's heading from disk and inserts the relevant address information into the data block. In addition, the program prefixes this information with a call-in code and control codes so that each station receives only its own generated-heading information. The heading-generation program transfers back to the nomination program which initiates the delivery by inserting the data-block address in the data-block address table and the character-output buffer address into the output-work table. The line-status table is updated to reflect that an output transmission is about to take place on the line by calling a subroutine in the message-transmission control program. The address of the output-processing register is placed in the line-status table.

6.8 *Message Transmission*

The message-transmission control program is responsible for transmitting the message to the line and retrieving data blocks from the message store as required. The output-sequence-control program acts in a subordinate role to disassemble data blocks and transmit characters to the line. The process is initiated when the output-sequence-control program detects the load-next-word output-control code in the data block associated with the output-work table entry (see Fig. 17). The data machine call-in sequence in the following word of the block is transmitted to the line and the program temporarily ceases transmitting to the line while waiting for the station response. In addition, an

1	0			CONTROL CODE
0	0	FILL	ENQ	CEC
1	0	SKIP ADDRESS		
		STATION HEADING		
1	0			CONTROL CODE
0	0	FILL	ENQ	CEC
1	0	SKIP ADDRESS		
		STATION HEADING		
1	0			CONTROL CODE
		COMMON LINE HEADING		
1	0			CONTROL CODE
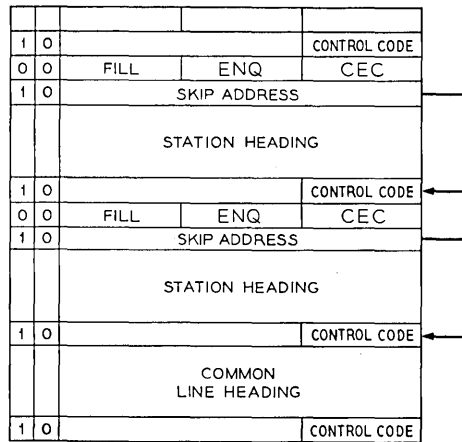
Fig. 17—Format of generated-heading data for a delivery to two stations on the same line.

entry is placed in the data-service-request hopper as a notification that the call-in sequence has been transmitted. When the data-service-request hopper is serviced, the message-transmission control program is informed of the transmission and accordingly updates the call status in the output-processing register and sets the response timer.

The data machine's response to the call-in sequence, its station identity code, is received by the input-sequence control program when it services the input-character hopper. The status bits in the word-status table direct the input-sequence-control program to load the response in the data-service-request hopper. When the data-service-request hopper is serviced, the response is analyzed by the message-transmission control program. If the response is not valid, the call-in sequence is repeated. If the response to this second attempt is also invalid, the message-transmission control program can use the skip address in the data block to skip over the station-heading information to transmit the call-in sequence for the next station on the line. In addition, the station-status table information is updated to reflect that the station is not ready, and a service message is transmitted to the transmission-plant craftsmen to indicate that the station has failed call-in. If the response to the call-in sequence is valid, the program updates the call state in the output-processing register. In addition, the program directs the output-sequence-control program to resume transmitting by advancing the data-block address in the data-block

address table to the first word of the station-heading information.

The output-sequence-control program proceeds to transmit the station-heading information which may consist of relevant address, time, date, and message number based on the station options. The process continues with the transmission of further call-in sequences and station headings until all stations to receive the message have received their, and only their, particular heading information. An output-control code in the data block directs the output-sequence-control program to make a data-service request entry. This entry informs the message-transmission-control program that the call-in sequence is complete. The program then directs the output-sequence-control program to transmit the line heading to all stations. The line heading is common information received by all stations, such as the originator's time, date, and message number based on line options. The first portion of this data consists of a control character sequence which turns on all stations that had previously received the call-in sequence. The heading data which was generated by the heading generation program is terminated by a control code which indicates whether the line is to receive the originator's heading or only the text of the message. The data-block address is retained in the data-block address table for transmission of the message to the line. The supplementary control code in the data-block address table is set to a no-op state until the first-message block can be retrieved from disk and a data-service-request hopper entry is made.

When this entry is serviced by the message-transmission control program, it consults the message-processing block to determine the disk address of the first block of the link list of message blocks stored on the message store. Using the message-store service routines, the program transfers the first heading block to the call-store block retained in the data-block address table. If the line is not to receive the originator's heading but only the text, the message-transmission control program uses a special message-store administration routine to progressively read the link addresses in the link list of message blocks stored on disk until the address of the first text block is retrieved. The first text block is then transferred to the data block retained in the data-block address table. In accomplishing this task, the message-transmission control program uses the forward-link addresses to retrieve the data blocks and uses the reverse-link addresses as a check that a valid block has been retrieved from disk. When the program is notified of the successful transfer of the requested block through the message store return hopper, transmission to the

line is initiated by updating the supplementary-control code in the data-block address table. This process continues until the complete message is transmitted to the line.

6.9 *End of Delivered Message*

The end of text control code in the last data block is detected by the output-sequence-control program. The data block is released and a data-service-request hopper entry is made. When the message-transmission control program receives this entry, it proceeds to confirm that each station which received the message is still functioning properly by transmitting a special character sequence, which is called roll-call, to each station. When these responses are received and validated, the program proceeds to divorce the message from the line and restore the line to the idle condition. The output-processing register address is removed from the line status table and the line status is set to the idle state. The output-work table slot and the output capacity are released. The message-transmission control program determines what program action should next take place on the line, based on the type of line and the poll-delivery criteria. Accordingly, the program will either set a bit in the polling map or the nomination map, thus completing the disconnect treatment on the line.

The remaining processing of the output message is similar to that for the input-message transmission. The permanent-file program is entered to update the station-retrieval tables stored on disk. The message-termination program is then called upon to update the billing information and the station and system traffic counters. If no other program is currently using the message-processing block and further deliveries of the message are required, the message-processing block is returned to disk. The call-store copy of the message-processing block and the output-processing register are released. If all deliveries of the message have been completed, the disk copy of the message-processing block is released and the generation of a permanent-file magnetic-tape record of the message, and a journal-file magnetic-tape record of the message transactions, are initiated. This is done by converting the message-processing block from a passive unit of memory to a processing register and entering this newly formed processing register on the permanent file hopper. The programs which perform the tape functions are discussed in an accompanying article in this issue.[2]

6.10 *Summary of Output Processing*

Figure 18 summarizes the sequencing of the programs involved in the output-message processing. The message-nomination program initiated the message delivery as a result of scanning the nomination map and brought the copy of the message-processing block into call store. The program then called upon the heading-generation program to construct the generated heading, consisting of the call-in sequence and the station heading for each station and the line heading. The response of each station to the call-in sequence was analyzed and the heading transmitted to each station. The message-transmission control program retrieved the message blocks and controlled the delivery until the end of the message. At this time, a special roll-call sequence was sent to each station to verify that the station had received the message.

After completion of the message delivery, the message-transmission control program restored the line to an idle state and initiated further processing on the line by setting either the poll map or the nomination map. The permanent file program was given control of the processing register to update the station-retrieval tables stored on disk. The message-termination program then ordered the appropriate billing logic and either returned the message processing block to disk or initiated the permanent file processing.

VII. SERVICE ROUTINES

7.1 *Translations*

The translation data base is located in the permanent magnet twistor program store. It contains all the necessary information to define the configuration of user's stations on lines, lines within a user network, and users' networks within a No. 1 ESS ADF office. Translation service routines are provided for use by the system programs to retrieve this information. The service routines eliminate the duplicate program code that would exist if each program requiring translation data retrieved it itself but, more importantly, they make the system programs insensitive to the organization of the data base. Each of the translation data structures discussed in Section 7.1.1 has associated with it at least one program subroutine which, when entered with the proper input data, will obtain from the data base the associated information. In many cases, more than one program subroutine is provided, some of which accept alternate forms of input information or derive output information from the data base for more than one translator.
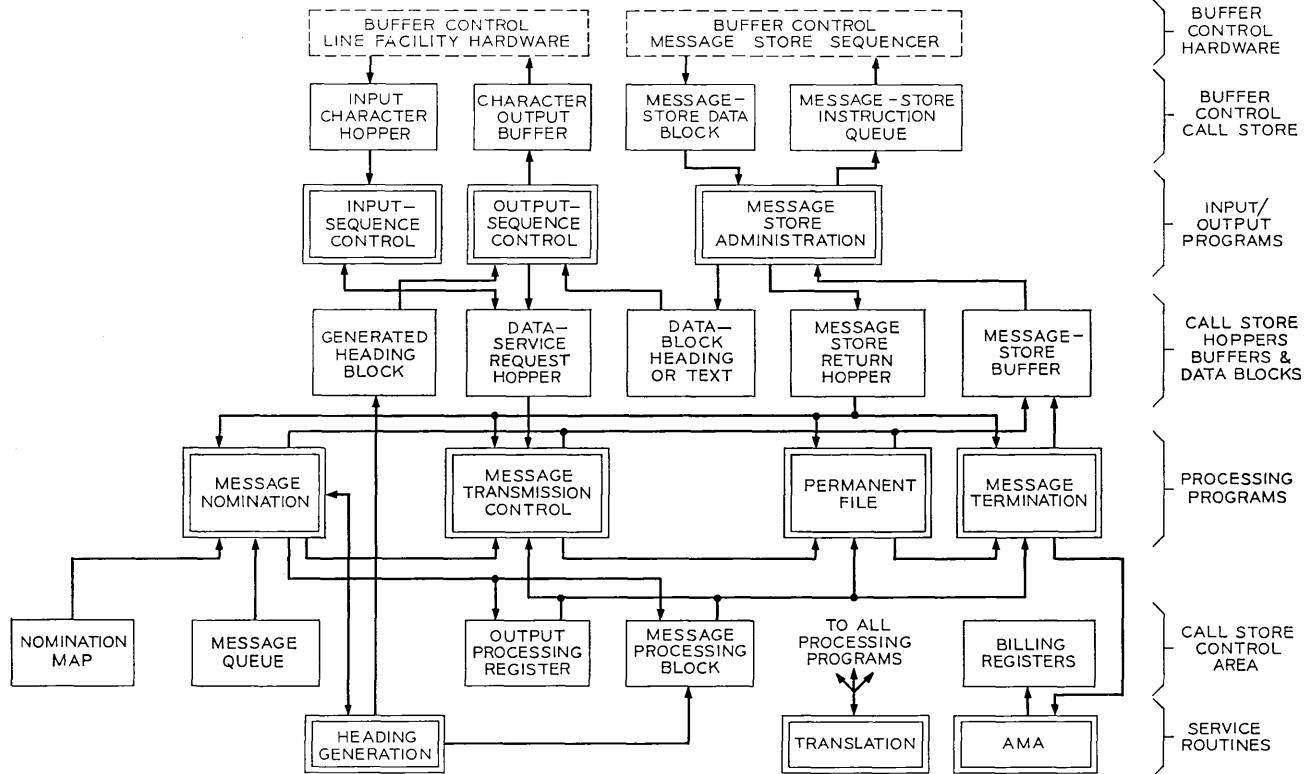
Fig. 18—Relationship between programs and memory areas concerned with the output-message processing.

Many of the basic concepts in the No. 1 ESS ADF translations are the same as those in No. 1 ESS:[4] the data base is in program store, changes are introduced via recent changes stored in call store, a card-writing process updates the program store, and after having been put into program store, the recent changes are removed from call store. Differences do exist, however, which are made possible and, in some cases necessary, by the differences in system organization and requirements. The relatively low-cost mass storage provided by the message store (disk) is used to provide a backup for the recent changes stored in call store; hence, errors introduced in the call store can be both detected and corrected.

The mnemonic translator is unique in that it utilizes a scatter technique to store its data. This technique makes the storage requirements and the recent-change program procedures for the mnemonic translator compatible with those of the other translators.

The following sections describe the contents and organization of the No. 1 ESS ADF translations, and the verification and audit procedures used to protect them.

7.1.1 The No. 1 ESS ADF Translators

7.1.1.1 *Basic Line Translation.* Each line in a No. 1 ESS ADF office is identified by a terminal equipment number known as a data-line number (DLN). This translator provides information on a per DLN basis. For a normally assigned line, the translator contains such information as a line class word, user-group identity, queue index, and a list of stations that are on the line.

The queue index can be used to locate the call store head cell for the queue of messages to be delivered to the line. The user identity is just that; it identifies the user network to which the line belongs. The line class word identifies such things as the speed of the line, the duplex nature (half or full) of the line, and the code (ASCII or Baudot) of the line. The list of stations on the line which this translator provides can be used to identify each of the stations on the line and to locate translators which identify the particular characteristics of each station.

7.1.1.2 *Basic Station Translation.* Information on each station in the system is contained in the basic station translator. The quantity used to locate a particular station within this translator is a station identity as obtained from the basic line translator. Each assigned station has the following information associated with it: normalized directory number, principal station mnemonic, alternate delivery directory number, station class word, call-enquiry code, heading-

format identity, and privacy list. The normalized directory number is the directory number of the station, normalized to take advantage of the restricted set of directory numbers that is possible within an office. The station mnemonic is a 1-to-7 character alphanumeric that is used by the system to identify the station in any messages that might be initiated by the system and delivered to the user. The alternate delivery directory number is the directory number to which, upon request, the system will alternate deliver all messages addressed to the station. The station class word contains such information as the originating and terminating heading format options for the station, the originator's heading forwarding option, the originator's precedence level insertion, and the action-request authority of the station.

Each station has associated with it particular characters which are unique to that station on the line; that is, they may be duplicated between stations on different lines but not between stations on the same line. These characters are used for control purposes in polling the station for traffic and calling the station in to deliver messages. The special characters for calling the station in to deliver messages (call-enquiry code) are contained in this translater.

Because a single ADF may serve the networks for more than one user, certain privacy facilities are provided. Each station has associated with it a privacy-screening code. This code may indicate: (*i*) that this station can receive messages only from other stations within its own user's network; or (*ii*) that this station can receive messages from any station regardless of its user affiliation; or (*iii*) that this station can receive messages from any station within its own user's network and, in addition, from any station within other selected users' networks; or (*iv*) that this station can receive messages from any station within its own user's network and, in addition, from any station identified in a list of directory numbers which is provided; or (*v*) that this station can receive messages from any station within its own user network and, in addition, from a combination of *iii* and *iv*.

A heading-format identifier is provided for each station. The identifier assigned to a particular station is dependent upon the code (ASCII or Baudot) of the station and upon the generated-heading format desired by the station. It is this identifier modified by options contained in the station class word that determines the format of the originating and terminating generated headings received by the station.

7.1.1.3 *Heading Format Translation.* This translator drives the heading-generation program (Section 7.2) in constructing the originating

and terminating generated headings delivered to stations upon the origination and receipt of messages. The particular heading-format table used is dependent upon the heading-format identifier associated with the station in the basic station translator, and upon whether the heading desired is for an originating message, and intraline terminating message, or an interline terminating message. Each table is a list of fixed data and subroutine calls. The order and contents of the information in this table determine the order and contents of the generated heading.

7.1.1.4 *Mnemonic Translation.* All messages addressed by users in the No. 1 ESS ADF system are addressed using 1-to-7 character alphanumeric mnemonics. This translator is used by the heading-analysis program to convert mnemonics in a message heading to the appropriate directory number or other information. A mnemonic may translate to:

(*i*) The directory number of a station.

(*ii*) A list of directory numbers. The message should be sent to all of these stations.

(*iii*) A DLN. This identifies a particular line and only has application in an action request.

(*iv*) A list of DLNs. This identifies a list of lines and only has application in an action request.

(*v*) Action-request indication. This identifies a message as one addressed to the No. 1 ESS ADF from a user's station. Such messages are known as action requests.

(*vi*) Message delivery precedence. This indicates that all deliveries, as a result of mnemonics in the message heading following this mnemonic, should be made with the precedence (one through four) to which this mnemonic translates.

Each user must define its own set of mnemonics; two or more users using the same mnemonic must each define it in the way that it is to be used by that user.

Defined as a special class of mnemonics in this translator are the mnemonics used to identify specific action requests. Each of these mnemonics translates to an index that is used by the program to determine the action to be performed. These mnemonics are defined only once, can be used by all users, and do not restrict the user's freedom to select mnemonics for his own network.

It is possible for a No. 1 ESS ADF office to have from one to 511 user networks. Every mnemonic requires three words of storage and

each user may have from a few to many thousand mnemonics defined in translations. The input to this translator (user identity plus mnemonic) is 51 bits. The special memory organization problems of the mnemonic translator are discussed in Section 7.1.2.

7.1.1.5 *Basic Directory Number Translation.* This translator is organized according to the normalized directory number. For each assigned directory number it provides the identity of a station and of the line on which the station resides.

This translator also provides a cross-reference file address. The cross-reference file is an area on the message store where a record of all originations and terminations is kept for message-retrieval purposes. Within the cross-reference file area, each station has a dedicated address which is provided by this translator.

7.1.1.6 *Basic Queue Translation.* This translator is organized according to the message-queue index. For each assigned message queue index it provides the associated DLN. Its purpose is to provide a means of translating between bits in matrices maintained for polling and message queuing and the associated DLN. In the case of multiline hunting groups, this translator provides a list of all of the DLNs associated with the hunt group.

7.1.1.7 *User Control Locations Translation.* Each user can assign certain stations within its network to be used for administrative control purposes. These stations are collectively known as the user's control location. This translator provides a list of the directory numbers associated with these stations.

7.1.1.8 *Report List Translation.* Each user segments his network into, at most, seven disjoint subsets of lines. It is possible for each of the user control location stations to maintain administrative control over one or more of these subsets. In addition, certain automatic traffic reports are generated by the system on a subset basis and are sent to these control stations. This translator provides a list of all of the data lines and multiline hunt groups in each subset for each user.

7.1.2 *Memory Organization*

Figure 19 illustrates the standard head table-subtable-auxiliary block structure of a translator. The input quantity is divided into two parts, a selector and a level. The selector is used to index the head
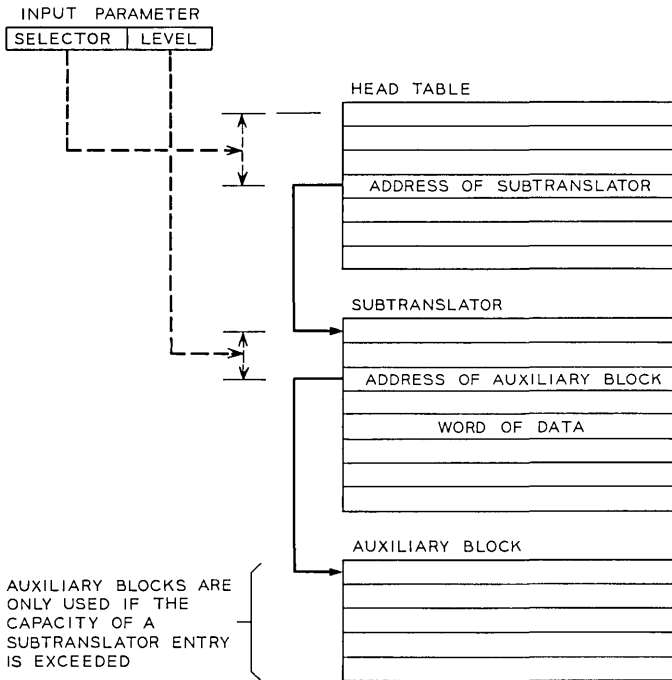
Fig. 19—Head table—subtranslator—auxiliary block organization of a translator.

table and the level to index the subtable. The subtable contains the required data or the address of an auxiliary block which contains the data. Every position in a subtable uniquely defines a specific input to the translator and, in a fully expanded translator, every input has a position in a subtranslator.

The mnemonic translator presents a problem in that its input information is a 51-bit quantity (9-bit user identity and 42 bits of ASCII alphanumeric characters). The 9-bit user identity may or may not be a densely packed set; clearly, the 42 bits of ASCII alphanumeric characters are very sparsely packed. A compression function is used to generate a 13-bit pseudorandom number from the 42 bits of alphanumeric characters. To this is added the 9-bit user identity. The 13 low-order bits of the sum are a 13-bit index used as input to a translator with the head table-subtable-auxiliary block organization (see Fig. 20).

The number of mnemonics that are defined in an office will usually exceed 8192 (2^{13}). Because of this, more than one 51-bit input will
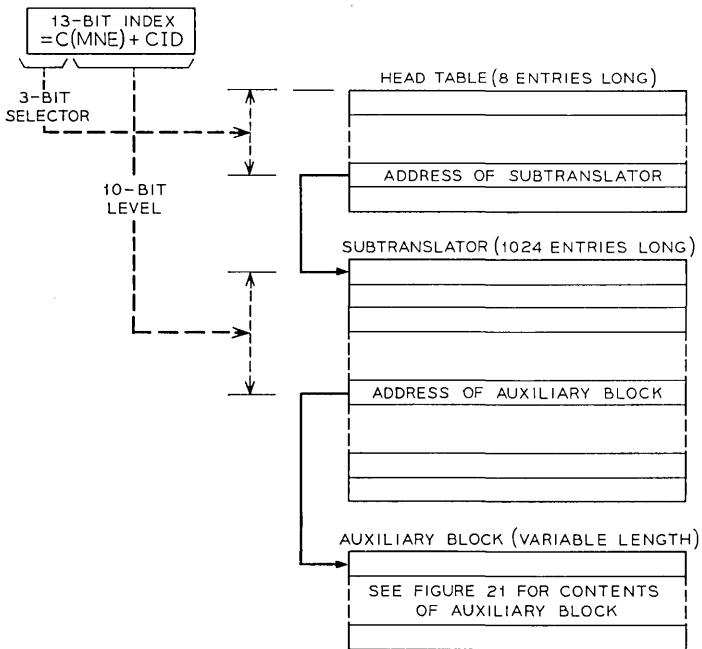
Fig. 20—Organization of mnemonic translator. C(MNE) = 13-bit pseudo random number generated by the compression function acting on 7 ASCII characters. CID = 9-bit user identity.

result in the same 13-bit index. These "collisions" are resolved by using an auxiliary block to store the data for all of the mnemonics with the same 13-bit index. Because the 13-bit index is not unique to a mnemonic, it is necessary to store the mnemonics in the auxiliary blocks along with their associated data (Fig. 21). Having found the proper auxiliary block by using the 13-bit index, the data is identified by doing a binary hunt over the mnemonics stored in the auxiliary block. By making the subtables longer than 2^9 (maximum number of users) it is not necessary to store the 9-bit user-group identity in the auxiliary block. This is so because the same mnemonic for two different users cannot fall in the same auxiliary block.

The mnemonic translator uses one eight-word head table and up to eight 1024-word subtables. In order to conserve translation program store space, the eight subtables are on the left side of program store*

---

* Words in program store have a 14-bit left-side portion and a 23-bit right-side portion. Most translation data is stored on the right side.

NUMBER OF WORDS IN AUXILIARY BLOCK	INFORMATION NEEDED TO DO BINARY HUNT		

DIRECTORY NUMBER OR OTHER DATA

$A_L(C_7)$	$A(C_1)$	$A(C_2)$	$A(C_3)$
$A_R(C_7)$	$A(C_4)$	$A(C_5)$	$A(C_6)$

ONE MNEMONIC ENTRY

Fig. 21—Auxiliary block for mnemonic translator.
$C_1C_2C_3C_4C_5C_6C_7$ = 7 character alphanumeric mnemonic.
$A(C_i)$ = 6-bit stripped ASCII representation of the alphanumeric character C_i
$A_L(C_i)$ = 3 high-order bits of $A(C_i)$
$A_R(C_i)$ = 3 low-order bits of $A(C_i)$.
Entries are ordered according to the value of the 21-bit number $A_L(C_7)A(C_1)A(C_2)$ $A(C_3)$. This allows a binary search to be used to locate mnemonics when there are many entries in the auxiliary block.

which would otherwise be "wasted." Since only one mnemonic translation is done per terminator, per message, the real-time penalty is not great. The head table and the auxiliary blocks for the mnemonic translator are on the right side of program store, as they are for most other translators.

The "scatter storage" organization for the mnemonic translator described above is no more costly in translation space than other schemes that were considered. Its primary feature is that, even considering imperfections in the compression function (unused values of the 13-bit index) the mnemonics are distributed over a large number of auxiliary blocks. The sizes of these auxiliary blocks are of the same magnitude as the sizes of auxiliary blocks for other translators. Hence, no special memory areas or special recent change techniques are necessary to administer the mnemonic translator.

7.1.3 *Recent Change Procedures*

A primary requirement is that the translation data base be readily changeable. This is accomplished by entering changes for the data base into call store via recent-change service orders, and requiring the translation service routines to hunt the call store for recent changes

to the data base before returning the requested information to their clients. No. 1 ESS ADF uses a field-oriented input format for recent changes from both the service order and maintenance teletypewriters. All recent changes are active immediately upon entry into the system. Certain translators that are infrequently changed are not checked for recent changes by their retrieval routine. In these cases, the new data will not be used until the program store is updated.

The card-writing procedure for updating the permanent magnet twistor cards in program store is basically the same as in the No. 1 ESS.[4]

7.1.4 *Translation Memory Protection*

Errors that are introduced into the call store recent-change area and that go undetected are eventually transcribed into the program store. It is possible for these errors to propagate and do extensive damage to the translation data base before being discovered. In an effort to guard against this problem, three audits as well as a backup for the recent-change call store are provided. The audits verify the format and order of the recent-change call store, cross-check the translation data for consistency, and verify that each word in the translation program store is used in one and only one translator. The backup for the recent-change call store is a copy on the message store of all permanent recent changes.

7.1.4.1 *Translations Audits.* The recent-change call store audit validates the format of the recent-change call store area. It checks counters and pointers used to administer the recent-change call store as well as checking the address order of the recent changes themselves.

As a part of the recent-change verification program, an audit is provided which cross-checks much of the operational translation data for consistency. This audit is initiated via a teletypewriter request and is driven from the mnemonic translation table. The translation data associated with all lines and stations which have mnemonics associated with them are cross-checked for consistency. While this is not a 100-percent consistency check of the translation area, it does check a large portion of the data. This verification will print out the inconsistencies and errors that it finds. Not all detected errors are by any means indicative of destruction in the translation area. In most cases, they are oversights on the part of the recent-change personnel and are correctable with additional recent changes.

Every station and line in the system is not required to have a

mnemonic associated with it in the mnemonic translator. It is desirable, however, if for no other reason than to enhance the capability of this audit.* Another audit, which is also a part of the translation verification program, can be called via a teletypewriter request to verify that all lines and stations do have a mnemonic in the mnemonic translator. It can be used to identify those lines or stations for which mnemonics are missing.

A basic sanity test for the translation program store is that each word is used once, and only once, and that all addresses in all head tables and subtranslators are within the translation program store range. This audit is a part of the translation-verification program and is called via a teletypewriter input message.

The program seizes auxiliary recent-change call store area to keep a busy-idle map for the address range of store being checked. Parameters are used to both define the address range of the translation program stores and the address ranges of subsets into which it is divided. A minimum of two subsets, defining the right and left sides of program store, is required. The busy-idle map is constructed for one subset at a time; hence, the subsets can be sized so that the busy-idle map will fit in the auxiliary recent-change area.

Errors found by this audit are printed out on the requesting teletypewriter. Each printout identifies the kind of error, the translator in which it was found, and the index into the translator when the error was detected.

7.1.4.2 *Message Store Backup for Recent Change Call Store.* In an effort to provide protection for the recent-change call store area and at the same time to take advantage of the storage media available in No. 1 ESS ADF, the message store (disk) is used as a backup for the recent-change primary and auxiliary call store areas. Input messages are provided which can be used from either the maintenance or the service-order teletypewriters to cause the permanent recent-change information in the recent-change call store to be copied onto the disk. Temporary recent changes are not copied onto the message store.

In response to an input from either the maintenance teletypewriter or from the service-order teletypewriter, the permanent recent-change information in the call store recent-change area is audited from the

---

* From the user's viewpoint, even stations and lines which never receive messages (send-only stations) should have mnemonics in order to use all the avaliable action requests.

information on the disk. The call store information is updated to agree with the information on the disk.

Two copies of the recent-change call store are maintained on the disk. When a request to copy the call store onto the disk is received, it is written onto the oldest copy. In this manner, should something happen to upset the copying process, the previous copy is not destroyed.

The recent-change information on the message store is protected by a check sum over 32-word segments. Should the check sum fail over a particular segment, that segment will not be used to change any call store information.

The procedure for inserting recent changes into the No. 1 ESS ADF requires that first, the recent-change call store be audited from the disk. The second step is to insert the recent changes into the call store. Thirdly, after appropriate verifications, the recent-change area should be copied onto the disk. At that point, the updated call store information is on the disk and, should the call store be destroyed, it can be reconstructed from the disk. Periodically (every hour) the recent-change call store is audited from the disk. The disk audit is also called automatically under certain conditions where, during the insertion of recent changes, the recent-change program discovers an impossible or unrecoverable situation. This allows the system to back up to the point where it was before the recent change was inserted. Protection against copying bad recent-change call store information onto the disk is provided by automatically changing any request for a copy into a request for an audit if there has been a maintenance interrupt since the last audit was run, or if any of the call store recent-change audits have discovered discrepancies.

7.2 *Heading Generation*

The heading generating service routines are used by the message reception control program upon the start of a message into No. 1 ESS ADF, and by the message nomination program when the delivery of a message to a station is initiated. These routines construct the generated heading by placing the appropriate characters and output control codes in data blocks in call store. The address of the first block in the linked list of blocks is returned to the calling program. That program proceeds to cause the generated heading to be sent to the station via the output-sequence-control program.

The format of an originating page copy is shown in Fig. 22. The originating message number, date and time are known as the generated heading. Figure 23 shows a typical terminating message. Its
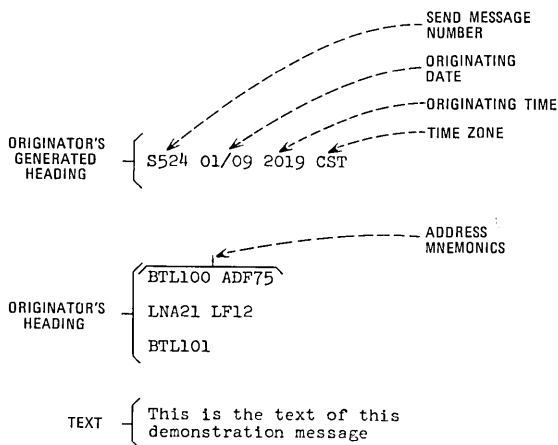
Fig. 22—Originator's page copy.

generated heading consists of the relevant address, the terminator's receive-message number, date and time, and the originator's send-message number, date and time. In addition, in the case of a termination, the generated heading contains those nonprinting character sequences necessary to call the station in for delivery.

There is great flexibility as to what a particular station is able to have in its generated headings. Optionally, a station need not receive all of the items shown in Figs. 22 and 23. Although it is not part of the generated heading, the originator's heading is an integral part of the heading options available to a terminating station. The terminating station is free to receive or not to receive the originator's heading, unless the originator specifically prohibits it via his translation option.

In addition to allowing a station the option of receiving those components of the generated heading shown in Figs. 22 and 23, and in allowing a station to choose whether or not to receive the originator's heading, it is possible for a station to specify additional fixed sequences of characters which are to be sent to it as a part of its originating or terminating generated heading. Also, it is possible for a station upon termination of a message to elect to receive a sequence of characters, if and only if, the address mnemonic with which the station was addressed was followed by a "+" symbol. A different sequence of characters can be sent to the station, if and only if, the
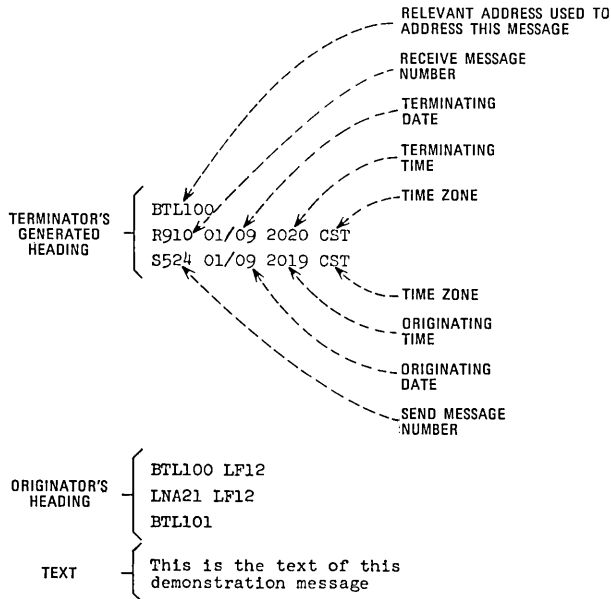
Fig. 23—Terminator's page copy.

address mnemonic with which the message was addressed was followed by a "*" symbol.†

In addition to optionally receiving the preceding components of the generated heading, it is possible for a station to receive them in nearly any order. The basic station translator contains a station class word, as well as a heading-format identifier. The station class word contains a series of bits to indicate the selection and nonselection of various items in the generated heading. The heading-format identifier is used to locate in the heading-format translator a data table that contains a list of fixed data and subroutine calls. The entries in the heading-format data table (HFDT), as well as their order, and the values of the bits in the station class word are used by the heading-generation program to determine the format for the generated heading.

The layout of entries in a HFDT is illustrated in Fig. 24. As an example, consider the HFDT shown in Fig. 25. This HFDT is for an ASCII HDX station which receives originator's message number, date and time, and time zone as its originating generated heading. By

---

† The "*" and "+" suffixes are used to cause sequences of characters to be sent to an addressee for the purpose of controlling auxiliary devices on the station equipment.

FORMAT
INDICATOR*

0 1	DATA		SI
0 0	C_1	C_2	C_3
1 0			OCC
1 1			SI

Fig. 24—Heading format data table layout.
 SI   Subroutine Identity—Control is transferred to the subroutine identified by
     this index.
 DATA   Certain subroutines need additional data such as the identity of a
     subformat table to use or the number of entries in the HFDT to be
     skipped based on a conditional test.
 C_1, C_2, C_3 Teletypewriter characters properly coded such that they can be sent
     to the station exactly as they appear in this table. C_1 is sent first,
     C_3, last.
 OCC   Output control code—a code used to give information to the output-
     sequence-control program.

 * The format indicators 01 and 11 are equivalent except that 11 indicates the
last entry in the HFDT.

changing entry number 4 in the HFDT in Fig. 25, the HFDT shown
in Fig. 26 results. This HFDT will result in the originator's send mes-
sage number only in the originating generated heading.

   Certain entries in the HFDT are required at fixed locations. For
example, entry number 1 in Fig. 25 must always be first and entries
6, 7, and 8 must always be in that order and be last in the HFDT.
This is because they specify necessary control characters for the sta-
tion and a control routine for the heading-generation program. Be-
tween these entries, however, any fixed data may be specified and any
subroutines which are appropriate may be called.

   HFDTs for terminating generated headings are in general longer
and more complex than for originating generated headings. They are,
however, constructed in exactly the same way as originating HFDTs.

   The heading-generation program is table driven from data in
translations that can be changed via recent changes. This, in effect,
allows the operating company the limited ability to program the ma-
chine in order to provide generated headings which contain, in almost
any order, the date and time of transmission, message number, the
relevant address on deliveries, and conditional and nonconditional
printing and nonprinting fixed character sequences. Up to 128 different
HFDTs can be defined for each of the three types: originating, intra-
line terminating, and interline terminating. Additional variations can
be obtained by the settings of the items in the station class word.

WORD
NO.

1	0 0	FILL	D C1	CARRIAGE RETURN	ASCII TTY CHARACTERS
2	0 0	LINE FEED	DELETE	S	ASCII TTY CHARACTERS
3	0 1	CODE TABLE INDEX = 0		SI = 4	SUBROUTINE : OBTAIN MN; ENTER IN MPB AND GH
4	0 1	SUBFORMAT TABLE INDEX = 0		SI = 1	SUBROUTINE : OBTAIN D/T; ENTER IN MPB AND GH
5	0 0	CARRIAGE RETURN	LINE FEED	DELETE	ASCII TTY CHARACTERS
6	0 0	FILL	START OF TEXT	FILL	ASCII TTY CHARACTERS
7	1 0			OCC = 32	OCC : START OF TEXT SENT
8	1 1			SI = 12	SUBROUTINE : CLEAN UP

THIS GENERATED
HEADING CONSISTS OF : ORIGINATOR'S MESSAGE NUMBER
                      ORIGINATING DATE , TIME , TIME ZONE
              EXAMPLE : S153 01/25 1150 CST

Fig. 25—Example of an originating HFDT for an ASCII half-duplex station. TTY, teletypewriter; MN, message number; MPB, message-processing block; GH, generated heading; and D/T, date and time.

VIII. ACTION REQUESTS AND SERVICE MESSAGES

An action request is a message from a station to the No. 1 ESS ADF. A service message is a message from the No. 1 ESS ADF to a station. As the name implies, an action request is used to request that the No. 1 ESS ADF perform some action. That action may be to collect and deliver certain data to a station, or it may be to alter the flow of messages through the network. Service messages are of two types. The first type is a message (queued service message) generated by the No. 1 ESS ADF in response to an action request, a reportable occurrence within the network, or according to a schedule. The second type (on-line service message) is a service message inserted into or appended onto a regular customer's message, either originating or terminating. Action requests and service messages are the tools which enable the telephone company (TELCO) to maintain administrative control over the entire network.

8.1 *Action Requests*

Figure 27 shows the action-request format. It should be noted that the entire action request is in the heading of the message. The heading-analysis program analyzes the mnemonics in the action request and

WORD
NO.

1	0 0	FILL	DC1	CARRIAGE RETURN
2	0 0	LINE FEED	DELETE	S
3	0 1	CODE TABLE INDEX = 0		SI = 4
4	0 1			SI = 0
5	0 0	CARRIAGE RETURN	LINE FEED	DELETE
6	0 0	FILL	START OF TEXT	FILL
7	1 0			OCC = 32
8	1 1			SI = 12

SEE FIGURE 25

SUBROUTINE: OBTAIN D/T;
ENTER IN MPB

SEE FIGURE 25

THIS GENERATED
HEADING CONSISTS OF : ORIGINATOR'S MESSAGE NUMBER
EXAMPLE : S153

Fig. 26—Example of an originating HFDT for an ASCII half-duplex station. D/T, data and time; MPB, message-processing block.

sets up a message-processing block to indicate the contents of the request. The various action-request programs then do further analysis and validity checking and perform the requested action.

Figure 28 is a partial list of the action requests that it is possible to input to the No. 1 ESS ADF. It should be noted that not all action requests can be submitted from all stations. Certain ones are reserved for TELCO use or for use by selected users' stations which maintain administrative control over the users' networks. These stations are known as the user-control locations.

The scope of the action requests is of particular note. Using them, it is possible to: stop message origination from a station (put the station on SKIP); stop message delivery to a station and hold the messages in a queue (put the station on HOLD); stop message delivery to a station and alternate deliver the message to some other station (put the station on ALT). The preceding action requests all affect the flow of messages through the network. In addition, it is possible to request data from the No. 1 ESS ADF. In particular, it is possible to request a report indicating the number of messages queued for a line and to request the status of a line to determine the stations that are on SKIP, HOLD, or ALT, or that are NOT-READY to receive (e.g., have low paper).
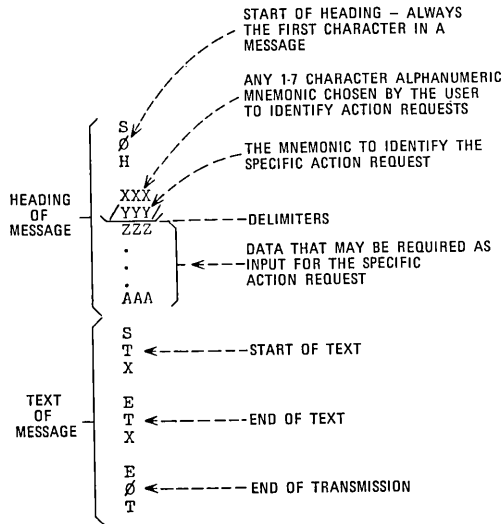
Fig. 27—Action-request format.

8.2 Queued Service Messages

8.2.1 Service Messages in Response to Action Requests

Most action requests receive a direct reply in the form of a service message. Figure 29 shows typical responses; it should be noted that all action requests which affect the ability of a station to either originate or terminate messages stimulate two service messages to be generated. The first message is a notification to the affected station. The second service message is a notification to the user's administrative control center (user-control location) that some action affecting the station's ability to originate or terminate messages has been taken. Both of these service messages are generated regardless of whether the action is requested by the station itself, by the user-control location, or by TELCO. This enables the user-control location to stay abreast of the latest status of every station for which it is responsible and, hence, administer the flow of messages through the network. Other action requests stimulate service messages which are sent either to the action-request originator or to some preassigned user-control location station, depending upon the type of action request.

All examples of service messages, except those in Fig. 30, show only the printing characters. Figure 30 shows an example of a service message

ACTION REQUEST ORDER	NECESSARY INPUT DATA	USED BY	MEANING
SKIP	STA ID(S)	UCL, TELCO, CSTC	STOP POLLING STATION FOR ORIGINATING TRAFFIC
OFFSKIP	STA ID(S)	UCL, TELCO, CSTC	SUSPEND SKIP CONDITION
HOLD	STA ID(S)	A	STOP DELIVERING MESSAGES TO STATION – RETAIN IN QUEUE
OFFHOLD	STA ID(S)	A	SUSPEND HOLD CONDITION
ALT	STA ID(S) ALTERNATE STA ID	A	REROUTE MESSAGES TO ALTERNATE STATION
OFFALT	STA ID(S)	A	SUSPEND ALT CONDITION
$\frac{Q}{Q\ LENGTH}$	LINE ID(S)	UCL TELCO, CSTC	SEND A SERVICE MESSAGE TO THE REQUESTING STATION INDICATING THE NUMBER OF MESSAGES IN THE QUEUE FOR EACH LINE
DIST	LINE ID	CSTC	MEASURE THE DISTORTION INCOMING ON THE LINE – REPORT TO REQUESTING STATION
STATUS	LINE ID	TELCO, CSTC	REPORT TO THE REQUESTING STATION THOSE STATIONS ON THE LINE THAT ARE ON SKIP, ON HOLD, ON ALT, OR IN A NOT READY CONDITION AND WHETHER OR NOT THE LINE IS BEING SERVED ON ITS NORMALLY ASSIGNED DATA LINE PORT

STA ID(S) – STATION IDENTITY(S) – NON TELEPHONE COMPANY USERS USE MNEMONICS, THE TELEPHONE COMPANY USES DIRECTORY NUMBERS

LINE ID(S) – LINE IDENTITY(S) – NON TELEPHONE COMPANY USERS USE MNEMONICS, THE TELEPHONE COMPANY USES DATA LINE NUMBERS

A   – ANY STATION

TELCO   – TELEPHONE COMPANY INTERCEPT CENTER, TRAFFIC DEPARTMENT

CSTC   – CONTROL SERVING TEST CENTER, PLANT DEPARTMENT

UCL   – USER-CONTROL LOCATION

NOTE: ONLY THE UCL, TELCO, AND CSTC CAN USE AN ACTION REQUEST TO PLACE A STATION, OTHER THAN THE REQUESTING STATION, ON HOLD OR ALT

Fig. 28—Partial list of action requests.

with some of its nonprinting characters. It should be noted that the sequence

$$\begin{matrix} E \\ S\ S \cdots\ 01\text{–}01 \cdots \\ C\ O \end{matrix}$$

and the sequence $\begin{matrix} E \\ S\ S \\ C\ I \end{matrix}$ bracket the service message. This is for conven-

ience of computer-type terminals; the $\begin{matrix} E \\ S\ S \\ C\ O \end{matrix}$ sequence allows the computer

```
ACTION REQUEST                              SERVICE MESSAGE(S)
ACT                 TO BTL100:                      TO USER-CONTROL LOCATION:
/SKIP/                  ...01-09...                     ...01-10...
BTL100                  THIS STATIØN ØN SKIP            PLACED ØN SKIP
                                                        BTL100

ACT                 TO BTL100:                      TO USER-CONTROL LOCATION.
/OFFSKIP/               ...01-11...                     ...01-12...
BTL100                  THIS STATIØN ØFF SKIP           PLACED ØFF SKIP
                                                        BTL100

ACT                 TO THE REQUESTING STATION:
/Q/                     ...01-17...
L621                    Q REPORT
L619                    LINE         Q
                        L621         4
                        L619         5

ACT                 TO THE REQUESTING STATION:
/STATUS/                STATUS 1602 00TTY00077-619 NORMAL
1602                    /CID/0002

                        ...01-14...
                        SKIP REPORT
                        BTL100

                        ...01-15...
                        HOLD REPORT
                        BTL100    ADF75

                        ...01-18...
                        NOT READY REPORT
                        LLIN2
```

Fig. 29—Typical service messages in response to action requests.

to recognize the beginning of a service message and to shift into an alternate mode where it may be properly handled; the number sequences allow easy identification; the $\begin{smallmatrix}E\\ S\ S\\ C\ I\end{smallmatrix}$ sequence terminates the service message.

8.2.2 Service Messages in Response to System-Recognized Occurrences

Typical service messages generated in response to system-recognized events are shown in Fig. 31. There are two types of these service messages. The first are the queue high-low reports. The queue-high service message notifies the user-control location that some predetermined threshold for the number of messages in the queue for a

```
           NONPRINTING
            ┌──┴──┐
             E
             S S
             C Ø ...01-01...          E
                                      S S
                 THIS STATIØN ØN HØLD  C I
                                      └──┬──┘
                                   NONPRINTING
```
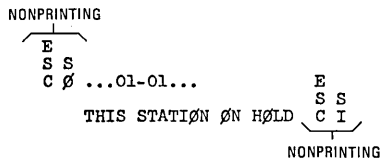
Fig. 30—Service message showing nonprinting characters.

QUEUE REPORT SERVICE MESSAGE TO THE
USER-CONTROL LOCATION:

    HIGH QUEUE REPORT:
        ...03-04...
        LINE Q HI TLCØRØL

    QUEUE OK REPORT: MNEMONIC TO IDENTIFY
        ...03-05...   THE LINE
        LINE Q ØK TLCØRØL

TROUBLE REPORTS TO THE CSTC:

    LOSS OF FACILITY:

        201 023 LOF 1799
                            DATA LINE NUMBER
                            LOSS OF FACILITY
                            REPORT NUMBER
                            OFFICE IDENTITY

    LOSS OF CONTROL:

        201 181 LOC 1564 04DD-03034-HM2
                            CIRCUIT NUMBER
                            DATA LINE NUMBER
                            LOSS OF CONTROL
                            REPORT NUMBER
                            OFFICE IDENTITY

    STATION FAILED POLLING:

        201 189 POL 1558 19T-TO3026-HS1 AM
                            POLLING CODE OF STATION
                            THAT FAILED
                            CIRCUIT NUMBER
                            DATA LINE NUMBER
                            POLLING FAILURE
                            REPORT NUMBER
                            OFFICE IDENTITY

Fig. 31—Service messages in response to system-recognized occurrences.

data line has been exceeded. Its purpose is to alert the user-control
location to some possible abnormal condition. A subsequent queue-
low report is sent to the user-control location when the queue for
the data line has returned to normal.

The second type of service messages in this category is the trouble
report sent to TELCO at the Control Serving Test Center. These
trouble reports notify TELCO of trouble conditions encountered in
polling a station, originating traffic, or in delivering traffic to a sta-
tion. They alert TELCO to problems with the user stations, many
times before the user himself is aware of it. This is an aid to fast
repair time with, of course, minimum down time for the user. It is an
especially important feature when many of the user stations may be
operating unattended.

8.2.3 *Periodic Service Messages*

As an aid in administering the network on a day-to-day basis and in collecting information to engineer the network, it is possible for both TELCO and for the user-control locations to receive scheduled service messages (see Fig. 32) from the No. 1 ESS ADF.

These service messages are of two types: status and traffic reports. The status report is generated and delivered according to some pre-selected hourly schedule to designated stations. Each station that receives a status report need not subscribe to the same hourly schedule. In addition, within certain restrictions, it is possible for each station that receives this report to receive information only concerning those stations in which it has an interest. Hence, it is possible for user-control location stations to receive periodic reports only on those parts of the user's network for which they have responsibility. The status report indicates all of those stations which are to be reported upon and which are on SKIP and hence cannot originate, or are on HOLD and hence cannot receive messages.

The second service message is the traffic report. As was the case

```
HOURLY STATUS REPORT
SERVICE MESSAGE:

    ...03-01...
    PERIODIC REPORT 02/02 1000 EST

    ...01-14...
    SKIP REPORT
    TRLC TRLB TRLF TRLE TRLD

    ...01-15...
    HOLD REPORT
    TRLB BTL100


DAILY TRAFFIC REPORT
SERVICE MESSAGE:

    ...03-03...
    DAILY TRAFFIC REPORT 02/02 1900 EST
    STATIONS          ORIGINATIONS              TERMINATIONS
                   MESSAGES    CHARACTERS     MESSAGES    CHARACTERS

    54-BA            20          24030           2          10590
    53-UA             3           3540           2           3880
    52-CA             2           2880          26          75030
    MHG-1             0              0           0              0
    TRLC             50          75030           0              0
    TRLB              1            320           0              0
    TRLF              1            490           1          11790
    TRLE             35          48010           5            380
    TRLD              0              0         120          55820

          TOTALS    112         154300         156         157490
```

Fig. 32—Scheduled service messages.

with the status reports, a traffic report may cover a subset of a user's set of stations. For each station included in the traffic report, counts of the number of messages originated and the number of messages terminated to the station and counts of the number of characters originated and the number of characters terminated to the station are given. The total counts are then included at the bottom of the report. The traffic report may be generated daily, monthly, or both. A monthly report may be generated on any day of the month that has been designated in translations. The daily report may be generated at any hour of the day that is specified in translations. If both reports are to be generated, the monthly report must be generated at the same hour of the day as the daily report.

8.3 *On-Line Service Messages*

These service messages are inserted into, or appended onto the end of the originating or terminating page copy on a user station. Their purpose is to alert the originator or the terminator of some peculiarity regarding either the originating or terminating transmission. Figures 33a and 33b illustrate typical service messages. Those service messages delivered on the originator's station interrupt the originating message. They indicate that for some reason the originating message may not be properly processed and delivered to all of the terminators.

Those service messages inserted into the beginning of a user's terminating message indicate some abnormality in the delivery, but do not nullify its validity. For example, the service message inserted at the beginning of a message delivery may indicate that the message was originally addressed to some other station, but has been alternate delivered to the station where it is now being delivered. Or, the service message may indicate that this is a possible duplicate message. That is, a previous attempt was made to deliver the message either to this or some other station.

Those service messages appended to a terminating message may or may not indicate that the delivered message is valid. For example, the service message may indicate that the delivered message should be disregarded. It may, however, indicate that the originator started this message into the system but never sent an end-of-text (ETX) character. Hence, the No. 1 ESS ADF does not know whether the originator transmitted the entire message. In this case, the delivery is followed by a possible incomplete service message. Figures 33a and 33b show examples of the way these service messages may appear on the terminator's or originator's page copy.

```
FORMAT ERROR:
                         ┌─────────────────────────ERROR
                         ↓
       BTL101 BTL100 ]ATT:  JØE DØE]

       ...04-04...
       IN FIELD 0003
       LAST MSG
       FORMAT ERROR RECEIVED
       PLEASE CHECK AND RESUBMIT


UNDEFINED ADDRESS:
                         ┌──────────────────────────ERROR
                         ↓
       BTL101 ABCD

       ...04-05...
       IN FIELD 0002
       ADDRESS ERRØR RECEIVED
       PLEASE CHECK AND RESUBMIT


MISSING CONTROL CODE:
                                                      NO END OF
                                                 ┌─ TEXT CONTROL
       BTL101                                     │  CODE
       THIS MESSAGE HAS NO ENDING CONTROL CODES. ↓


       ...04-01...
       INCØMPLETE MSG RECEIVED

                                (a)
```

Fig. 33a—Typical service messages inserted on-line on originator's page copy.


```
MISSING CONTROL CODE:

       BTL101
       THIS MESSAGE HAS NO ENDING CONTROL CODES.◄───ORIGINATING
                                                    MESSAGE LACKED
                                                    AN END OF TEXT
       ...06-03...                                  CONTROL CODE
       POSSIBLE INCOMPLETE MSG


DUPLICATE DELIVERY:

       ...05-04...PØSSIBLE DUPLICATE MSG◄───────THIS IS NOT THE
                                                FIRST ATTEMPT TO
                                                DELIVER THIS
                                                MESSAGE
       BTL100
       THE INITIAL DELIVERY ØF THIS MESSAGE WAS ABØRTED FØR
       THE PURPØSE ØF THIS DEMØNSTRATIØN.


ALTERNATE DELIVERED MESSAGE:

       ...05-01...ALT DEL FROM BTL100◄─────────MESSAGE WAS
                                                DELIVERED AT
                                                BTL101, THE
                                                ORIGINAL
       BTL100                                   TERMINATOR WAS
       THIS MESSAGE WAS ADDRESSED BTL100        BTL100
       BUT DELIVERED TO BTL101
                                (b)
```

Fig. 33b—Typical service messages inserted on-line on terminator's page copy.

8.4 *Service-Message Generation*

8.4.1 *Generation of Queued Service Messages*

All service messages, other than the on-line service messages, are delivered in a fashion as similar as possible to the manner in which regular messages are delivered. The program initiating a service message forms a message-processing block in which is indicated all of the terminators. This special message-processing block is then flagged as a service-message message-processing block and additional data needed for the generation of the appropriate service message is entered into it; in some cases, additional data may be placed on the message store. In these instances, the additional data in linked to the message-processing block in exactly the same fashion that the heading and text of a normal message are linked to it. The message-processing block is used to queue the service message for delivery in the same manner that any other message is queued for delivery. It should be noted that all of the data collection and generation necessary for the service message is done before the service message is queued for delivery. Also, all actions (e.g., placing a station on SKIP) are performed before the service message is queued.

The process of nominating a queued service message for delivery is the same as that for nominating any other message. Once the receiving station is called in, the message-transmission control program reads the first data block from the disk (if there is such data) and then enters the service-message generation program. This program, using data from the message-processing block and data which the message-transmission control program read from the disk (if there is such data), generates in call store a data block that contains the message that is to be transmitted. The service-message generation program returns to the message-transmission control program with this data block and the message-transmission control program uses it in exactly the same fashion that it would use a text block read from the message store for an ordinary message; the data block is sent to the station via output-sequence control.

After a complete data block has been transmitted to the station, the normal procedure is for the message-transmission control program to fetch the next data block from the message store. If this service message is one that has data blocks on message store, the procedure just described is followed. In any event, either immediately, or after having obtained the next data block from message store, the service-

message generation program is again entered and it generates in call store a data block that contains the next characters to be sent out to the terminating station. In effect, the service-message generation program just replaces or supplements the logic used to pull the next text block from the message store. The service-message generation program's function is primarily one of reforming or expanding data which may be in the message-processing block or in data blocks on the message store.

8.4.2 *Generation of On-Line Service Messages*

On-line service messages are generated when, in the message-originating process, or in the message-delivery process, some program finds an error or an abnormal situation which requires that the station be notified in order to maintain system integrity. The mechanism for generating these service messages is basically independent of whether it is to be sent to an originator or to a terminator. In either case, the cognizant program enters the service-message generation program indicating the kind of service message that should be generated. The service-message program generates the necessary characters in data blocks in call store for the service message to be sent to the station. In the case of a service message appended to the end of a message, the service message is sent to the station and that terminates the transmission. In the case of a service message inserted at the beginning of a regular message, the data blocks containing the service message are incorporated into the generated heading.

IX. MESSAGE INTEGRITY FEATURES

It was recognized early in the development of No. 1 ESS ADF that hardware and software problems would occur that could affect the content and delivery of the users' messages. A message-switching system is especially sensitive to the effects of processing errors since the user is not supervising all phases of the delivery of his messages. For this reason, it is not sufficient to provide audit programs which only protect the capability of the system to transmit messages. It is also necessary that audit programs assist in accounting for all the messages stored in the system. The audit programs could not perform this message protection without the provision of operational procedures and memory layouts which facilitate message accounting.

As noted in the above description of the ADF operational program, certain in-line processing steps have been taken and the disk-memory assignment has been organized to facilitate accounting for all

messages. The first step in the procedure is to store a preliminary copy of the first message-processing block on the disk at the time of the initial decision to start accepting the message into the system. At this point in the sequence of accepting a message, the first message-processing block contains the preliminary originator information consisting of the originator's identity, message number, and time and date. This is sufficient information with which to generate a service message indicating a system failure on the specific user's message. At a later time when the input message is complete, and further terminations are necessary but not in progress, the complete message-processing block is stored on the disk. In order to be able to distinguish between first message-processing blocks that are actively associated with messages and those that are idle, the blocks are written with an idle code when they are released. This step insures that the audit programs can recognize the existence of the proper message-queue stimulus for all those messages which have not been delivered to all terminators.

Several uses are made by the audits of the first message-processing block information as stored on the disk. The first use is the periodic search of the disk area in which the first message-processing blocks are stored. This is called the daily message audit, although it is performed twice daily. This audit checks each busy first message-processing block for a proper call-store stimulus. If a first message-processing block is marked preliminary, then there should be a call-store copy of the message-processing block. If it is marked complete, there should be a master register. If there is no call-store copy corresponding to the preliminary disk copy, then the program generates a service message to be sent to the TELCO position. This service message gives the information necessary to notify the originator of the specific message that may not have been delivered to all terminators by the system. These service messages are sent to the TELCO position primarily for two reasons. First, they are an indication of the grade of service being provided by the system. Second, this permits delivery, even if the user's station is in an out-of-service state. If there is no master register for a complete message-processing block, then the message-processing block is read into call store and new entries are made in the message queues for all undelivered terminators.

The second use of the first message-processing block information stored on the disk is for the message-recovery process after a severe system disturbance. The first message-processing block area on the disk is scanned for two purposes. One is to rebuild the busy-idle map

of block usage on the disk, and the other is to requeue messages that are complete on the disk and to send service messages for incomplete messages. In this case, the service messages are sent to the originator. The reason for directing the service message to the originator instead of to the TELCO position is twofold. First, it is assumed that the message was incomplete because the originating message was interrupted by the emergency-action process, and the user is served better by an immediate notification. Secondly, the TELCO position would otherwise be flooded with a large number of messages.

The typical method of clearing the problem that caused a severe emergency-action phase is to obtain a system which operates in a sane manner by clearing the system's history and using initialization or restart procedures. In order to provide message protection and a continuity to the processing, it is necessary to save information through the system-recovery process. When selecting information to be saved, it is necessary to weigh the factors of data value, the sensitivity of the system to errors in the data, and the ability to correct errors in the data to be saved. In the No. 1 ESS ADF system, some of the information selectively saved through call-store zeroing are the message-number tables and station-status tables. Both of these are necessary to provide continuity of processing from the user's viewpoint and the system sensitivity to errors in these tables is quite low. The effect of an error in a message number should be no worse for the user than a zeroing of the message number. The station-status tables can be checked for invalid or mutually exclusive states and placed in a valid state if necessary. Periodic reports will indicate to the user the status of his stations and if incorrect states are noticed, the user may put the stations back into the desired state.

The combination of audits and operational defensive checks as used in the No. 1 ESS ADF provide a message-switching system with a high degree of message integrity. Since cutover of the system, a number of lost-message complaints from users have been analyzed using the message and journal file retrieval capabilities. Consistently, it has been found that the message was properly delivered and lost by the station attendant or that a service message concerning the given message was disregarded by the station attendant.

X. CONCLUSION

The No. 1 ESS ADF is a modern store-and-forward data-switching system. It is an extension of the No. 1 ESS hardware and program technology to provide message-switching service to data users. This

paper has described in detail the operational programs and storage organization used to poll stations for originating traffic, to process the message transmissions to and from the stations, and to queue and store messages waiting to be delivered. Also, it described the unique characteristics of the program which provide for a high level of message integrity and provide the user with the ability to exercise administrative control over his network.

XI. ACKNOWLEDGMENT

REFERENCES

1. Harr, J. A., Hoover, Mrs. E. S., and Smith, R. B., "Organization of the No. 1 ESS Stored Program," B.S.T.J., *43*, No. 5, Part 1 (September 1964), pp. 1923–1959.
2. Ewin, J. C., and Giloth, P. K., "No. 1 ESS ADF: System Organization and Objectives," B.S.T.J., this issue, pp. 2733–2752.
3. Potter, J. L., Strebendt, Mrs. F. B., and Williams, J. R., "No. 1 ESS ADF: Magnetic Tape Subsystem," B.S.T.J., this issue, pp. 2915–2940.
4. Ulrich, W., and Vellenzer, Mrs. H. M., "Translations in the No. 1 Electronic Switching System," B.S.T.J., *43*, No. 5, Part 2 (September 1964), pp. 2533–2573.

No. 1 ESS ADF:

Maintenance Plan

By E. J. AITCHESON and R. F. COOK

The No. 1 ESS ADF message switching system provides a store and forward data service which places special demands on system dependability and maintainability. This paper discusses the hardware and software features used to detect and sectionalize troubles, as well as the recovery techniques used to restore service quickly. Maintenance of the line facilities, use of circuit redundancy, and message data protection are also included.

I. INTRODUCTION

A communication switching system must be designed with dependability and maintainability as an integral part of the overall plan. The No. 1 ESS ADF store and forward message switching system is no exception. Continuous high quality service is of vital importance. The characteristics of high quality data service include good error performance, 24-hour service with a minimum of interruptions, fast restoral of service, and no loss of messages when interruptions do occur.

The system's error performance objective for basic station-to-station messages is: on the average, no more than one error in 10^5 bits— 99 percent of the time while continually transmitting. The error performance will be determined largely by the station access lines since the error rates within the switching office are much lower. The switching center hardware was designed to include optional error detection and correction features (by retransmission) to achieve even greater transmission accuracy.

The reliability objective for the No. 1 ESS ADF system is to provide continuous service with system downtime not exceeding 2 hours in 40 years. The store and forward data features make it possible to preserve message information under the most severe fault conditions so messages can be retransmitted or retrieved when service resumes. Once a store and forward message office accepts incoming traffic for

delivery at a later time, it is of utmost importance that the message and delivery stimulus are never lost.

The No. 1 ESS ADF maintainability objectives provide a system whereby most faults can be located automatically and repaired quickly with minimum effect on service.

II. GENERAL MAINTENANCE PLAN

In the No. 1 ESS ADF system, all message data is routed through common processing units. The transmitted teletypewriter data from user stations is converted into computer words by an autonomous data scanner-distributor (DSD) and the autonomous buffer control. The message is assembled into information blocks in call store memory, buffered for delivery in a disk memory, and permanently stored on magnetic tape for retrieval purposes. The consequences of a failure in these common units, through which all messages may pass, can be severe. Fast recovery from failures is vital, as interruptions can affect all messages in the process of being transmitted or received. For example, if buffer control I/O processing is interrupted for longer than 66 milliseconds, input messages from all 150 baud stations must be retransmitted. To avoid complete system failure when a single component fails, circuit redundancy is used. With circuit redundancy, service can be maintained during fault diagnosis, fault repair, and routine maintenance.

The maintenance goal is to recover from faults before service is appreciably affected so that the user is unaware of trouble. To accomplish this goal, errors and faults must be detected quickly before incorrect information propagates into other units in the system. Continuous hardware checks provide the principal means for detecting faults in the common processing units. When a hardware check fails, an interrupt sequencer in the central control transfers program control to maintenance fault recognition programs. These programs isolate the faulty unit and switch a duplicate unit into service. The standby duplicated units are normally run in synchronism with the active unit to keep the contents of standby units up to date, thus making them instantly available for use when a faulty unit is removed from service. For many faults, the trouble detection and reconfiguration process is sufficiently fast to avoid service interruptions from a user's viewpoint.

When fault recognition programs experience difficulty in restoring service, error analysis routines are used. Error analysis programs record a history of system interrupts, troubles, and configurations. These

programs are used in conjunction with fault recognition routines to isolate units with marginal faults or with faults that are difficult to locate. The error analysis programs, which use a statistical approach to fault isolation, can be considered as a backup to assist in recovering the system.

Interruptions in service may occur for some faults that are difficult to locate. In these cases, the customer automatically receives service messages that will assist in determining the corrective action to be taken. Interrupted input messages must be resubmitted for delivery to the office by the user. Interrupted output messages will be retransmitted to the station automatically under program control.

After call processing has resumed, diagnostic programs are scheduled to be run on the unit removed from service. The purpose of these programs is to test the unit thoroughly and to supply test results to the maintenance craftsman. Maintenance trouble locating manuals translate the test results and list the circuit packs that might be faulty.

The system also includes fault detection capability for facilities dedicated to a user's line. Automatic in-service performance checks executed by the system are used to test both active and idle lines. Troubles that degrade user service can be detected and corrected before they become catastrophic; for example, parity over each character in the message checks terminal circuits and the quality of the transmission facility. Failure of the station to respond correctly to polling signals sent by the switching center can initiate corrective action for idle lines. When line faults or marginal station troubles are detected, the system is not interrupted. A control serving test center is notified of the problem by a teletypewriter message, where the necessary action is taken to sectionalize and clear the trouble.

The user is also provided with service features, which can be used when difficulties are encountered. For example, the user can request retransmission or retrieval of messages that were received with errors. Alternate terminals can be specified to receive messages addressed to a faulty terminal. Traffic statistics can be requested periodically that include the number of messages delivered to and transmitted by each station.

The dependability of the system is enhanced by the use of conservative circuit designs and long-life components. Wherever possible, tried and proven No. 1 ESS units, packs, and components are used. The same design principles, using liberal operating margins, worst-case circuit designs, long-life silicon and magnetic devices that have

proved effective in past projects are applied to this message switching system to obtain reliable units and a low trouble rate.

The principal features of the maintenance plan are as follows:

(*i*) Conservative circuit designs and long-life components are used to obtain reliable units.

(*ii*) Redundant units are used to provide service in the presence of failures and for routine preventive maintenance.

(*iii*) Rapid detection of faults by continuous hardware and software checks.

(*iv*) Recovery procedures by fault recognition programs are designed to preserve message information while testing and configuring the system around faulty units.

(*v*) Error analysis programs are used to distinguish between occasional errors and marginal or intermittent faults.

(*vi*) Diagnostic programs, interleaved with message processing programs, are automatically scheduled to isolate faults to replaceable plug-in circuit packages.

(*vii*) In-service checks of user lines provide rapid detection of faults and marginal troubles.

The following sections describe the redundancy plan, maintenance circuits, and maintenance programs. Those maintenance features for the central processor and other No. 1 ESS units are covered in the No. 1 Electronic Switching System described in the September, 1964, issue of the B.S.T.J.

III. CIRCUIT REDUNDANCY

The ADF system consists of a No. 1 ESS central processor and a community of ADF units to perform the store and forward message switching functions (Fig. 1). These units include an autonomous data scanner-distributor[1] to access the lines, a message store[2] (disk store) to assemble and hold messages awaiting delivery, a magnetic tape store[3] to provide a permanent file for messages, a buffer store[4] for scratch pad use, and a buffer control to perform repetitive tasks related to disk, tape, and I/O operations. Operational programs load commands and data for the buffer control into dedicated task dispenser queues in the buffer store. The queues are unloaded by independent wired logic sequencers in the buffer control which interpret the commands and perform the data transfers.

As shown on Fig. 1, the buffer control, buffer stores, message stores,
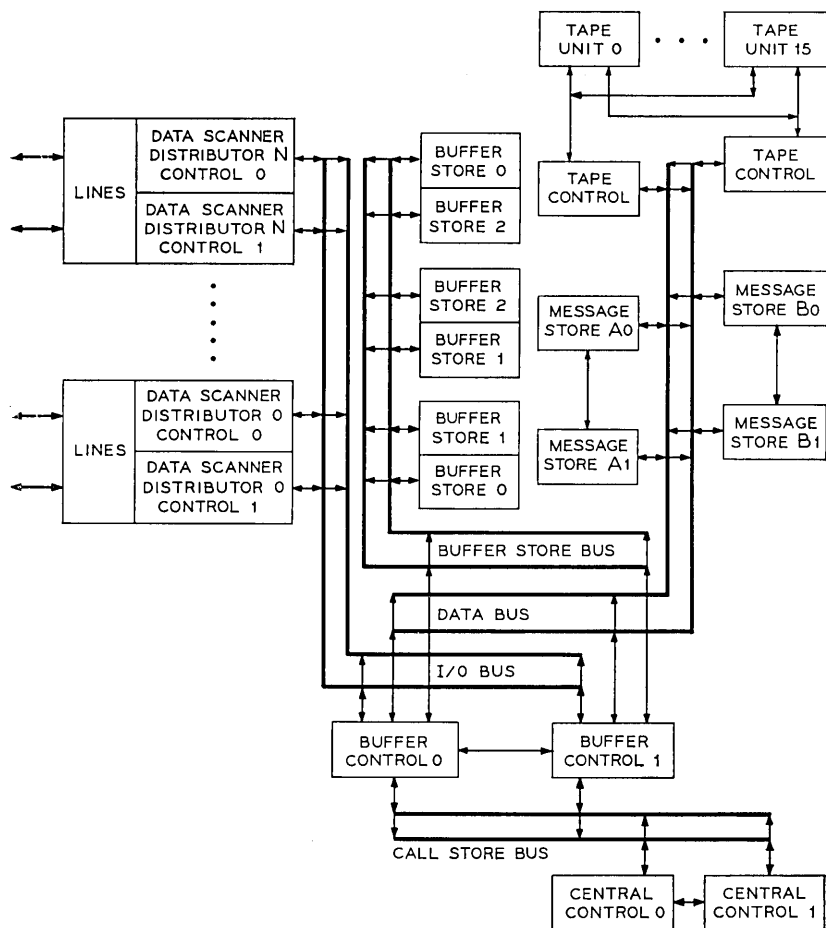
Fig. 1—Duplication of message processing units.

and communication buses are duplicated and the units are operated in a synchronous matching mode.

The autonomous data scanner-distributor units are partially duplicated. These units are used to convert input message characters, that arrive as a serial bit stream, into characters which are sent to the buffer control in parallel word form over the I/O bus. For output messages, the autonomous data scanner-distributor units receive characters from the buffer control in parallel, convert these characters into a serial bit stream, and route the data to the designated output line

terminal. The line terminal logic in the autonomous data scanner-distributor is not duplicated since faults in this logic will affect, at most, only 8 lines. The remaining logic in the autonomous data scanner-distributor unit, which uses time division techniques to perform the serial-to-parallel conversion and buffer the data, is duplicated.

Two tape unit controls operate in a simplex mode to provide simultaneous, but independent, tape storage operations. Under fault conditions, this redundancy allows messages to be put on a permanent tape file while deferrable tasks, such as message retrieval, are postponed. Each tape unit control can access a maximum of 16 tape units that provide sufficient spares for normal tape mounting and demounting operations, as well as for routine maintenance.

All buffer control data communication buses are fully duplicated. Each unit can be configured to receive data from either bus or send data on either or both buses by means of route control flip-flops. Normally each unit is configured to send and receive data on the same bus. Half of the duplicated units or controllers send and receive data on bus 0, while the other half use bus 1. When one unit is removed from service for maintenance purposes, the routing for the other unit is configured to retain as much of the duplicated system as possible. If the remaining unit sends data on both buses, then the buffer control can continue to access and match all other duplicated units on the same bus system in a normal manner. Because the tape unit control is not duplicated, it receives data on one bus and sends data on both buses to the buffer controls.

IV. BUFFER CONTROL COMMUNITY MAINTENANCE

The buffer control coordinates the transmission of data between all ADF units and verifies that these units and buses are functioning correctly. A malfunction in an ADF unit may be discovered by buffer control through several sources, which include parity failures during a bus transmission, status reports from the units, match failures at the buffer control, or by the failure of a unit to send buffer control an all-seems-well (ASW) response. The buffer control may react to these malfunctions by repeating the failed operation, incrementing error counters, reporting the trouble to operational programs via software queues, by interrupting normal processing with a maintenance interrupt, or by a combination of the above actions. The circuit features used to detect and report troubles in the ADF units are covered in the following sections.

4.1 *Interrupts*

When troubles are detected in the system, a wired sequencer in the central control interrupts the program in progress and transfers to a maintenance program that determines the source of the interrupt and takes corrective action. It is possible that more than one unit may detect and report a fault at the same time. To handle this problem, the trouble sources are grouped into interrupt levels and ranked according to the seriousness of the trouble source. Interrupt levels A through E are caused by central processor related faults or are manually induced.[5]

The ADF equipment, consisting of the buffer control and its peripheral communities, can generate F-level maintenance interrupts when malfunctions are detected. The interrupt will always be issued by the active buffer control, which is the only ADF unit that can interrupt the central control directly. An ADF peripheral unit can cause a maintenance interrupt only by inhibiting its all-seems-well signal to buffer control. This, in turn, will cause the buffer control to issue the maintenance interrupt to central control with only a single functional sequencer stopped.

If the central control receives an F-level maintenance interrupt, it will transfer program control to the F-level filter program. If the F-level source is the buffer control, the filter program will interrogate the buffer control error indicators to determine which buffer control or peripheral community is at fault. Failures of the central control peripheral units are also sources of F-level interrupts. Once the source is determined, the filter threads together the fault recognition programs to be executed to isolate and configure around the faulty unit.

Software checks of buffer control operations can detect errors and transfer control to maintenance routines. Since the central control and buffer control communicate with one another by software task dispensers, the main program can detect functional troubles by monitoring the progress and status of these queues. When buffer control completes a task, it overwrites the command in the queue with a passing or failing status report. If an operational program detects that the queue contains incorrect status reports, it can enter J-level fault recognition routines to test associated hardware. The fault recognition routine reissues the command on a half directed basis. In this manner, the faulty unit, not able to process the command correctly, is isolated.

4.2 *Fault Detection Maintenance Features*

4.2.1 *All-Seems-Well*

Each time the buffer control addresses an ADF peripheral unit, a 1-bit ASW signal from the unit is expected. The ASW signal indicates that the maintenance checks performed on the bus instruction have passed and the unit is functioning correctly. The ADF unit informs the buffer control that an error has been detected by inhibiting its ASW signal.

Table I summarizes the maintenance checks performed by the ADF peripheral units causing ASW failures. The disk and tape unit control perform other checks not shown in Table I that are reported through the use of an instruction queue as described in Section 4.1.

The buffer control reacts to the ASW failure as follows:

(*i*) The buffer control will first repeat the instruction and cause an F-level interrupt only if the repeat fails. On the other hand, if the ASW fails on a class of instructions referred to as central control read instructions, then the ASW failure is passed on to the central control by inhibiting the ASW on the call store bus. For this case, the central control has the responsibility of repeating or interrupting the system.

(*ii*) The ASW failure also selectively stops the logic sequencer in the buffer control responsible for the bus instruction that failed. By doing this, the state of the logic is frozen, thereby preventing the fault from

TABLE I—MAINTENANCE ALL-SEEMS-WELL CHECKS

Message Store
  1. Parity on instructions received from buffer control.
  2. Synchronization of duplicated disks (servo check).
  3. Internal clock check.
  4. Maintenance order received during normal operation (mode check).
  5. Buffer control out of sync with disk (sector match check).

Tape Unit Control
  1. Parity on data received from buffer control.

Autonomous Data Scanner-Distributor
  1. Match of input data from lines.
  2. Match of output data to lines.
  3. Match of data sent to buffer control.
  4. Match of time slot address.
  5. Parity from input line to bus access.
  6. Parity from bus access to output line.
  7. Parity over bus address and data.
  8. Address translator.
  9. Buffer control fails to acknowledge data sent by the DSD.
  10. Unit name decoder for bus instruction.

propagating. Other sequencers in the buffer control are allowed to continue normal processing until the fault recognition routines enter to test the buffer control.

4.2.2 *Parity Protection*

Data transmitted between the ADF units is protected by parity bits. The ADF peripheral units check parity on instructions received from the buffer control and will inhibit the ASW if the parity check fails. The buffer control checks parity on all data received from the peripheral units with one exception: central control may read from memory locations or registers in any ADF unit. The data from these central control read instructions is passed through buffer control, and central control is responsible for checking parity and reacting to parity failures. Most parity failures are treated similar to ASW failures. However, a special block repeat procedure is used for parity failures on instructions which read data from disk. A parity failure on disk reads is recorded for later use and buffer control finishes reading the block of data from disk. The buffer control sets a repeat flag in the queue status word and the buffer control rereads the entire block at a later time. If the block repeat fails, a program which administers the disk instruction queue calls in a fault recognition maintenance program. F-level interrupts do not occur for this type of failure.

4.2.3 *Error Rate Check*

Each bus sequencer is designed to automatically retry an operation if a parity or ASW failure is detected on the first attempt. A match interrupt is inhibited by an ASW or parity failure on the first attempt. Each time a bus error (failure on first try, success on second try) is encountered, an error counter dedicated to the bus is advanced by 1. When a count of 32 is reached, an overflow bit is set. This bit is periodically scanned and the error counter reset by the central control under command of a maintenance program. If the overflow bit is set as a result of an error rate in excess of a predetermined software threshold, the bus sequencer is forced to stop on the first failure and generate an F-level interrupt. In this way, the unit causing a high single error rate can be identified and removed for diagnosis.

4.2.4 *Matching*

The buffer control community and its buses are fully duplicated and run in a synchronous matching mode. All external bus operations are

matched, bit by bit, using hardware matchers. The information sent on the buses includes an address field, used to access a specified register within a unit, and a data field. The active buffer control matches address and the standby matches data. A mismatch in either buffer control will cause the bus sequencer in both buffer controls, handling the operation, to stop and freeze the bus priority F/F associated with the sequencer using the bus during that cycle. An F-level maintenance interrupt is then sent to both central controls by the active buffer control. Operations by other sequencers not requiring the stopped bus are unaffected. Normally, match failures cause an immediate interrupt. However, if parity or ASW failures also occur at the same time, then the instruction may be repeated as described in Sections 4.2.1 and 4.2.2.

A directed or off-normal match mode is provided where the circuits to be matched and the time the match is to take place are specified by program. This mode is used by the buffer control diagnostic program.

4.2.5 Internal Sequencer Check

All internal wired logic sequencers in buffer control are designed to advance through a wired series of sequencer states and, upon completion, recycle to a starting point. These sequencers receive their external stimulus from associated peripheral controllers in the form of service requests. Internal stimulus is provided by permission to use an internal or external bus. The response to this stimulus is controlled by the hard-wired sequencer logic. Since at each point the sequencer knows what to expect next, wired checks are made to verify the sequencer operation. Thus, invalid or out-of-sequence service requests (external stimulus) can be detected by the buffer control. They will cause that sequencer to stop and generate an F-level maintenance interrupt. All sequencer faults not detected in the above manner will be detected when that sequencer attempts to perform an external bus operation. If the duplicated sequencers are out of step, a data and address mismatch will result when one sequencer attempts to use the bus and its mate does not. Should the fault occur in an area common to the buffer controls, such as the service request decoders, the external peripheral equipment being addressed will inhibit its ASW response because it receives an out-of-sequence order.

4.2.6 Clock Checks

The buffer control clock is a 22-phase ring feedback chain driven by a 2-MHz source provided by the active central control. The phase

relationship of the ring is synchronized to that of the central control every 5.5 μs using a sync pulse generated by the active central control. Checks are made to verify that the 2-MHz clock is present and that the clock's phases are generated correctly. A clock fault stops all sequencers in both buffer controls and generates an immediate F-level interrupt. Error indicators related to clock circuits are accessed by scan points external to the buffer control circuits so the fault recognition and diagnostic programs can isolate the faulty unit without requiring an internal buffer control bus read.

V. MAINTENANCE PROGRAMS

5.1 *Fault Recognition and Recovery*

Fault recognition programs are called in when errors or faults are reported by the interrupt logic, base level, or low priority nondeferrable programs.[5] The purpose of the fault recognition programs is to determine the source of the trouble, remove the faulty unit from service, and restore the system message processing capability as quickly as possible. These programs also distinguish between errors and faults and may take no action other than recording that an error was detected. After corrective action has been taken, recovery routines initialize hardware and return to normal processing as gracefully as possible. In many cases, processing resumes at the point where the interrupt occurred. The fault recognition and recovery process emphasizes fast recovery to avoid destroying message information. In addition, special procedures are used to insure that messages on disk are not destroyed when severe problems are encountered.

5.1.1 *Disk Recovery and Message Protection*

There are two duplicated disk communities, each capable of storing 57 million bits of binary information. A portion of this data provides a present and past history record for the entire system and must be maintained over long periods of time. If one disk gets out of date and its mate experiences a failure, the system loses its ability to retrieve from that community. An out of date disk is never automatically configured into the system. Two separate recovery strategies are provided for this situation. The first requires a manual emergency action phase 5, which will clear all past history (time 0 start) and bootstraps the switcher into a workable configuration. Because the No. 1 ESS ADF is a store-and-forward system, hundreds of messages awaiting delivery would be permanently lost and no notification could

be sent to the sender to retransmit undelivered traffic. To avoid this gross loss of traffic, a second recovery plan is provided. When the duplex disk failure is encountered, all operational processing is halted, and notification of a duplex disk failure is given at the maintenance control center. Office personnel can then examine maintenance tele-typewriter printouts and decide which disk file had the last active copy of system records. The maintenance craftsman then protects the file from being overwritten by simply retracting the read/write data heads away from the memory surface. When the disk controller trouble is cleared, the protected disk is bootstrapped into the active system by a manual emergency action phase 4 restart initiated at the master control center. A phase 4 restart bootstraps the equipment and initial-izes the call stores and buffer store communities. Disk records are assumed to be accurate. All traffic being held for delivery at the time of the failure is then delivered in a normal fashion.

5.1.2 *Buffer Control Recovery*

The buffer control contains a number of sequencers that must be initialized before the buffer control can be restored to service. For example, the disk sequencers in buffer control must be synchronized with the disk and always know the address (sector) positioned under the reading and writing heads. The buffer control uses three types of service request signals from the disk to aid in the communication between these two units. The disk is divided into 16 pie-shaped sectors. At the start of each sector, the buffer control receives an instruction request. The buffer control responds with an instruction, telling the disk the operation to be performed during the sector, as well as the specific data location addresses involved in the instruction. While the disk is moving through the sector, the data is transferred between the buffer control and the disk in response to data request signals sent from the disk. At the end of the sector, the disk sends a status request that signals the end of the operation. The buffer control reads the status report from the disk which indicates the present sector ad-dress and contains trouble status information. The status information is loaded into an instruction queue and examined at a later time by program. Before a buffer control can be restored to service, the disk sequencer in the buffer control must be initialized with the present sector address.

To achieve synchronization, the buffer control disk sequencer, under program control, is initialized to look for status requests only. When

coincident status requests are received from the two duplicated disks, the sequencer reads the disk status reports. In the start up mode, the sequencer extracts the four bits from the status report which corresponds to the current disk sector being accessed by the disk controller. The sequencer queue counter is set to the value of these four bits plus 1 (+1). The sequencer then resets the start up control flip-flop, and advances to state 0 to preload a task for the next sector. Thus, the next instruction request received by the buffer control is honored and the test to be performed is executed.

The tape sequencer must be initialized in one of two states, depending on what it was executing when the stop occurred. If a buffer control to tape transfer was in progress, the sequencer must be initialized to honor a status request. Otherwise, it is initialized to look for a new instruction request. The queue counter must be readjusted since it acts as an operational job pointer. Since the queue may contain an operational tape stop operation, the queue counter must be set up so all tasks will be executed before reaching the operational stop.

Each of these decisions and the initial state of the sequencers are set up by a software maintenance quickstart program. Once the hardware is initialized, startup is directly associated with external hardware stimulus. Restart is only required after buffer control has been stopped by a fault or a maintenance program.

5.1.3 *Error Analysis*

The fault recognition programs are designed to restore a faulty system to normal operation within a few milliseconds. The fault recognition programs accomplish this objective for most faults. However, these programs do not have time to exhaustively test suspect units because message handling will be affected each time a maintenance interrupt occurs. (The fault recognition programs must make a decision on the basis of a brief examination of the suspect units. For most faults, the correct unit is removed from active service and processing continue without any loss of data or service.) Some marginal faults are more difficult to isolate and fault recognition may not discover the fault or may remove the wrong unit from service. Maintenance interrupts will continue to occur until the faulty unit is isolated from service. Error analysis and emergency action routines are used to restore service when persistent interrupts occur. The error analysis routines keep a record of error counts, previous system configurations, and the active-standby status of the units to assist the fault recogni-

tion programs. If interrupts continue to occur, more drastic action is taken by emergency action routines.[5]

5.1.4 *Monitor Mode*

An electromechanical device, such as a disk file, can generate a low level of errors which are not reproducible during fault recognition testing. Although the situation must be ultimately corrected, its minimal effects on the operating system warranted taking several seconds to allow careful programmed analysis of the trouble condition. The most serious consideration is to avoid removing the wrong disk from the active system, thereby causing its contents to get out of date with the active copy. The updating process requires about six minutes and assumes read access to the entire active disk. The fault recognition monitor program interrogates software error counters to detect if a predetermined error threshold has been exceeded by a duplex disk community. Once the error rate has been exceeded, and its source not isolated to a suspect disk, the fault recognition program selects a disk and configures it to respond for both itself and its mate. Rather than removing the remaining disk completely from the active system, it is configured to listen and record only. If the errors persist, the unit removed can be immediately restored to active service. (No update required.) Its mate is then assumed to be faulty and can be completely removed for programmed diagnosis. Although simple, this technique has been extremely effective in preventing user messages from being permanently lost prior to delivery.

5.1.5 *Bootstrap Routines*

Emergency action recovery of ADF equipment is accomplished by software bootstrap programs. This maintenance software decouples all bus configurations and rejoins simplex equipments in a semirandom fashion. Once a complete system is established, it is restarted and monitored for excessive interrupts over a short interval. If interrupts continue, the bootstrap software is repeatedly entered. Since this process is semirandom, a working system will be established, possibly after multiple attempts. The faulty unit is detected when an automatic diagnostic program is executed before joining it to the already working system. Units passing the diagnostic program are updated and joined to the working half.

To avoid simplexing and having mass memory become outdated, two types of bootstrap routines are employed. A hard bootstrap sim-

plexes all units and rejoins them only after a successful diagnostic test. A soft bootstrap which takes less time configures units according to their last known status record that is maintained in the call store complex. Both hard and soft type bootstraps are threaded together for a particular recovery strategy.

5.2 *Diagnostics*

The purpose of diagnostic programs is to thoroughly test a unit that has been removed from service and to generate sufficient test data to isolate the fault to within a few replaceable circuit packs. These programs are run in short segments interleaved with call programs. The diagnostic program for a unit consists of a control program and a series of test routines that are followed in a fixed sequence. These test routines are grouped together to form a sequence (phase) which tests a specific function in the unit, such as the bus access logic. The buffer control, for example, has a diagnostic consisting of a control program and 28 phases of tests.

When a diagnostic test fails, two courses of action are possible. The remaining tests may be run to get additional test data, thereby more accurately pinpointing the faulty pack; or, the diagnostic may be terminated on the basis that further tests will generate inconsistent or misleading results. For either case, the maintenance teletypewriter printout displays the phases that failed, the test results in an octal code (raw data), and a 12-digit trouble number. A maintenance dictionary is used to translate the trouble number by listing all circuit pack locations that could cause the trouble number. (Figure 2 shows a typical teletypewriter trouble report for a fault in tape unit control zero.) The diagnostic results are listed, which include the universal trouble number. A section of the tape unit control dictionary is also shown in Fig. 2 listing the pack location for that trouble number. If the trouble number generated by the diagnostic is not found in the dictionary, then the raw test data listed in the teletypewriter printout is analyzed. A manual which lists the tests, the expected test results, and the logic circuit tested is available to aid in resolving marginal or inconsistent faults.

5.3 *Routine Exercise*

All units in the system are periodically removed from service and diagnosed to test maintenance error detection hardware. This insures the ability of a unit to detect and respond to faults in operational

TELETYPEWRITER TROUBLE REPORT

SAMPLE DICTIONARY ENTRY
FOR TAPE UNIT CONTROL

```
                         TAPE UNIT
**28 DR02 RAW TUCO ⎱- CONTROL 0        BELL TELEPHONE LABORATORIES, INCORPORATED

PH 1 ATP ⎞  PHASES 1-3 ALL TEST PASS
PH 2 ATP ⎬  PHASE 4 SOME TEST FAILED     TROUBLE NUMBER EQ. LOC.  TYPE        REMARKS
PH 3 ATP ⎠
PH 4 STF                                  0746 4048 5439   0-03-22,A006
00000000 ⎞                                                 0-03-25,A332
00000000 ⎟                                0747 2158 4394   0-03-26,A006
00000000 ⎬ RAW DATA                       0751 0482 3095   0-32-24,A299
00000200 ⎟ FOR PHASE 4                    0751 0536 6281   0-17-07,A006
    ⋮   ⎟                                 0751 3469 6342   0-26-15,A011
    ⋮   ⎟                                 0751 4531 6534   0-09-44,A011
00000000 ⎠                                                 0-09-45,A006
4751 3084 8108                            0751 5011 0756   0-09-45,A006    PACK TYPE A6
                                                           0-09-46,A011    LOCATED IN
PH 5 STF                            ⟶    0755 0578 2810   0-07-25,A006 ⎞-  FRAME - 0
00000000 ⎞                            ⟋   0756 0590 4702   0-32-24,A299    ROW NO. - 7
00000000 ⎟                          ⟋                      0-32-35,A295    PACK NO. - 25
00000000 ⎟                         ⟋     0758 1735 2293   0-05-40,A006
    ⋮   ⎟                        ⟋       0759 3958 2358   0-30-19,A006
10020000 ⎟                        ⟋                        0-30-20,A011
00000000 ⎬ RAW DATA              ⟋                         0-30-30,A006
00000000 ⎟ FOR PHASE 5          ⟋                          0-30-39,A004
00200000 ⎟                     ⟋                           0-30-44,A004
00000004 ⎟                     ⟋          0759 5204 2603   0-28-25,A006
00000000 ⎟                    ⟋           0760 7132 6654   0-24-41,A006
00000000 ⎟                   ⟋
00100000 ⎟                  ⟋
    ⋮   ⎟                 ⟋
00000000 ⎠              ⟋

** 2/13/1970 FRI 11:29 ⟋
   4196 5686 6681     ⟋
   UNIV TBL NO.     ⟋
   0755 0578 2810 --⟋
```

Fig. 2—Sample teletypewriter trouble report and sample dictionary entry for tape unit control.

hardware should they occur. The activity of the buffer control community is periodically switched between the buffer controls to detect errors in cross-coupled error indicators and match buses. A history of all errors causing interrupts is maintained and printed out on the maintenance teleypewriter, when a unit is removed by routine exercise for diagnosis.

5.4 Maintenance Audits

A buffer control community contains four sequencers associated with three peripheral controllers, and three bus sequencers. Each of these seven functional sequencers is capable of being started and stopped independently by the central control under maintenance program control. Since F-level maintenance work can be aborted by higher level work, possibly leaving a sequencer in the stopped state, a base-level audit is performed every 8 seconds to look for stopped

sequencers. If no source for a stopped sequencer can be determined, the sequencer is reinitialized and restarted.

Critical parameters and constants are stored in the buffer stores. These constants are related to the number of autonomous data scanner-distributors the buffer control should scan: class of service, character type, speed, and error control. Should the data become overwritten or otherwise destroyed, message processing can stop. To prevent this critical data from being lost and going unnoticed for long periods of time, the system audits the area every 8 seconds. If a bad data word is found, the audit will initiate an emergency action phase 1, causing all of the critical constants stored in buffer store to be reinitialized.

Under normal operating conditions, no maintenance interrupts, the operational processing program is the only means by which the system can be alerted of trouble conditions. These programs, in addition to performing their operational work, must verify that data is kept moving to the various peripheral controllers. Normally, faults which cause buffer control to stop processing disk data also cause a maintenance interrupt which brings in fault recognition and diagnostic programs. However, under unusual circumstances, a maintenance program which has temporarily inhibited disk service request signals may be aborted. Under these conditions, buffer control sequencers are stopped because of lack of stimulus from the disk. The hardware audit would find the sequencer stop flip-flop reset (normal) and release control back to normal processing. For this class of fault, the operational program administering the task queue must schedule a base-level fault recognition test. This program will first interrogate the buffer control error indicators and find no flags set. It will then thread-in a software buffer control restart program so it can monitor the queue. Restarting the buffer control will cause the inhibit service request flip-flop to be reset and the sequencer will cause the backed up task in the queue to be executed. The fault recognition program will conclude all is normal and release control. The operational program, detecting that the queues are now being processed, will discontinue entering maintenance routines. Other sequences are protected from being left stopped in the absence of maintenance interrupts in a similar manner.

VI. AIDS TO MANUAL PROCEDURES

Although most of the switching center maintenance procedures are accomplished using direct program control, other semiautomatic and

manual test procedures are provided where they have become necessary.

Off-line equipment tests can be executed by configuring a central control and the desired equipments on one-half of the duplicated bus system, while the active system is running normally on the other. The active central control can then be made to start and stop the off-line central control, causing it to execute program orders. Normal call processing is unaffected.

Each time program control enters base-level work, the software operates a central pulse distributor point. This pulse drives a meter calibrated in milliseconds, indicating call processing activity. When maintenance software is being entered excessively by soft or hard interrupts or the system is operating with a unit causing high rates of single errors, the meter will fluctuate and indicate higher values. This meter provides a continuous indication of traffic load at the maintenance control center, and alerts office personnel of a trouble condition that must be closely monitored.

An ADF office is equipped with special equipment to maintain disk files. A special disk exercise unit is provided so that all disk addresses can be tested off-line. This test set is used to verify a disk when returned from the factory after repair. In addition, special cleaning and purging equipment is on hand to maintain disk files. A master disk clock writer is also provided to write the clock (program) onto a new file received from the factory.

VII. MAINTENANCE DICTIONARY PRODUCTION

The maintenance dictionaries are used to convert the diagnostic results received from the maintenance teletypewriter to a list of circuit pack locations. The dictionaries were generated basically in the same way as the No. 1 ESS dictionaries—that is, by inserting faults in a test model unit, running the diagnostics, recording the diagnostic results, sorting the data, and printing the results along with the package location. However, improvements were made in the fault insertion procedure. A program was written to search a Western Electric tape containing wiring information for all circuits in a unit. The information was used to generate a complete list of faults for all circuit packs. Faults for spare or unused circuits on these packs have been automatically eliminated. The list of faults for each pack was coded on punched cards and used to control the fault insertion equipment. Programs were designed to store the diagnostic results on any specified

No. 1 ESS ADF permanent file 9-track tape. An IBM computer was used to process the tapes, compute trouble numbers, and print the dictionaries. Approximately 200,000 faults were inserted to produce the dictionaries for the No. 1 ESS ADF units.

The results of the dictionary production indicate the diagnostics are about 85 percent effective in locating the faults detected by the maintenance hardware and software checks. About 15 percent of the faults inserted were not located to replaceable units by the diagnostics or produced inconsistent results. Manual procedures must be used in conjunction with off-line operation to isolate faults not detected by the diagnostics.

To produce a more effective diagnostic and dictionary, better and faster feedback is needed than is possible in the dictionary manual fault insertion procedures. The use of large scale digital computer simulation of logic circuits appears to be the answer.[6] By using digital simulation, diagnostic design and fault insertion can more nearly parallel the logic design phase of a system. Program and logic changes can be made to isolate nearly 100 percent of the faults simulated before hardware designs are frozen. It is likely that future designs will depend heavily on digital simulation to produce fault dictionaries, and significant improvements in the effectiveness of diagnostics can be expected.

VIII. LINE MAINTENANCE

8.1 *Introduction*

The purpose of the line facilities is to provide an interface between the user's station and the common processing units. The ADF maintenance plan includes features for detecting, reporting, and isolating troubles in this equipment dedicated to a user's line. In general, the line maintenance approach is somewhat different from the plan for maintaining common hardware. For example, fault detection in common hardware is based mostly on hardware checks, and the system is interrupted when faults are detected. On the other hand, faults in line facilities are detected by both hardware and software (mostly software), and the system is not interrupted when faults are detected. Message processing continues while fault isolation tests are made. The maintenance procedures involve facilities at the customer location, at local or remote test centers, and at the switching center. Line maintenance, therefore, requires the cooperation of craftsmen at several locations and involves manual as well as programmed tests.

8.2 *Test Center*

The data lines from user stations have appearances at test boards in serving test centers. A line from the switching center to a user terminal could pass through several test centers (as shown in Fig. 3). The serving test center closest to the switching center, through which all lines must pass, is called the control serving test center. The craftsmen at the control serving test center are responsible for troubleshooting and maintaining the transmission and terminal facilities. Troubles with subscriber lines may be detected by monitoring circuits within the test center or by monitoring circuits and in-service message tests at the switching center. When line troubles are detected by the switching center, the test center receives service messages, indicating the type of check that failed and the identity of the station or line in trouble. All test centers involved in suspect
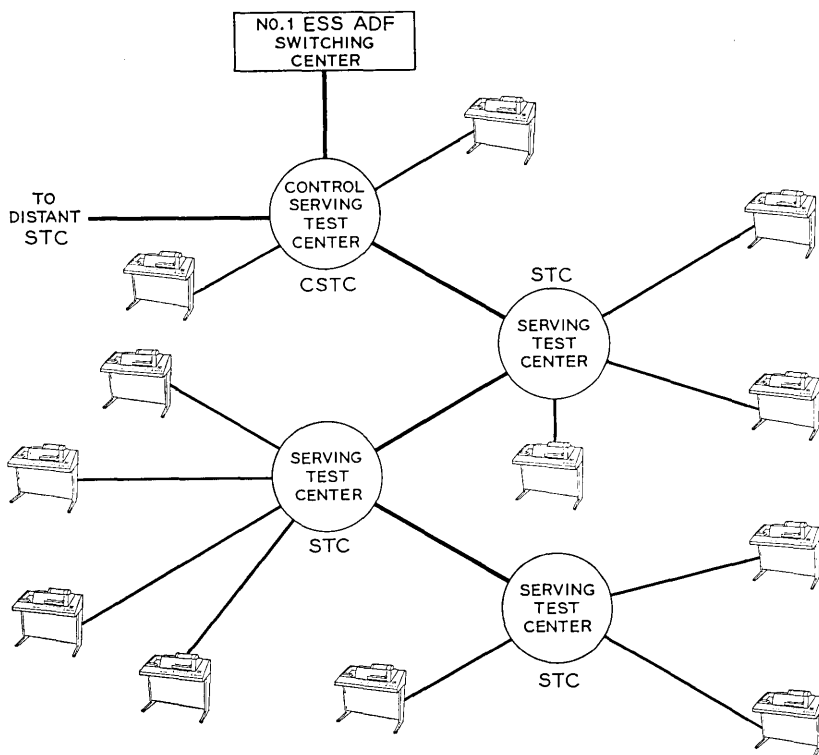
Fig. 3—Facilities for testing user lines.

lines cooperate in testing the complete facility. As part of the trouble-shooting procedure, the control serving test center may send action request messages to the switching center requesting certain tests be performed. These switching center tests do not interrupt service and can detect marginal conditions that degrade service as well as facility failures.

8.2.1 *Control Serving Test Center Features*

The main facilities provided at the test center include a test service board of the type used for private line telegraph service, monitor teletypewriters, and a teletypewriter station to the switching center that is serviced in the same way as a user station. These facilities are used to perform tests that include the following:

(*i*) A continuous open-line monitor checks the line for breaks.

(*ii*) A high signal distortion check monitors the quality of the signals.

(*iii*) Loop tests from the test center test board to the data set at the switching center and back to the test board are used to check the link between the test center and the switching center. This test requires a special circuit pack be inserted in the data set at the switching center to connect the send and receive lines.

(*iv*) Loop tests from the test board to the user's terminal and back to the test center can sectionalize faults in the transmission facilities, station controllers, and terminals.

(*v*) A monitor teletypewriter is available to manually test stations by sending character sequences or to monitor the line.

(*vi*) A patching capability is available to transfer user facilities to spare lines between the test center and the switching center when transmission or terminal troubles are encountered.

8.2.2 *Station Facilities and Action Requests*

The test center must work closely with the switching center and make full use of the system's capability. The switching center can detect service degradation, line troubles, and assist in the test procedures. To communicate with the switching center, teletypewriter stations are used to receive service messages and to send action requests. These stations are connected to the switching center through the autonomous data scanner distributor and are serviced in the same way as a user station. Requests may be sent by the test center per-

sonnel requesting the switching system or craftsman to perform the following functions:

(*i*) Place a station on skip, intercept, or alternate delivery.

(*ii*) Change a station from one autonomous data scanner-distributor port to another. This action may be prompted because of troubles in the facilities between the test center and the switching center or an autonomous data scanner-distributor.

(*iii*) Restore a station to its assigned line.

(*iv*) Perform a distortion measurement on a specified input line.

(*v*) Provide a status report of each station on a specified line, such as stations which are on alternate delivery, hold, or skip.

(*vi*) Stop the delivery of messages to a station or cause a station to stop sending a message.

8.3 *Switching Center Line Maintenance*

8.3.1 *In-Service Checks and Service Messages*

The No. 1 ESS ADF office is programmed to perform in-service checks on messages being processed. For example, the switching center sends special characters to determine the status of the stations, to prepare the stations to send or receive messages, to terminate messages, or to verify reception of messages. If the station fails to respond or gives an incorrect response, the office repeats the sequence; if the failure repeats, the test center is informed of the trouble. The craftsmen at the test center are actively working on clearing troubles before the user recognizes a trouble. Service messages informing the control serving test center of line troubles include:

(*i*) *Polling failure*—Idle stations are periodically polled to determine if the stations have input messages. Failure to receive a valid "no message" response or a "yes, I have a message" response is reported. The polling procedure, therefore, provides a continuous check on the ability of idle stations to communicate with the office.

(*ii*) *Transmitting call enquiry code failure*—A polling response may indicate that a station requests to send a message. In this case, part of the message origination procedure requires the office to send a call enquiry code (CEC) to the station. If the station responds correctly with a start of heading (SOH) character, the office then sends heading and message number information to the station. A failure to respond correctly to the CEC is reported.

(*iii*) *Failure to restart*—After the office has sent heading and message number information to the station, the office restarts the station

teletypewriter transmitter. The station then sends the message. A failure to respond to the restart is reported.

(*iv*) *Station call-in failure*—Before a message is delivered to a station, the office determines if the station is ready to receive. Failure of the station to send a valid "ready" or "not ready" response is reported.

(*v*) *Roll call failure*—After a message is delivered to one or more stations on a line, the office roll calls each station receiving a message. If a negative roll call response is received, the message delivery is repeated. If a negative roll call response occurs again, the trouble is reported.

(*vi*) *Loss of control*—The office reports it has lost control of a line when a transmitting station will not respond to a request from the office to stop sending.

(*vii*) *Loss of facility*—The line from the test center terminates in a data set at the switching center. An open line will be detected at the data set by a ferrod monitor. A failure will be reported if an open line exists.

8.3.2 *Character Parity*

For stations that use the ASCII code, a parity bit is included for each character. The buffer control checks the parity of each character and replaces the character with a slash (\) symbol if the parity fails. When the message is delivered, the customer has the option of requesting the message originator to retransmit the message if a vital character was lost. The terminating station also checks parity on each character, and an underline (_) character is printed if a parity failure is detected. In this case, the error occurred in the facilities used for message delivery. If vital characters were lost, the message can be retrieved from the switching center by an action request.

8.3.3 *Line Facility Loop Test*

The line facilities within the switching center can be tested by connecting the transmit line to the receive line at the switching center data set. This test loop is set up by replacing a pack in the data set with a special loop pack. Test messages can be sent under program control to the transmit channel, looped around to the input or receive channel, and checked by the program. This feature is used to help sectionalize troubles in the link between the control serving test center and the switching center. The test may be requested by an action request from the test center.
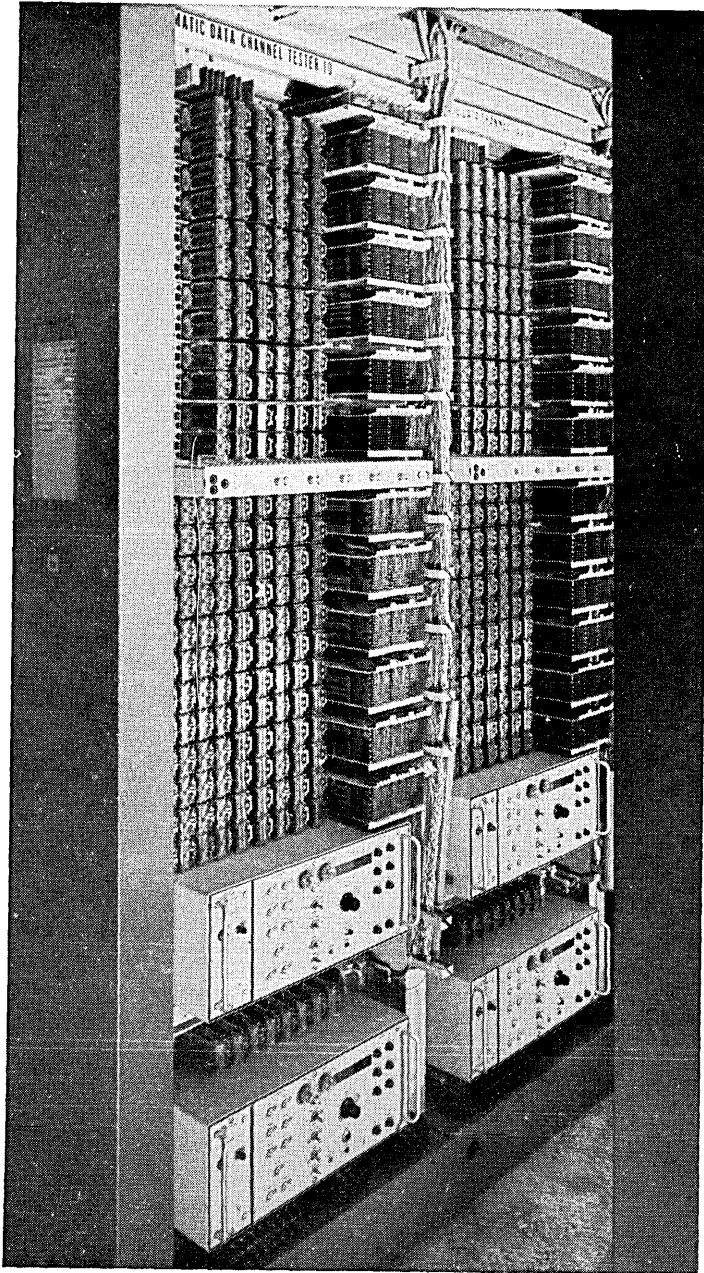
Fig. 4—Automatic data channel test facility.

8.3.4 Automatic Data Channel Test

The quality of the input signals transmitted from a user's station to the switching center can be determined by the automatic data channel test facility, Fig. 4. Each input line is provided with a bridged connection to a distortion measuring set that tests the quality of input messages. The line to be tested is specified by an action request teletypewriter message. A relay network under program control provides access from any designated line to one of four distortion measuring sets. The test set measures the element transitions within each character and compares them with the theoretical element duration. The highest distortion detected for the duration of the test is indicated to the system. A teletypewriter message reports one of seven ranges of distortion for the line under test. After a valid distortion reading has been obtained, the program releases all connections to the test facility.

An action request teletypewriter message from the control serving test center may request the distortion test on a specific line as part of a fault isolation procedure.

IX. CONCLUSIONS

A great deal of hardware and software has been devoted to implementing the maintenance plan described. In addition to modifying existing No. 1 ESS maintenance programs, approximately 100,000 words of new maintenance programs were written. About 60 percent of the stored program is devoted to maintenance procedures and duplication is used extensively to achieve reliability. A No. 1 ESS ADF office has been in operation since February, 1969.

The performance of the system has been good. As might be expected with any new system, improvements and corrections have been made as weak points in the program and hardware were uncovered. Based on the experience to date and the improvements that have been made, the system is performing as expected and should meet the long term performance and reliability objectives.

X. ACKNOWLEDGMENTS

REFERENCES

1. Corbin, J. E., Lehman, H., and Townley, R. C., "No. 1 ESS ADF: Autonomous Data Scanner and Distributor," B.S.T.J., this issue, pp. 2857–2885.
2. Aitcheson, E. J., Ault, C. F., and Spencer, R. G., "No. 1 ESS ADF: Message Store—A Disk Memory System," B.S.T.J., this issue, pp. 2887–2913.
3. Potter, J. L., Strebendt, Mrs. F. B., and Williams, J. R., "No. 1 ESS ADF: Magnetic Tape Subsystem," B.S.T.J., this issue, pp. 2915–2940.
4. Genke, R. M., Harding, P. A., and Staehler, R. E., "No. 1 ESS Call Store," B.S.T.J., *43*, No. 5, Part 1 (September, 1964), p. 2147.
5. Downing, R. W., Nowak, J. S., and Tuomenoksa, L. S., "No. 1 ESS Maintenance Plan," B.S.T.J., *43*, No. 5, Part 1 (September, 1964), pp. 1988–1993; 2016–2018.
6. Seshu, S., "On an Improved Diagnosis Program," IEEE Transactions on Electronic Computers, *EC-14* (February, 1965), pp. 76–79.

No. 1 ESS ADF:

Autonomous Data Scanner and Distributor

By J. E. CORBIN, H. R. LEHMAN and R. C. TOWNLEY

(Manuscript received August 13, 1970)

This paper describes the operation and design features of the data-scanner distributor for the No. 1 ESS ADF. The objective was to design a unit which could handle handshaking, addressing, and message transmission for a large number of duplex data lines at a low cost and with the ability to accommodate all speeds and codes now commonly used in low-speed data machines.

The storage medium chosen employs ultrasonic delay lines in a highly time-shared mode. This permitted a compact design which provides input/ output capabilities to serve 512 fully duplex data lines. By using a high-speed scan, each character bit of each line is sampled many times. Thus, highly distorted input data can be successfully processed, and output data is essentially distortion free.

I. INTRODUCTION

Equipment to perform the input/output function in a data-switching system handling a large number of connecting data lines has been one of the most expensive and space-consuming parts of the system. This is due primarily to complex handshaking requirements and to the need for handling addressing and all message text by the input/ output unit. To achieve an economical solution, a wired logic unit was designed with the extensive use of ultrasonic delay lines in a time-shared, multiplexed mode.

This paper describes the system organization, the logic, and the circuits that are unique to the data-scanner distributor (DSD) for the No. 1 Electronic Switching System, Arranged with Data Features (No. 1 ESS ADF). Unique designs—such as the data line circuit, the delay-line memory loops, the addressing structure, and the control unit logic—are presented. Also, some of the unique maintenance features required for an autonomous wired logic unit that were employed are described.

II. GENERAL DESCRIPTION

The autonomous DSD terminates 512 half-duplex (HDX) or full-duplex (FDX) lines operating with Baudot codes at 60, 75, and 100 words per minute; CCITT code at 66.6 words per minute; and ASCII codes at 100 and 150 words per minute. The DSD's basic function is to perform the serial-to-parallel and parallel-to-serial conversions on the data bit streams; these conversions provide a compatible interface between a large number of low-speed data lines and the ADF system and provide the time buffering needed because of speed differences at the interfaces. The DSD changes the Baudot or ASCII characters which are in serial form on the customer's line into parallel characters for efficient handling in the system and reconverts the parallel words being sent to the data stations into serial characters. The DSD must perform these functions to agree with the format of the data machines on each customer line.

A simplified block diagram of the DSD unit is shown in Fig. 1. This DSD unit is duplicated except for the line-terminating circuits and is capable of providing service for 512 FDX, low-speed asychronous data lines, each capable of being assigned to any one of six TTY rate/code types.

The DSD unit is a time-division multiplex system with 1024 time-slots—an input and an output slot for each of 512 lines. The common control of the DSD samples each line approximately 1650 times a second and stores the sample at the center of each data bit. This sampling frequency permits sending characters with a maximum of
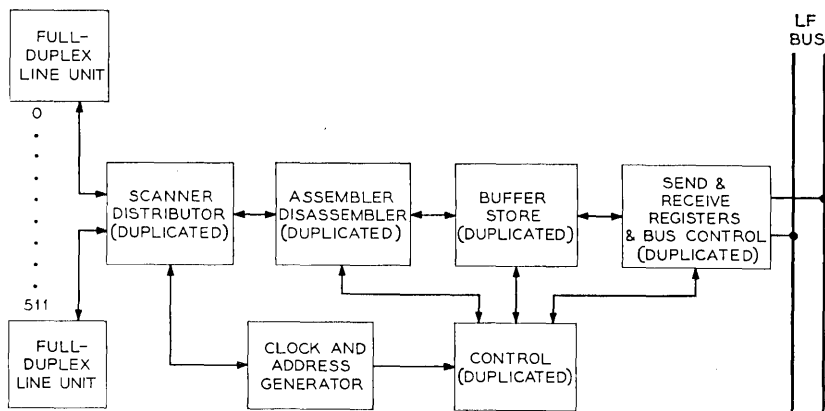
Fig. 1—Block diagram of the data-scanner distributor.

one percent distortion and accepting characters with up to 45 percent distortion at 150 baud.[1]

A high-capacity, line-address-oriented dynamic delay line memory, with access at the scan rate, is used to handle character assembly, disassembly, and control requirements. Sufficient buffering is provided to allow reasonable transfer time across the interface between the DSD and the processing system. To meet these needs, the scanner distributor provides storage for six data characters and two control words for each customer duplex line served as follows:

(i) an input character being assembled,

(ii) a completed input-data character,

(iii) an output-data character being disassembled,

(iv) three output-data characters awaiting transfer to the disassembler,

(v) a control word for input-character scanning, and

(vi) a control word for output-character transmission and sequencing.

To comply with reliability and maintenance requirements of the ADF system, the DSD has two independent, duplicate halves, each receiving inputs from a common line-terminal area but having separate access to the system. Furthermore, the two halves operate in the duplex and synchronized mode as well as in a divorced simplex mode in which either half can continue to serve the entire line-terminal load while the other half is out of service for maintenance.

The DSD communicates with the buffer control (BC) by means of the line facility bus system, which is electrically equivalent to the No. 1 ESS call-store bus system.[2] A reliable high-speed communication bus between the scanner and BC is necessary to handle the magnitude of data transfers, service requests, and maintenance operations. All DSDs in the office share duplicated send, control, and receive bus systems with each DSD capable of accessing either bus. Each scanner responds to its own name code (sent on the control bus) and communicates over either or both duplicated bus systems as directed by the central control. Central control (CC) has the option of changing or rerouting the bus assignments when a unit requires maintenance. The DSD maintenance programs update the DSD memories and resynchronize a DSD to restore it to duplicate processing when the out-of-service unit is to be returned to service.

Each DSD possesses flags in the form of ferrods which are scanned periodically to determine the state of health of each DSD. If failure

occurs, the unit divorces itself from the line circuits and operates the proper flag which, in turn, calls in the CC to perform fault recognition and diagnostics.

Power to each half of the DSD must be independent of power to the other half. However, common circuits such as line-terminal units, clock circuits, and delay-line oven heaters must receive power on a logic OR basis from separate power buses so that they are always functional as long as power is on either DSD.

III. SYSTEM OPERATION

3.1 *General*

The organization of the DSD makes it equivalent to a start-stop TTY with a capacity to handle simultaneously 512 duplex-line terminals. Each connecting line terminates in a line circuit serving each of the duplicated DSDs. A fully equipped scanner distributor utilizes 3103 No. 1 ESS type circuit boards of which 1024 are used for line termination. For every 606.06 μs, each line is addressed for 1.184 μs; this time is devoted to both inputting from the customer line and outputting to the line, assuming it is FDX.

3.2 *Input/Output Processing*

If during the input time the line is found in a "space" state (0), this fact is sent through the converging tree to the start-detect delay line and stored in the time-slot corresponding to the particular address. This operation requires 0.592 μs of the address time; similarly, during the other 0.592 μs of address time a bit is sent to the same connecting line. Since the lines are addressed in sequence, they are connected to the assembler/disassembler through a converging ORing circuit for inputting, and through a diverging ORing circuit for outputting. Each line circuit is connected through these OR circuits to the "type" gates, depending on how the TTY terminal is equipped. When the address comes up, the line-terminating circuit actuates a select circuit, which enables the proper type gates for the line being addressed for both into and out of the assembler/disassembler. A pulse from the control unit decoders enables the type gates at the appropriate count permitting data bits to be gated into the assembler during inputting time and gating the data bits out of the disassembler during outputting time. A block diagram of the input function is shown in Fig. 2.
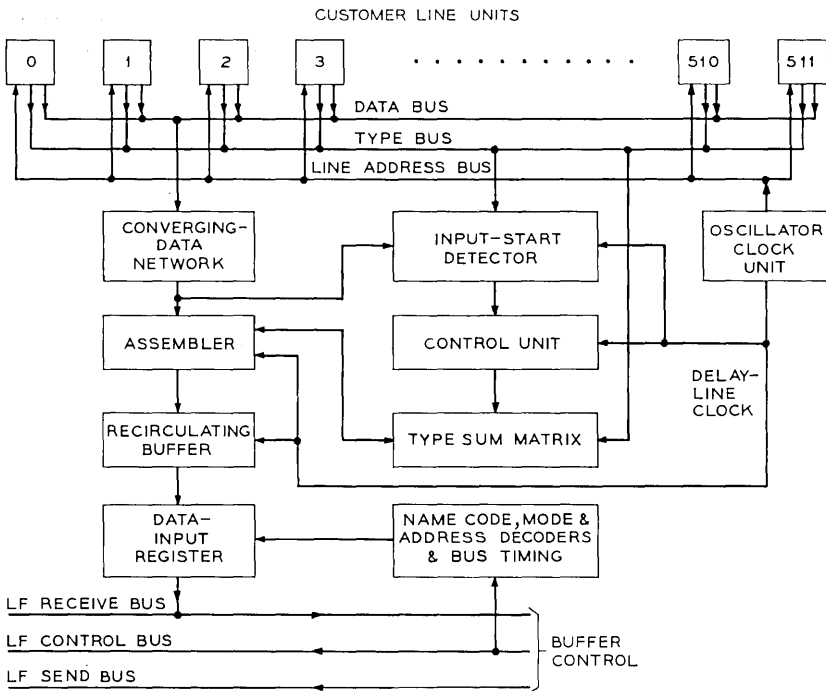
Fig. 2—Simplified functional block diagram DSD input portion.

The assembler/disassembler is a group of 11 delay lines used simultaneously to assemble and disassemble serial characters as required for data line operation: line 1 stores the start bit, line 2 stores the first data bit, etc. The number of bits stored depends on whether a Baudot or ASCII code is required for a particular line. In addition one of the 11 lines is used for parity, internally generated for maintenance purposes. The filling of the assembler/disassembler or the emptying of it is controlled from the control unit. This process is referenced to the start bit which is always the first bit of the character string.

When the counter has completed the count for a particular line address, the contents of the assembler are gated out in parallel form to the recirculating buffer. The character remains in this circulating buffer store until it finds the data-input register empty, at which time it is gated into the register along with the line address from the address counter. A service request is set up, and a parity bit is added

to the data and line address. The data now waits for the buffer control to recognize the service request and accept the contents of the data-input register.

For outputting, a similar process is followed (see Fig. 3). The control unit empties the disassembler one bit at a time to the subscriber line. When a character is completed, as determined by the control counter, the next character is gated out of the recirculating buffer in parallel form into the disassembler. Since the output buffer receives three characters at a time from BC, sequence logic is required to control orderly flow of the characters into the assembler/disassambler. Two delay lines are used to store these control bits for each customer line address.

As Table I indicates, the DSD insures a marking state on the customer line even though the line is idle. As long as a customer line is active, the control count continuously recycles, and the sequence bits rotate through the 3-count sequence for characters 1, 2, and 3 without returning to the idle state.
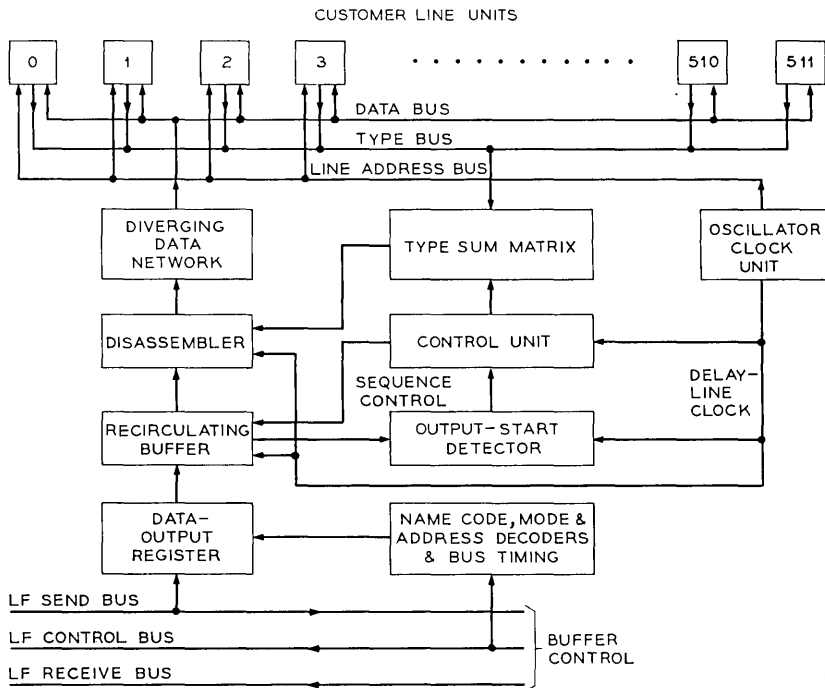
Fig. 3—Simplified functional block diagram DSD output portion.

TABLE I—SEQUENCE BITS
(Control of Buffer-to-Assembler/Disassembler Interface)

SEQ Bit 0	SEQ Bit 1	Function
0	0	Idle (Marking)
1	0	Gate Character 1 and Start Count
0	1	Gate Character 2 and Start Count
1	1	Gate Character 3 and Start Count

The sequence bits also set up a service request when the third char-
acter has been loaded into the disassembler. Along with the service
request is a line address which makes it possible for the BC to fetch
the proper data for the line address being served.

The control system in the DSD consists of the clock, the address
counters, and the time-shared arithmetic unit. The latter unit pro-
vides the gating signals for loading and unloading the assembler/dis-
assembler at the times required by the line types. It also provides
the signals for loading and unloading the recirculating buffer at the
assembler/disassembler interface. When a customer line becomes ac-
tive, the start bit is added at each turn of the delay lines to the
contents of the 9-bit control counter, which indexes the number by
one. When the binary number in the control-unit recirculating store
reaches the predetermined count for the type of line, a gate pulse is
produced to transfer a character bit. Thus, it is this time-shared adder
that controls the flow of characters through the DSD.

IV. MEMORY DESIGN

4.1 *Delay Line Memories*

The memory medium selected for the DSD was the ultrasonic alum-
inum delay line.[3] Twenty lines are housed in a 23″ by 10.3″ by 6″
temperature-controlled box. Thus, 18″ of rack space contain 61,440
bits of serial memory with a 606.06-μs recycle time. The logic involved
in the delay line memory loops is shown in Fig. 4. The delay lines are
nominally 300 Ω at the input and output terminals with an insertion
loss of 36 dB ± 0.5 dB. To provide for the various loop-transmission
tolerances, the line had to provide a 5-to-1 pulse to side lobe ratio
under the worst-case bit pattern of alternate 1s and 0s. Because of the
relatively high temperature coefficient of the aluminum used as the
delay medium, the lines were housed in an insulated box and held at
60°C ± 0.03°C controlled by an on/off power switch with a fine mercury-
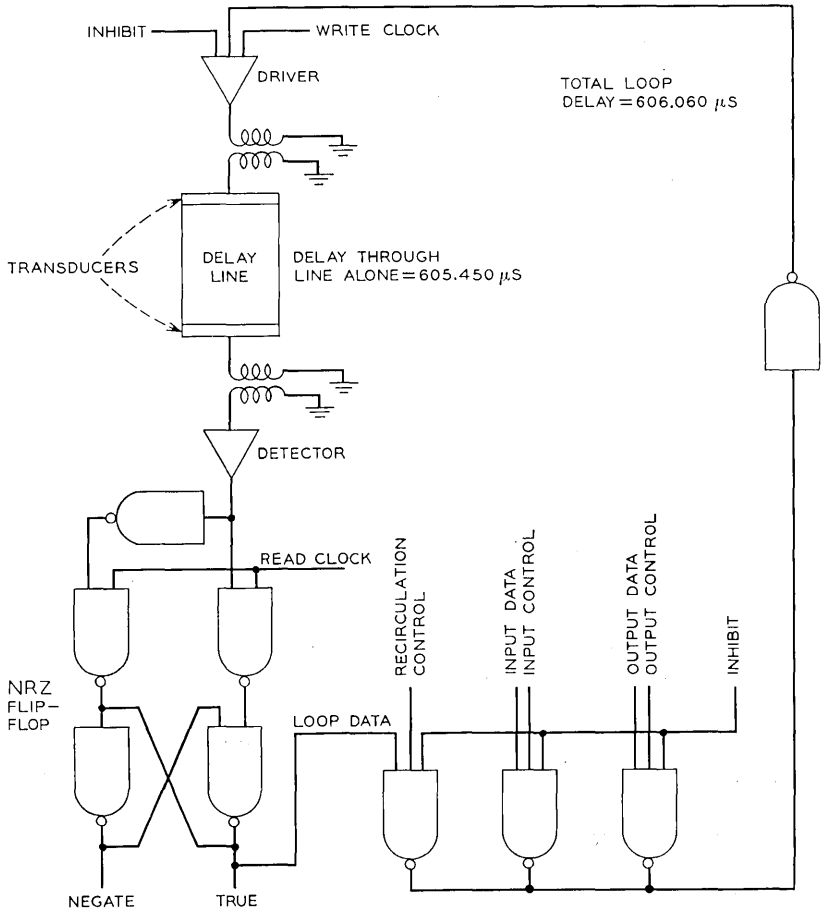
Fig. 4—Typical delay-line loop.

column thermostat as the sensing element. Two alarm thermostats are included to sense when the temperature-control circuits develop trouble. These are set to notify the system of failure in the temperature-control circuit before the line delay has changed sufficiently to cause data-processing errors. When this occurs, one of the alarm thermostats is actuated and the fault-recognition program removes the faulty DSD half from service.

To achieve the main lobe to side lobe ratio required, the piezo-electric transducers located at each end of the aluminum strip were

tuned about 25 percent higher in frequency than the scanner-clock frequency.

This technique made the side-lobe vector sum when a 0 appears between two or more 1s in a data stream to be no higher in amplitude than a side lobe of an individual isolated 1 pulse. This transducer tuning assures the required signal-to-noise ratio for any bit combination stored in a line loop. Typical waveforms for several bit patterns at the delay line output are shown in Fig. 5.

Each scanner utilizes 58 delay lines to implement both the input and output functions. The address time of 1.184 μs is divided, the first 0.592 μs is used for outputting, and the last 0.592 μs is used for inputting. The timing cycle for each delay line is shown in Fig. 6.

The 58 lines are arranged in the following functional groupings: 11 lines in the assembler/disassembler, 10 data and 1 parity; 33 lines
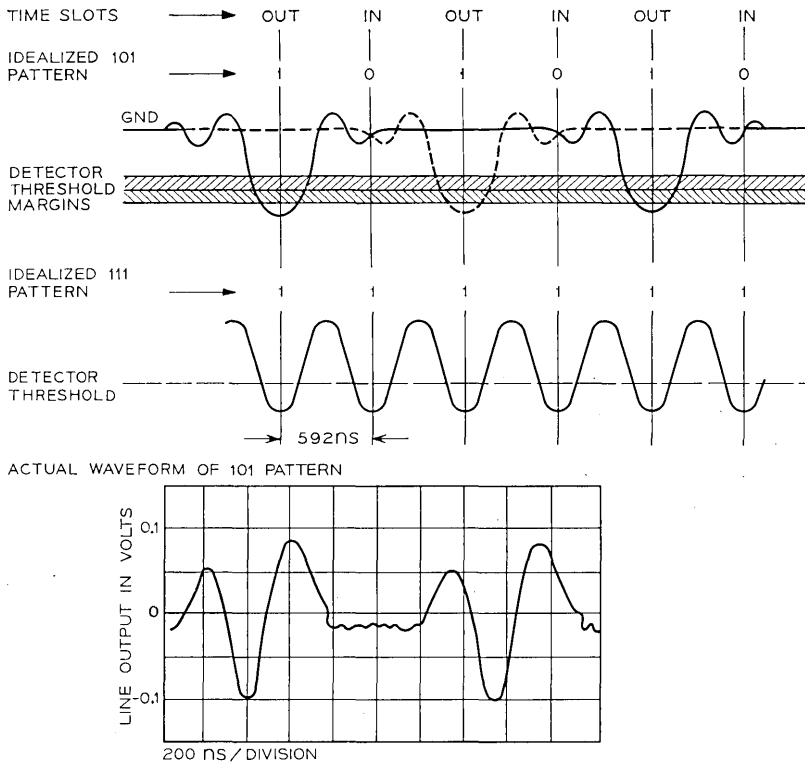
Fig. 5—Delay-line waveforms.
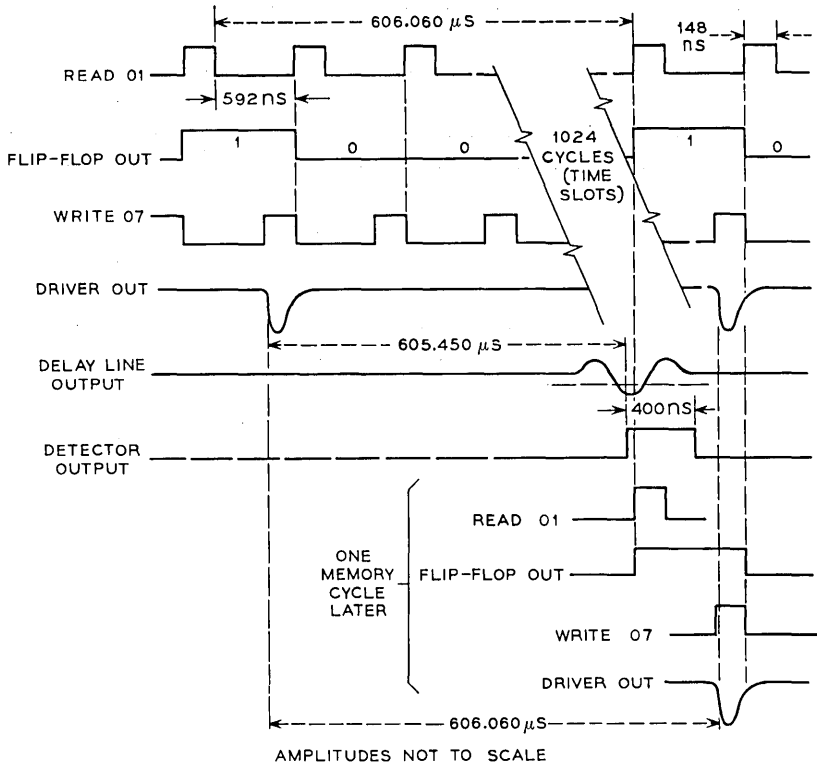
Fig. 6—Delay-line loop timing.

in buffer store, 10 data and 1 parity for each of the three characters; 14 lines in the control unit.

The delay-line driver circuit shown in Fig. 7a provides a constant drive current independent of power-voltage variations by regulating the base voltage of the driving transistor. Figure 7b shows the receiver circuit with approximately 20-dB gain stabilized by feedback plus a pulse stretcher for absorbing loop tolerances.

V. INTERNAL LOGIC STRUCTURE

5.1 *General*

A complete DSD consists of the basic functional sections as shown in Fig. 8. All normal operating logic has maintenance circuits and operational logic interspersed. Physically the duplex unit is housed

in five racks arranged symmetrically about the center rack. Common circuits such as the oscillators in the center rack and the line terminations arranged in the two end racks are equally divided.

5.2 Clock

The oscillator-clock unit (OCU) is designed to supply the primary timing waveforms for the DSD with no discontinuity in frequency or phase. The OCU utilizes a pair of crystal oscillators phase-locked in a master slave configuration. The design is such that a catastrophic failure of either oscillator or the phase-locked circuit by itself will cause minimal phase discontinuity of the waveforms. Figure 9 is a simplified block diagram of the OCU.

The oscillators generate the primary frequency of 3.3792 MC with
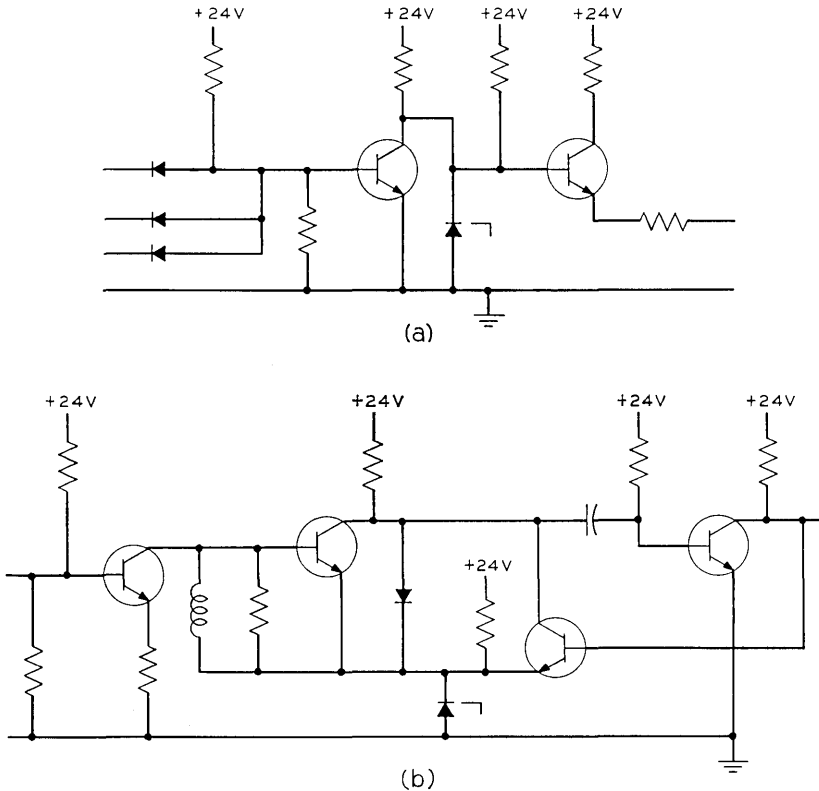
Fig. 7—(a) Delay-line driver. (b) Delay-line receiver.
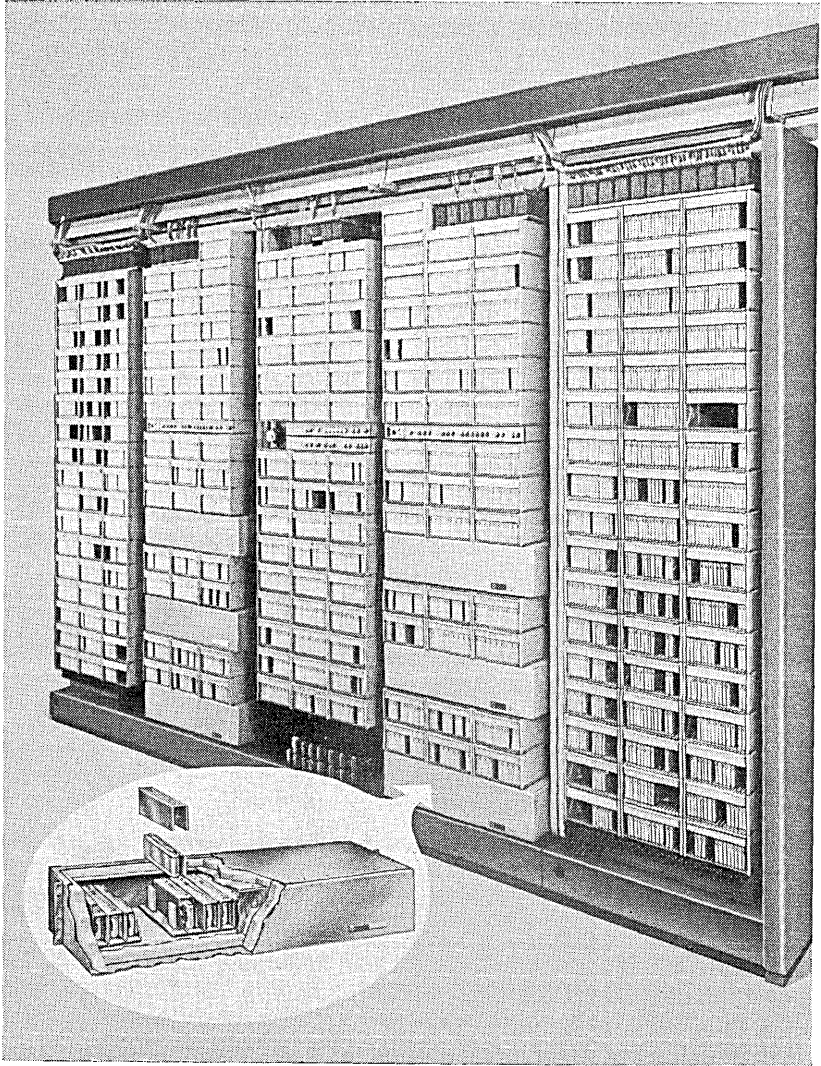
Fig. 8—DSD with detail of delay-line memory.

an accuracy of 20 ppm over a temperature of 0°C to 55°C. The phase-locked sinusoidal signal from the slave oscillator is mixed on a binary-weighted basis with the signal from the master oscillator. The resultant sine-wave output is the vector sum of the individual oscillator signals. This signal is next resolved into two phases. One phase is that of

the resultant signal itself, and the other is derived by means of a 90° (74-ns) phase-shifting network. The two phases are individually shaped into symmetrical ac-coupled square waves by means of a nonsaturating differential amplifier stage. The shaped signals are routed to individual phase-splitting circuits where each wave is further resolved into two signals 180° out of phase. In this manner the OCU produces four phases of primary clock waveforms for one simplex DSD and duplicates these stages for the other simplex DSD beginning with the binary mixer.

5.3 *Address Structure*

Each simplex DSD unit contains two sets of binary counters known as the left and right address counters, which are toggled by the high-frequency output gates of the clock. Waveforms for the address and toggle pulses are shown in Fig. 10.

The toggle is used as the least-significant bit of the 9-bit address. The eight bits of the left counter address lines 0 through 255, and the right 8-bit counter addresses lines 256 through 511. The toggle produces an interleaving of the two groups of line units so the addressing goes 0, 256, 1, 257, ···, 255, 511. In this way each counter is running at half-rate and thus permits the line-terminating logic ample time to settle before data is received or sent to the customer's line. The resultant waveforms from the left counter overlap those of the right counter by an amount equal to the toggle half-period as indicated in Fig. 10.

Figure 11 is a simplified block diagram of the address-counter sys-
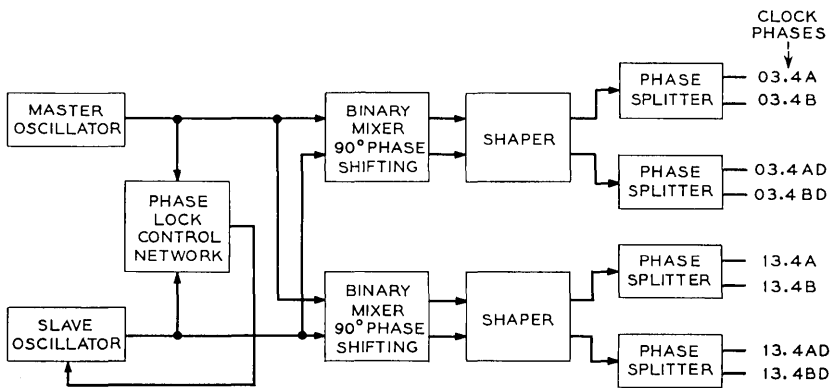
Fig. 9—Block diagram of oscillator-clock unit.
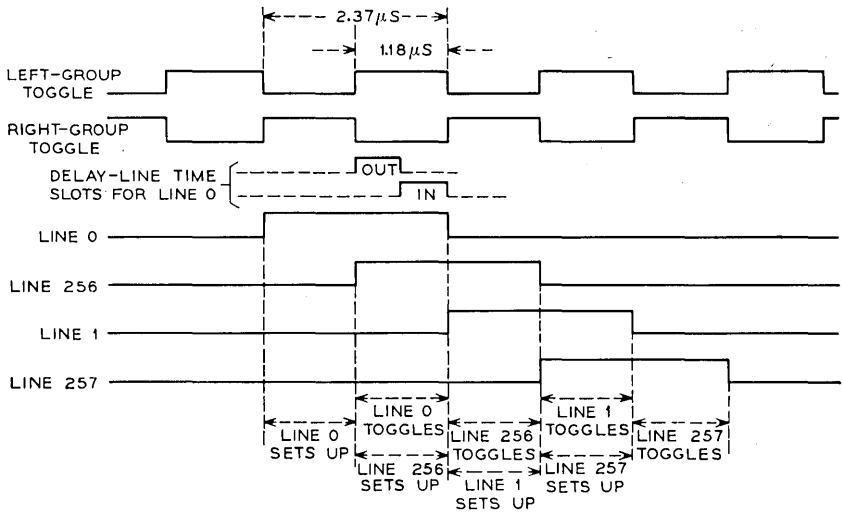
Fig. 10—Line address and toggle system.
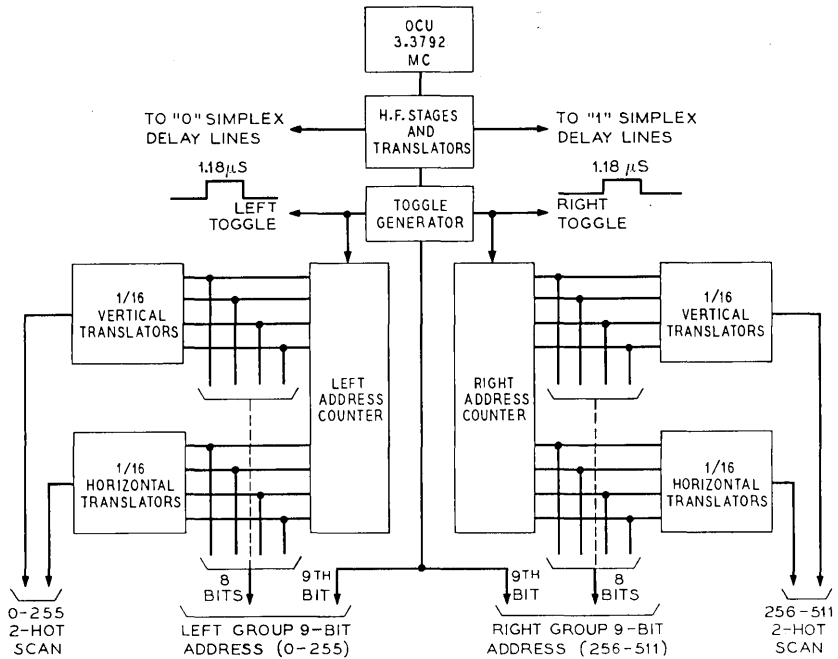
Fig. 11—Block diagram of address counters.

tem. In addition to the 9-bit address, each of the counters, left and right, drives two sets of 1-of-16 translators to develop the 2-hot scan system to the line terminals. The last stage of translation of 1-of-512 is in each line terminal unit. The matrix formed by the horizontal and vertical translator output distribution determines the sequence of line addressing.

5.4 *Data Line Input/Output Circuit*

Data is transmitted in and out of the DSD to and from the data network through standard multilead switchboard cable used in an unbalanced dc connection. It is expected that some installations of these cable lengths may be several thousand feet. Noise and crosstalk might be higher than could be tolerated with the standard No. 1 ESS logic. To provide sufficient margin, a special line-terminating pack was designed. The logic circuit is shown in Fig. 12. The logic level was
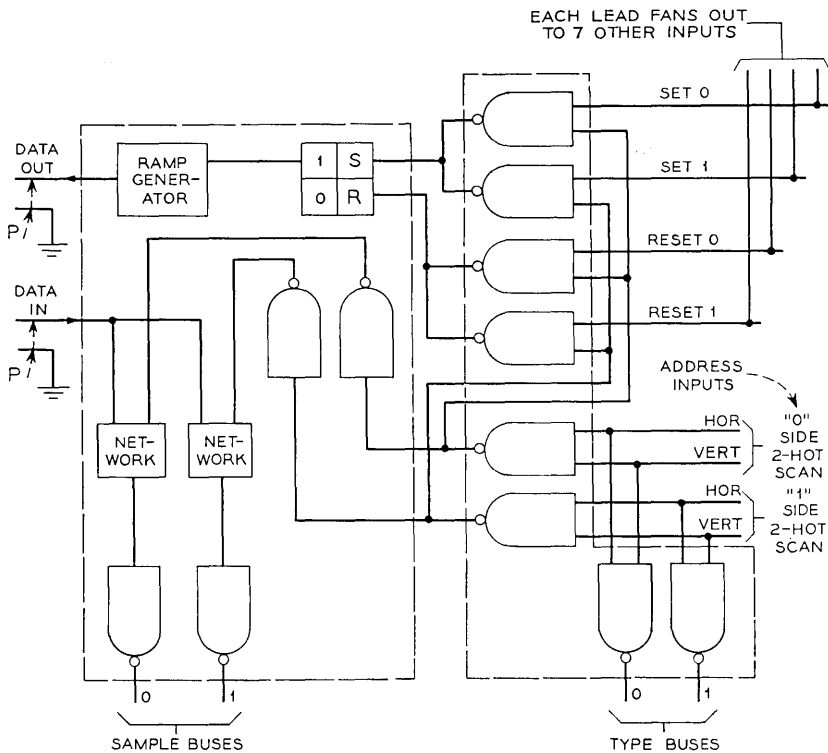
Fig. 12—Line-terminal unit.

raised to 12 V, and the leading and trailing edges of the character pulse were made to rise and fall linearly at approximately 1 volt per microsecond.

When addressed, the line unit logic is unclamped allowing input and output transmission to occur. Also, a type signal is placed on the appropriate type bus to gate this data in and out of the assembler/disassembler.

5.5 Scan Control

The control count at which specific actions are taken during the process of taking samples of input data or generating transitions of output data is determined by the type of code assigned to serve the line-terminating unit (LTU). Figure 13 shows an example of the control count sampling intervals for an ASCII 150-word per minute character. The element duration for this code is 6.67 ms, and the scan rate of 606.06 μs allows a minimum of 11 scans per element. As soon as an input start is detected, the control counter begins to count; when the count of five is reached a sample of the start bit, which is present on the customer's data input, is taken. The sample is within the window bounded by the middle 10 percent of the element and is thus taken at a distortionless point as required in high distortion TTY signals. Succeeding samples are taken at 11 scan intervals, such as 16 for bit 1, 27 for bit 2, and so on to the end of character count.

1. START DETECTED	CONTROL UNIT COUNT STARTED	7. BIT FIVE READ IN	COUNT 60
2. START BIT READ IN	COUNT 5	8. BIT SIX READ IN	COUNT 71
3. BIT ONE READ IN	COUNT 16	9. BIT SEVEN READ IN	COUNT 82
4. BIT TWO READ IN	COUNT 27	10. BIT EIGHT READ IN	COUNT 93
5. BIT THREE READ IN	COUNT 38	11. STOP BIT READ IN	COUNT 104
6. BIT FOUR READ IN	COUNT 49	12. END OF CHARACTER	COUNT 105
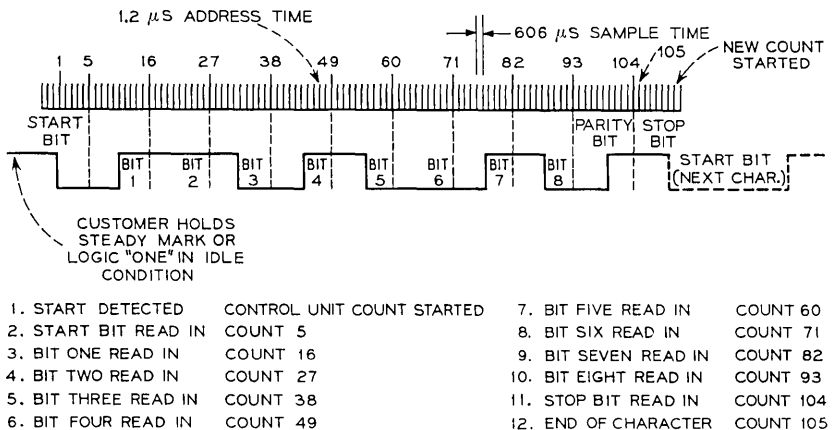
Fig. 13—Control-count intervals for inputting the TTY character K. (Binary 1101001 ASCII 150 with even parity.)

The character now assembled is transferred to the buffer by the counter after which it is reset to zero awaiting the start of the next input characters. Each of the sampling points is thus predetermined for a given code type, and a fixed-count sum can be established to enable proper sample-scanning action. The samples are derived from wired logic patterns from the outputs of the control delay line group. The control unit utilizes nine delay-line loops in addition to the start detect loop in its add-one counter section. A fast-carry logic incrementer is used to steer the counting action to the delay line loops. Figure 14 illustrates a segment of the control counter. The outputs from the counter enable the choice of any integral 9-bit code required for a sampling point. The highest sum required is determined by the slowest code type. The total number of gates required for sampling one code type depends on the number of elements in the code. A Baudot 60 line, for example, requires a count of 269 to complete the generation of an output character, but requires fewer sum gates for sampling since it is a 7-element code. The type-sum gates are an array of gates that input and output data to and from the assembler/disassembler. These gates are controlled by ANDing the type bus and the decoded control unit output and the data pulses for each of the 1024 time slots. The array of sum gates are arranged by type in a matrix. The matrix consists of a 6 by 10 gate array corresponding to the number of types (6) and the maximum number of bits in the ASCII code. All of the outputs of the type-sum matrix converge to the input gates of the assembler/disassembler.

5.6 *Assembler/Disassembler Unit*

The assembler/disassembler is a group of 11 delay-line memories used as a time-shared shift register for both input and output functions. The outputs of the type-sum matrix are fed in order to the input gates of each delay-line loop in the assembler/disassembler where the data sample bus also converges. Each element of an incoming character is sampled into the proper time slot and delay line as the simultaneous occurrence of the customer's scan interval and a particular count in the control unit take a sample of the data bit from the LTU. For example, as shown in Fig. 13 for an ASCII 150 coded line, the start element is sampled on the fifth interval after the start transition has been detected and the control count has started. The start-element sample is stored in the first delay-line loop of the assembler. In the same manner, the following elements are sampled
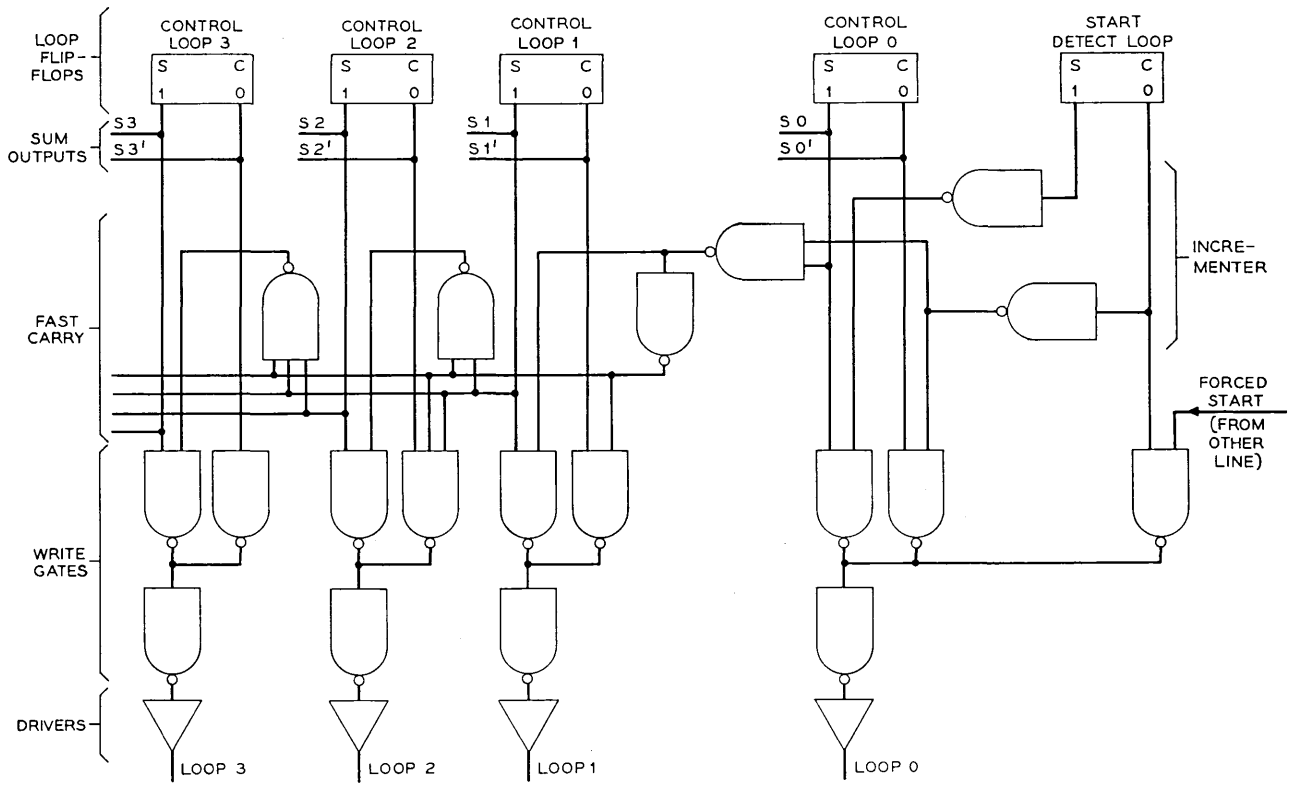
Fig. 14—Simplified sketch of control counter.

at later count intervals and are stored in proper sequence: bit 1 in the second line of the memory, bit 2 in the third line of the memory, and so on, until a complete character is stored in the same time-slot of each memory line. Each successive sample continues to circulate the delay-line memory loops until all samples have been stored for the character being assembled. When completed it will be read out in parallel form to the recirculating buffer. Each set of time-slots in the assembler is under independent control of the control unit. The assembler may be assembling at one time as many as 512 incoming characters in various stages of completion.

These functions of processing the characters through the DSD for both data input and output are controlled by the time-shared control counter described in Section 5.5.

The output time-slots of the assembler/disassembler unit are used for the disassembling process of output data. The output operation is just the reverse of the input operation. When a parallel character enters the disassembler from the recirculating buffer it is read out in serial to the customer line. An output count is started in the control unit at the time the character is read out of the recirculating buffer. At specific count increments, determined by the proper spacing for undistorted elements, the data stored is sampled out of the disassembler, element by element, to set or reset the output flip-flop in the LTU for the customer line to which it is directed.

Data in all delay lines of the assembler/disassembler is always stored in true form. A mark is represented by a logic 1 in the line, and a space is represented by a logic 0. Because the Baudot and CCITT codes do not involve as many elements as the ASCII codes, ten delay-line loops are sufficient to store all code types. Only one stop element is stored for any code type. The unused delay-line time-slots for the shorter codes remain in the cleared (logic 0) state at all times. One delay-line loop in the assembler/disassembler is used for internal parity on both input and output characters.

5.7 Recirculating Buffer Unit

The recirculating buffer unit is composed of a total of 33 delay-line memories sufficient for three ASCII characters plus parity for each character. These memories are addressed synchronously with those of the control unit and of the assembler/disassembler unit. Fully assembled input characters are transferred from the assembler/disassembler unit to the recirculating buffer at end-of-character count. The

assembled input character elements plus input parity from the assembler/disassembler occupy the input character 1 slots of the recirculating buffer. The character is stored in the recirculating buffer until such time as it gains access to the data-input register where it awaits a normal read request from BC to transmit the character to the system. In the output direction, the recirculating buffer receives three characters from BC. The characters await entry in proper sequence to the disassembler at the time the line address occurs. The sequence bits of the control unit are used for sequentially gating characters 1, 2, and 3 from the recirculating buffer to the disassembler. An individual parity bit is carried through the output process for each of the three characters.

5.8 *Input/Output Registers*

The DSD has seven major logic registers to transfer information between the DSD and the system. Table II lists the registers and their functions. Each of the registers is specifically interrogated by the system by means of mode orders on the control bus. Periodic scans keep input and output data flowing with the assistance of service requests to assure legitimate transfers.

TABLE II—INPUT/OUTPUT REGISTERS OF DSD

Register	Use	Function
Normal Read	Input	Transfer input character to system. (9 data bits, 9 address bits, 1 input service request bit, 1 overwrite bit)
Scan Output	Input	Transfer output service request into system. (9 address bits)
Input Maintenance	Maintenance Read	Transfer maintenance data into system. (12 data bits, 9 address bits)
Output No. 1 and Output No. 2	Output	Receive output data from system. Two registers with priority to No. 1. (Each register 33 data bits, and 9 address bits)
Output Maintenance	Maintenance Write	Receive maintenance instructions from system. (30 data bits, 9 address bits)
Error Status	Maintenance Read	Store trouble reports for transfer to system. (22 data bits)

Maintenance registers are used during routine exercises or diagnostic examinations. The status register is a fast means of reporting the states of vital functions which are internally monitored by maintenance circuits.

Entrance of an input character from the recirculating buffer to the normal-read register is a random operation controlled by the start bit of the character, which is used as the input service request, and a circuit which monitors the empty status of this register. When the normal-read register becomes empty, it accepts the next start bit that appears at the output of the recirculating buffer as a request for service for the address corresponding to the address of the start bit. Simultaneously, the character and the associated address are gated into the normal-read register. The start bit is transmitted with the data to signify that the transfer is legitimate. The overwrite bit is used to indicate when a character has stayed too long in the recirculating buffer and has been overwritten by the next character. This should not happen as long as the BC scan is frequent enough to keep up with the traffic. The BC must scan for service requests fast enough to read a character out of each of the 512 time slots of the recirculating buffer within a period less than the length of time necessary to assemble characters of the fastest code type. Thus, for a DSD connected to 512 ASCII 150 type lines, BC must scan for service requests at least once every 130 μs. The DSD responds to the normal-read mode by dumping the contents of the normal-read register onto the receive bus. This register is cleared by the receipt of an acknowledge signal from BC and is then ready to receive the next data transfer from the recirculating buffer.

The scan output register is used for the transfer of output service requests to BC. Access to the register is on a first-come first-served basis for any customer addresses requiring output service. The DSD automatically sets the output-service request bits for any active line when the output control count has proceeded through two characters and the third character has been gated to the assembler/disassembler unit for disassembly. An output-service request consists of the 9-bit line address for the line requiring service plus a bit in the service-request bit position. BC has one full character time to answer the output-service request to maintain continuous data flow. In the event that it takes longer than one character interval for the 3-character transmission to reach the output registers, the DSD will automatically produce a marking state on the customer line until BC responds with

the next three characters. Once again, the DSD must be scanned for output-service requests often enough to satisfy all 512 lines. One of the two output registers must be available before a service request can be issued but, because of the 3-character transmissions, less time is necessary to service output-service requests than input-service requests. In addition, output servicing can be temporarily deferred, if necessary, in order to devote more time to inputting when traffic is heavy.

Two output registers, identical in design, are used for the outgoing traffic in order to give the DSD more output capability. Since there is no synchronization between the appearance of the output-service request in the scan output register and the reply from BC, the data sent in response to the service request must wait in an output register before release to the RB. The output count must reach the proper level and the memory cycle must progress to an address match for the data to reach the proper time slot. Two output registers help to shorten the waiting interval as well as make more efficient use of the line facility bus time.

5.9 Duplex Synchronization

To achieve duplex operation of the DSD, all clock and address counters and all delay-line loops must be initialized in synchronism. Assuming that one unit is operating on-line, the initialization program resets and holds the counters and clears the delay lines of the off-line unit. This is done after first divorcing all matching circuits between the two units and disconnecting the off-line unit from the LTU. The held counters are released to continue counting when the counters of the on-line unit pass through the zero state. Both sets of counters are thus synchronized and are continuously matched to insure that they stay together. Delay-line synchronization is automatic as long as the high-frequency clock stages are synchronized but the data content of the memories must be allowed to update gradually. Output time slots can be updated by ceasing the flow of output data temporarily until all active customer lines can be restarted in the off-line unit and reach an equivalent all-marking state in both units. Input time slots are updated by a process known as forced start, wherein the control unit of the off-line unit is cross-coupled to the input start detect circuit of the on-line half so that input counts are established simultaneouly in both halves. After a fixed interval of time long enough (three character times at the lowest data rate) for the memories to equalize, duplex operation can be restored.

VI. MAINTENANCE FEATURES

6.1 *General*

DSD maintenance is complicated by the use of time-division techniques and by the restricted access to the delay-line memories. In addition, the large number of customer lines terminating in the DSD requires extensive converging and diverging networks which make per line fault-detecting difficult. Further complications arise from the duplex nature of the DSD and the necessity for keeping two on-line units in synchronism. To satisfy the maintenance requirements for the DSD, several special techniques had to be provided. These techniques were designed to enable rapid detection of faults and to allow reconfiguration and diagnosis with no interference to normal data processing.

The two major categories of DSD maintenance can be classified as:

- (*i*) External—that which applies to the interface circuits of each simplex.
  - (*a*) Line facility bus checks.
  - (*b*) Match checks between simplex halves.
- (*ii*) Internal—that which applies to the circuits contained within each simplex.
  - (*a*) Internal parity.
  - (*b*) Automatic disconnects.

One of the techniques used extensively throughout the DSD is known as the 1-hot (or 1-cold) check. This technique is used to monitor circuits where only one out of many similar inputs is to be active at a single instant of operation. The 1-hot check is particularly adapted to the many converging and diverging functions of the DSD. It is also efficient for monitoring-address translators and counter circuits. Special networks and detectors were developed to perform this function.

Another feature used quite extensively in the DSD involves the use of special registers and control-write instructions to direct the extraction of information from any of several points in the data-processing sequence. A total of 23 points in each unit may be interrogated by the control-write point procedure. In addition, use is made of the same access method to set or reset control flip-flops and clamp or hold counters for diagnostic examination. A total of 66 control-group instructions are used for these purposes.

The DSD has been designed to make it possible to repair either

unit without interfering with the operation of the other. Each unit has a completely independent power system and access by program for maintenance purposes. The primary functions such as oscillator, delay-line oven heaters, and customer-line terminals are supplied with ORed power to maintain continuous service. Use of the control group program modes to isolate a unit by divorce or disconnect aids in insuring independence. The DSD also contains eight test-line terminals, each dedicated to a different code type (two are duplicated); these terminals may be used as vehicles for test messages in the course of diagnosis of trouble. Finally, the DSD makes use of visual indicators such as the phase-lock meter for the oscillator and under-and-over temperature alarm circuits as well as lights for the ovens as aids in trouble diagnosis.

6.2 External Maintenance Checks

The maintenance techniques classified as external include those associated with line facility bus transmissions and those that match the status of one simplex unit against the other. Transmission checks are similar to those used in the No. 1 ESS and consist of odd parity, all-seems-well (ASW), and acknowledge functions. Cross-coupled match checks are made between on-line simplex halves to detect operational differences that will arise from a fault condition appearing in a simplex. These external matches are generally not capable of determining which unit is in trouble; when a match failure occurs, however, each unit detects the mismatch and inhibits its own ASW signal. In addition, ASW and acknowledge signals are cross-coupled to prevent either of the duplexed units from starting to process data before the other. Information obtained from internal maintenance circuits is used to localize the trouble. These internal checks cause an automatic disconnect of output from the faulty unit to prevent mutilation of data to the customer. Table III lists all external maintenance facilities and their functions.

6.3 Internal Maintenance Facilities

The maintenance techniques classified as internal include circuits which are self-contained in each unit and which generate and check internal parities, monitor counters and translators, and perform 1-hot checks on converging and diverging functions. Most internally detected troubles automatically disconnect the output of the faulty unit immediately to prevent mutilation of customer data. Circuits which

TABLE III—EXTERNAL MAINTENANCE

Send Bus Parity (Data) Control Bus Parity (Mode and Address) Stored Parity (Name Code) All Seems Well (ASW) Address Error Bit (ADE) Acknowledge (ACK) Mode Decoder 1-Hot (1/16)	Fault Recognition in Communications
Duplicate Bus Transmission Send and Receive Bus Selection Control Modes	Communications Diagnosis
Forced Start Cross-Match Type Bus Cross-Match Sample Bus Cross-Match Output (Set-Reset) Bus Cross-Match Output Register Cross-Match Address Counter Cross-Match Acknowledge Cross-Match ASW Cross-Match	Duplex Fault Recognition
Monitor Bus Scan Points Central Pulse Distributor	DSD Status Tests

can take such drastic action are limited to those that can determine without outside reference which unit is bad. An input disconnect is never applied automatically; nor can a unit disconnect its output if the other unit is already disconnected. Each unit has a disconnect inhibit to prevent such an occurrence. Control write instructions and control group orders utilize special registers and flip-flops to set up and examine internal circuits of each unit and are therefore classified as internal maintenance.

As a check against the operation of the delay-line loops of the assembler/disassembler and recirculating-buffer units and associated gates and registers, an internal parity is generated and carried through in both input and output directions for every character processed. These internal parities are both odd and are independent of bus parities. Table IV lists the internal-maintenance facilities and their functions:

6.4 One-Hot Monitors

The term 1-hot applies to a monitor that looks for one low-level signal out of a group of high-level signals. Conversely, a 1-cold monitor

TABLE IV—INTERNAL MAINTENANCE

Input Data Parity Output Data Parity High-Frequency Toggle Match	Delay Line Fault Recognition
Type Bus (1/192) Type Code (1/6) Type-Sum Gates (1/13)	1-Hot Monitors
Sample Bus (1/32) Set-Reset Bus (1/64) Horiz. Address Trans. (L&R) (1/16) Vert. Address Trans. (L&R) (1/16)	For Logic Diagnostic
Trouble Flip-Flops Divorce Flip-Flops Disconnect Flip-Flops Disconnect-Inhibit Flip-Flops R0, S0, S1 Flip-Flops	Configuration Control
Oscillator Level Detectors Phase Lock Meter Address Counter (L-R) Match	Timing Fault Recognition
Output Maintenance Register Input Maintenance Register Error Status Register Address Error Register	Programmed Maintenance Access and Communication

looks for one high-level signal out of many low-level signals. In each case, a multidiode network connects the entire group to be monitored to the input of a detector. The diode networks for use in 1-hot applications are poled oppositely to those for use in the 1-cold applications. In this way, the same detector design may be used for both cases, and similar voltage levels may be used at the input to the detector to distinguish between normal and alarm conditions. Figure 15 shows a chart of the various input-voltage levels versus output-logic levels for the normal condition (1-hot) and the alarm conditions (no-hot, two or more hot). Conditions for the 1-cold case are also shown. The logic states of the detector outputs, A and B, are used to control the alarm logic. Normally quiescent, as long as a 1-hot (or 1-cold) condition prevails during a given time interval such as an address interval, the logic will set a cell in the error status register if an alarm state occurs. The output of the detector is strobed by a timing pulse to prevent false setting of cells during transitions between intervals. An input-voltage level within 6 to 12 volts dc is interpreted by the detector as normal, in either the 1-hot or 1-cold case, and the logic circuitry takes no action during the interval. The
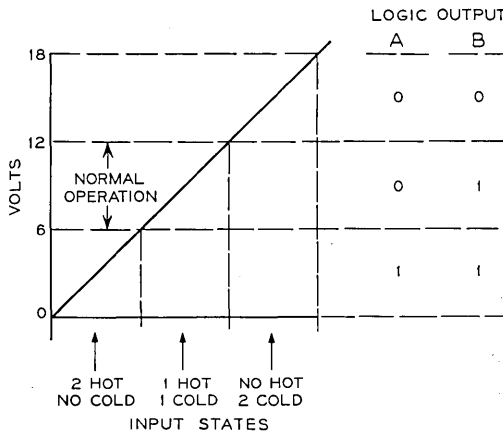
Fig. 15—Hi-Lo detector output states.

normal voltage level is established by the diode network as long as current is drawn through only one diode path in the network as a result of the 1-hot (or 1-cold) situation. However, should no diodes, or more than one diode, conduct current the voltage at the detector changes accordingly, and the output of the detector assumes one of the alarm states.

6.5 *Maintenance Registers*

Three of the major logic registers not described previously are used exclusively for maintenance and diagnostic purposes. Listed in Table IV are the three major maintenance registers which in turn accept maintenance instructions into the DSD, transfer data resulting from these instructions back to BC, and store status reports for transfer to BC.

The output maintenance register stores control write instructions and control group orders sent to it by BC over regular data bit positions of the send bus under the control mode of operation. Once stored in the output-maintenance register, the data is translated into an instruction or order which in conjunction with a match of the accompanying line address, and in some cases, a control unit count, performs operations as follows:

(i) Directs a parallel word readout from any one of several stages in the DSD input or output process including delay-line loops of the assembler/disassembler unit, recirculating-buffer unit, and control unit, and other registers and gates of importance

(*ii*) Sets or resets specified flip-flops for such purposes as bus configuration, divorce, disconnect, and maintenance detector tests.

Readout operations directed by the output-maintenance register store data in the input-maintenance register which then requires a control read to complete the transfer to BC. The data is stored in the input-maintenance register at the instant of the address match and if required, at the instant of the count match specified by the contents of the output-maintenance register. An address match occurs once every memory cycle (606.06 μs) but storage can be delayed for any multiple of memory cycles by requiring that a control-count match coincide with the address match. Thus, any point within the assembly or disassembly process can be examined to the closest integral memory cycle of any address.

The input-maintenance register stores data and address bits representing the answer to a control write instruction directed to the DSD by BC. One cell of this register records the fact that data has been gated into the input-maintenance register and is ready for transfer to BC. The output of the cell is monitored by a ferrod which operates to inform the system that the data is ready for interrogation and transfer to BC. A subsequent control-read mode is used to release the contents of the input-maintenance register to the receive bus. After the input-maintenance register has performed the readout to the receive bus, it is cleared by an acknowledge from BC.

The error-status register stores the trouble reports from maintenance circuits within the DSD unit. There are 22 specific cells in the error-status register. Each cell monitors a particular function at periodic intervals determined by timing pulses arising from the mode transmission or generated by the internal clock. When a failure is detected, the error-status cell associated with the circuit in trouble is set. Some, but not all, of the 22 cells inhibit the ASW for the associated unit so that the system will be informed quickly in case of trouble in the DSD; in the other cases, a slower method of reporting to the system exists in the form of a summary ferrod activated when any one of the 22 cells is set. The error-status register must be interrogated by a control read mode before it transfers its contents to BC via the receive bus. The error-status register is reset by CPD under normal circumstances although a maintenance-program reset is available for diagnostic purposes.

A fourth register, of considerable importance to diagnostics, the address-error register, is used to store the A and B bits from 1-hot

monitors and the address (6 bits) interval during which a 1-hot detector exhibits an alarm state. This register, which does not have a direct-access path to the bus system as do the other three buses, communicates via the input maintenance register.

VII. SUMMARY

A data-scanner distributor has been designed that achieves performance and low cost-per-line objectives. The extensive use of No. 1 ESS hardware resulted in a design with the reliability and maintainability consistent with No. 1 ESS.[4] The unit is designed for 512 low-speed FDX data subscriber line connections.

A very compact design was achieved by the use of ultrasonic delay lines in a highly time-shared fashion. The unit is fully duplicated utilizing 116 delay lines in the two units. By scanning every input bit many times under control of the high-speed time-shared counter, the DSD can accept highly distorted input data as well as transmit essentially distortionless data.

Maintenance circuitry allows detailed diagnosis of the DSD by CC programs. Also, the DSD possesses certain internal-check circuits providing the DSD the ability to divorce a faulty unit in case one of them fails these checks. This function is provided to prevent garbled subscriber messages when hardware failures occur.

VIII. ACKNOWLEDGMENTS

REFERENCES

1. Rea, W. T., "Effect of Telegraph Distortion on the Margins of Operation of Start-Stop Receivers," B.S.T.J., *33*, No. 3 (July 1944), pp. 207–233.
2. Connell, J. B., Hussey, L. W., and Ketchledge, R. W., "No. 1 ESS Bus System," B.S.T.J., *43*, No. 5, Part 1 (September 1964), pp. 2021–2054.
3. Meitzler, A. H., "Ultrasonic Delay Lines for Digital Data Storage," IRE Trans. on Ultrasonic Eng., *UE-9*, No. 12 (December 1962), pp. 30–37.
4. "No. 1 Electronic Switching System," B.S.T.J., *43*, No. 5 (September 1964), pp. 1831–2609.

No. 1 ESS ADF:

Message Store—A Disk Memory System

By E. J. AITCHESON, C. F. AULT and R. G. SPENCER

A high thruput, 60 megabit, fixed head per track disk memory system was designed to meet the bulk message store requirements for the No. 1 ESS ADF System. The application in this store and forward message switcher demanded that the disk store meet the stringent reliability requirements consistent with those of an electronic switching system. The article describes the system design features, a temperature-compensated read/write scheme, a synchronizing motor servo system, and automatic maintenance techniques.

I. INTRODUCTION

An essential element of the No. 1 ESS ADF store and forward data handling system is an economical, nondestructive readout, bulk memory with an average thruput of 153 kb/s. Core and ferrite sheet memories meet the access requirement, but their costs and destructive readout make them less attractive than the slower access disk or drum large capacity memories.[1]

In a study conducted to select a memory file that would best fit the ADF application, the following requirements were established for the memory:

(*i*) present commercial production and indication of continuing supply over the years,

(*ii*) performance over the expected 40-year life of the system with minimum downtime and repair cost,

(*iii*) fixed head/track access with a minimum of mechanical equipment, and

(*iv*) capacity of at least 50 megabits.

The file selected, a Burroughs Corporation model BC475 Disk File, has a fixed head/track access scheme, a capacity of 60 megabits, and an average access time of 20 milliseconds (1/2 of a revolution).
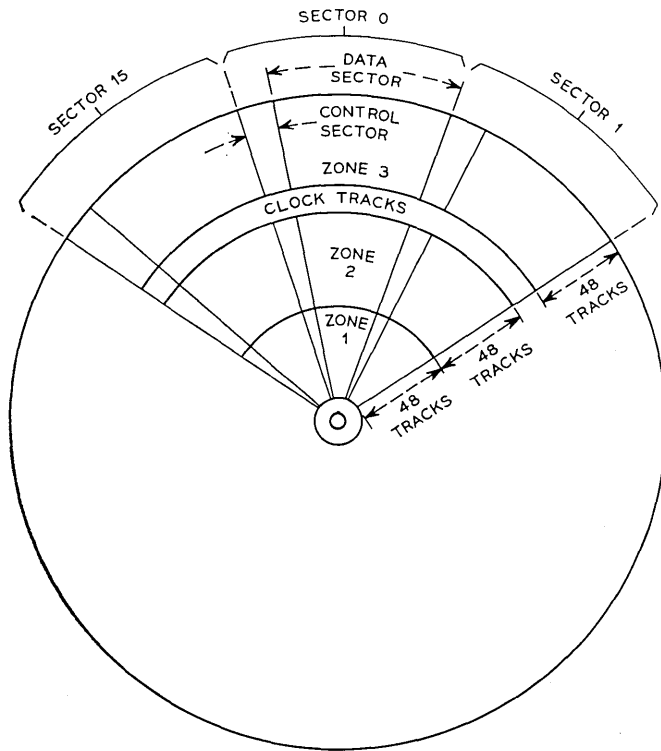
Fig. 1—Disk surface geometry as determined by word clock.

The average message thruput requirement for the No. 1 ESS ADF application was obtained by partitioning the disk faces into 16 pie-shaped sections (Fig. 1) and by the application of queueing techniques. Thus, during a given disk revolution, a maximum of 16 blocks by 32 words/block by 24 bits/word can be transferred to or from any given disk address providing a maximum thruput of 308 kb/s.

The disk file is incorporated in the message store subsystem, which consists of a logic controller, access circuits, synchronizing servo system, and the 60-megabit disk file. The logic controller, access circuits, and servo were designed at Bell Telephone Laboratories and manufactured by the Western Electric Company. Important design parameters are tabulated in Table I. The controller was designed with No. 1 ESS technology and used available circuit packs, wherever possible; thus, the unit meets No. 1 ESS circuit reliability objectives.[2] The

design also includes self-checking maintenance hardware and sufficient access to allow program-controlled maintenance testing consistent with the ESS philosophy.[3] Maintenance programs executed from the common-control ADF central processor detect and isolate faulty components to within three replaceable, plug-in circuit packs (Fig. 2).

Desired reliability and message protection are achieved by fully duplicating the message store subsystem and running the two halves in a synchronous match mode. Duplicate communication buses are provided so that the units can be kept in step and configured in a few microseconds, by program control, in the presence of a fault. This provides maximum protection against mutilation or loss of call data.

II. DISK FILE

The file, motor drive, and head pressure system are manufactured by the Burroughs Corporation. Each file contains four 26-inch diameter brass disks (eight memory faces), plated with a thin film of nickel cobalt. The disks are belt driven on a common shaft from a central pedestal position. To prevent contamination, the disks and read/write heads are sealed in a dust-tight enclosure maintained at a pressure of one inch of water by an external blower-filter arrangement. The disk file incorporates a fixed head/track access matrix with the heads held in near proximity to the surface by a pneumatically driven head piston working against the counter force of a laminar air cushion and retracting spring.

The file and dust-sealed enclosure are mounted in an environmentally controlled temperature chamber held at an elevated temperature with respect to the office ambient. This provides file stability over the office operating range of 32° to 120°F.

TABLE I—MESSAGE STORE CHARACTERISTICS

Controller power	840W @ +24VDC, 960W @ -48VDC
Motor inverter power input	1300 W
Motor speed	1725 rpm
Motor type	2 hp, 208 volts,
	3 ϕ (inverter powered)
Data packing density	1100 bits/inch
Type of recording	NRZ
Bits/file	60 megabits
Writing surfaces	8 (4 disks)
	1/8" thick 26" diameter
Number data tracks	1152
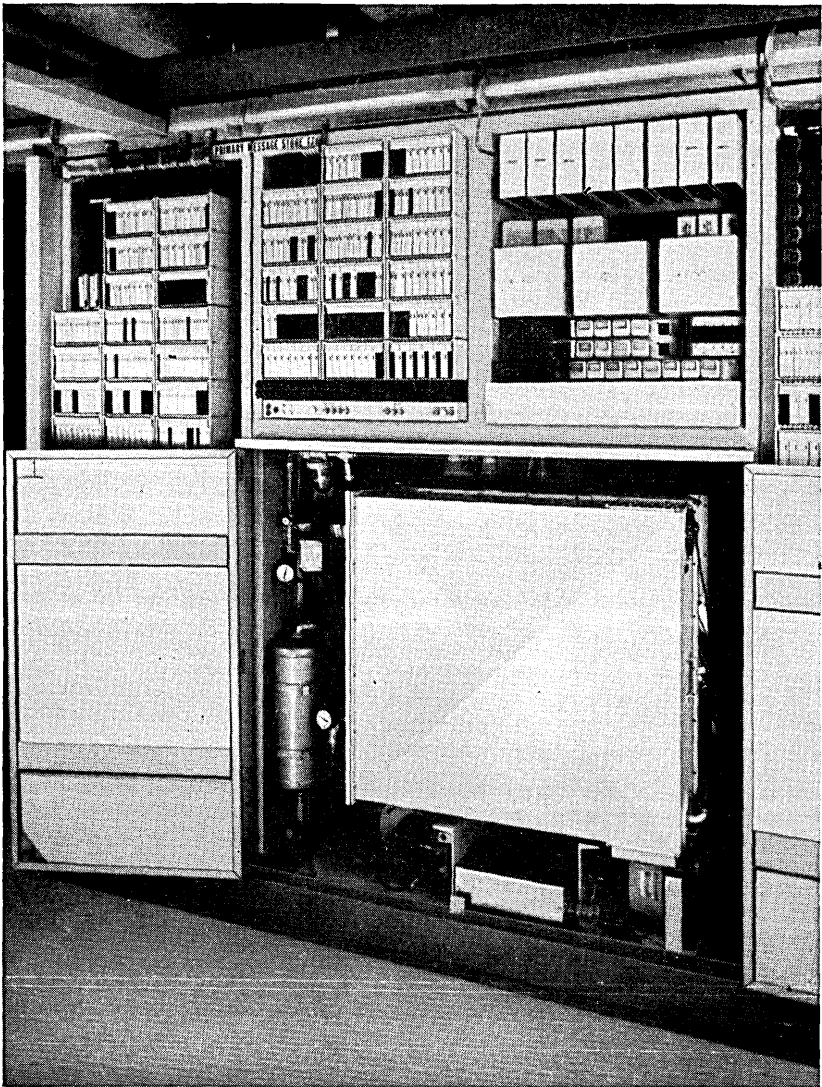
Fig. 2—Message store.

2.1 *Nitrogen System*

The disk heads are flown close to the memory surface by a closed pneumatic gas system. The gas is provided from a bottled nitrogen system mounted remote from the logic frame. The nitrogen system is duplicated with each simplex system, providing working pressure to a simplex half of a duplicated disk. Thus, a single failure in the nitrogen system will not cause a complete loss of disk facilities in either community. The pressure is reduced from the bottle pressure to the required 40 psi by a series of three regulators. The use of multiple regulators provides over-pressure protection to the file. Monitor alarms are provided to allow program detection of out-of-limits gas pressure.

III. SYSTEM INTERFACE

The central processor communicates with the message store via the high-speed buffer control call store memory and a wired logic buffer control (Fig. 3). This arrangement allows the processor to transfer
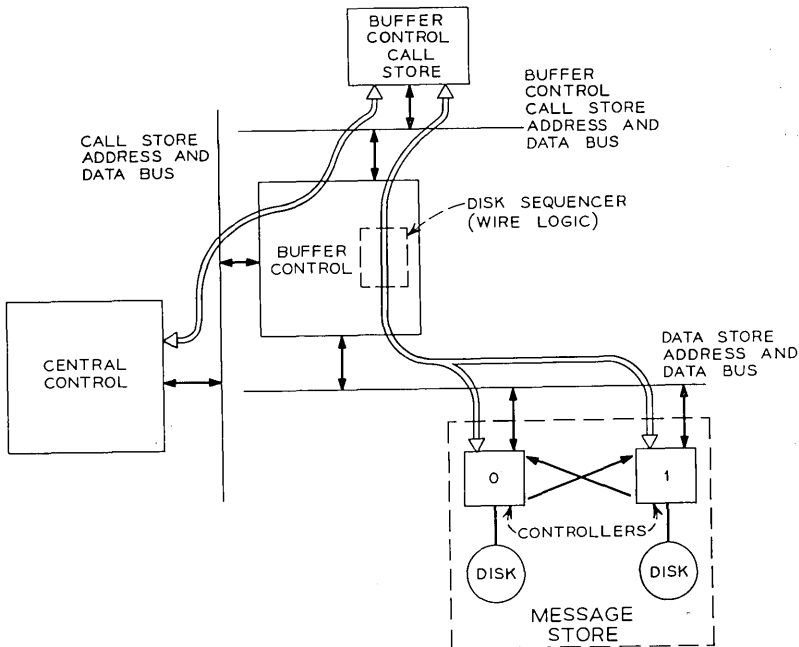
Fig. 3—No. 1 ESS ADF message store system interface.

disk data to the buffer control call store at a 5.5 microsecond/word rate. The wired logic sequencer, in the buffer control, then transfers the disk data from the buffer control call store to the primary message store at the 80 microsecond/word disk rate. Thus, the processor is released to continue its real-time work and not required to scan the message store for its asynchronous requests on a real-time basis.

The wired logic disk sequencer is one of the four peripheral sequencers, centrally located in the buffer control, which time-shares the common buffer control call store memory. A wired logic bus sequencer handles all requests for use of the buffer control call store bus by the common peripheral sequencers, and grants priority in a predetermined order.

The disk speed and phase are asynchronous to the fixed buffer control processing cycle and require the message store to issue commands when it is prepared to read or write the disk. The following requests are sent to the buffer control disk sequencer by private requests leads:

(i) *Instruction Request*—Message store ready to receive an instruction from buffer control for the current sector.

(ii) *Data Request*—Message store ready to receive a 24-bit data word (write mode) or requesting its 24-bit data register to be read (read mode) by buffer control.

(iii) *Status Request*—Message store requesting the buffer control to read its error status register to determine error history over the last block of 32 data words.

These requests are generated in the primary message store by decoding the *word* clock which is permanently magnetized on the disk surface. A *bit* clock provides a train of pulses to serially shift the contents of the 24-bit primary message store data register, one bit at a time, onto the disk.

A maximum of 12,288 bits of data (16 sectors/revolution by 32 words/sector by 24 bits/word) can be written or read during one revolution of the disk (40 milliseconds).

IV. DATA ORGANIZATION

To maintain a nearly uniform packing density of data on the disk surface, a disk is divided into three zones, as shown in Fig. 1. Each zone has its own clock-determined bit rate. Zone 1 has a bit frequency of 1.1 megabits/second, zone 2 has a bit frequency of 1.5 megabits/sec-

ond, and zone 3 has a bit frequency of 1.9 megabits/second. These three frequencies provide a nominal packing density of data on the disk surface of approximately 1,100 bits per inch. The disk surfaces are further segmented into 16 sectors. Each of the 16 sectors is composed of a *control* sector followed by a *data* sector.

The data portion of the sector is a block of 32 words written in an interlaced arrangement, as shown in Fig. 4. Zone 1 has three blocks interlaced, zone 2 has four blocks interlaced, and zone 3 has five blocks interlaced. Hence, the first, fourth, seventh, etc., words form a block of data in zone 1. Only one interlaced block may be written or read during a given sector and zone. This interlacing of data provides a sufficiently low data rate on the data store bus so that it may be shared simultaneously among many units. It also allows the duplicated and synchronized message store to read and write data at slightly different times and then, by individual time buffering, perform bus operations simultaneously.

4.1 *Clock and Timing Generation*

Each zone has a *bit* clock and a *word* clock used in reading and
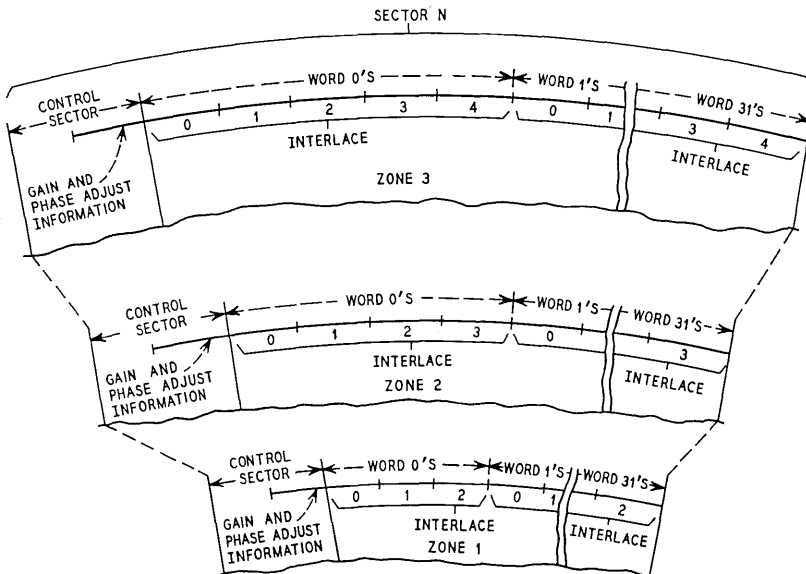
Fig. 4—Disk control and interlace arrangement as determined from word clock.

writing of data in that zone. The bit clock defines the bit positions within a word. The word clock provides word, interlace, and miscellaneous control information. The information in the word clock is coded in a binary, serial manner. This serial bit stream is shifted through a 9-bit shift register and periodically decoded with *decision time* pulses to obtain the disk logic timing (see Fig. 5). The decision time pulse is derived from a logical combination of the bit and the word clock. Normally, every pulse in the word clock has a corresponding pulse in the bit clock. Periodically, however, a pulse appears in the word clock with no corresponding pulse in the bit clock. This condition generates the decision time pulse, which causes decoding of the present contents of the shift register. The disk word clock generates ten timing marks in this manner.

The timing marks thus generated define interlace boundaries, start of sector, end of sector, and necessary timing pulses in the control sector for communications with the processor. In addition to generating the timing mark at a decision time, some timing marks cause gating
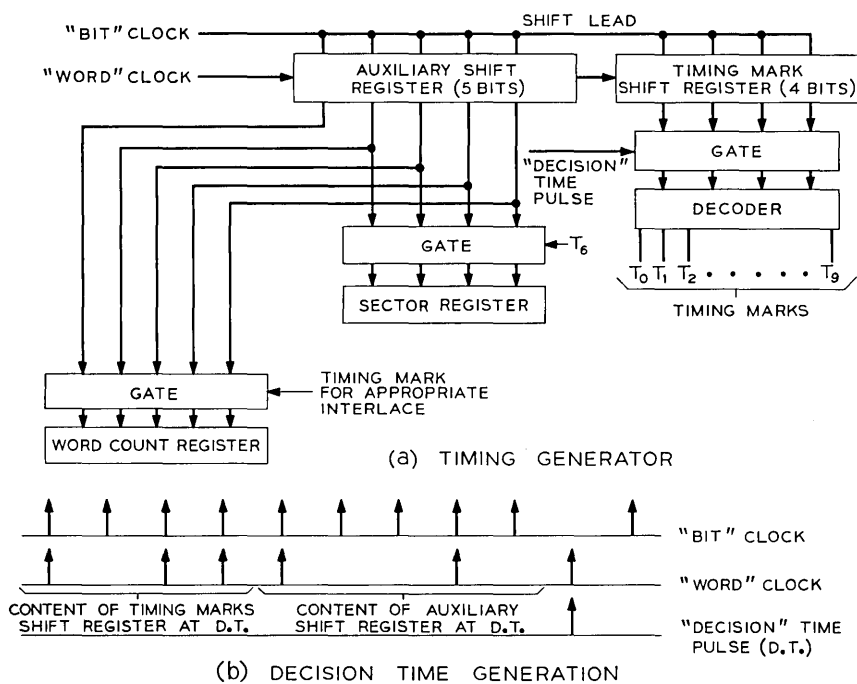
Fig. 5—Relationship of disk word and bit clocks.

of information in the auxiliary shift register to special purpose registers (see Fig. 5). In this manner positional information such as sector and word number is available without counting time marks. The clock information is written in the clock tracks by a special apparatus previous to the installation of the disk file into the message store. Once there, the clocks are write protected and can only be read by the disk logic circuits.

V. SERVO SYSTEM

The disk stores are operated in pairs, so if one should fail, the other will carry on. The two disks are servoed to permit matching of their outputs while they are being read and to permit simultaneous storage of messages.

Each disk reads its sector location from the *word* clock track. The 4-bit sector address is cross-coupled to its duplicated mate, and each disk compares its address with its mate's address. One will be slow, the other fast. The sign of the error controls the frequency of the servo oscillator, causing the disk to speed up or to slow down. The speed is controlled by a 360-Hz voltage-controlled oscillator. The oscillator is counted down to control the three-phase 60-Hz power for the motor.

All power for the message store is derived from the office battery. The disk motor operates from 208 volts, 3-phase, 60-Hz. A transistorized inverter, controlled by the counted down 360-Hz oscillator, inverts the battery voltage to the 208 volts required for the motor. The motor rotates nominally at 1725 rpm and is belt-coupled to the disk shaft. The disk speed is reduced to 1500 rpm by the belt pulley combination.

The following is a list of some of the pertinent design requirements which the *servo system* must possess:

(*i*) fully automatic,

(*ii*) no master-slave relationship,

(*iii*) all servo positional and frequency data derived from the normal clock signals, and

(*iv*) four-microsecond maximum phase error between servoed disks. (Forty milliseconds per revolution.)

Since a small phase error is required, a high phase gain is necessary. To make the system stable with high gains, rate feedback is required. The rate feedback could be derived by comparing the speed of two disks; instead, it is derived by comparing the disk clock to an internal
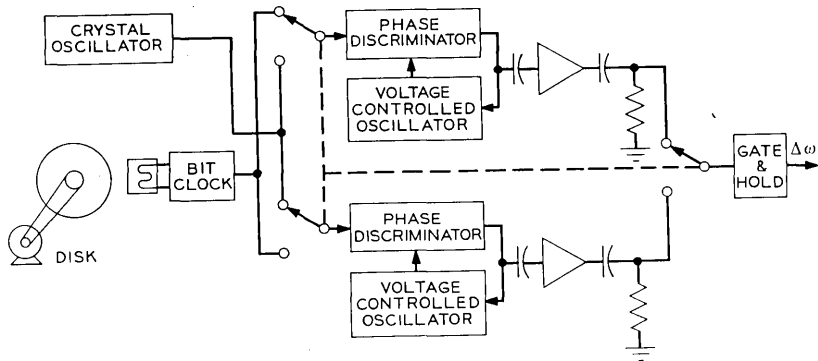
Fig. 6—Frequency comparator.

crystal oscillator. This eliminates cross-coupling frequency informa-
tion and causes each disk to rotate with the same accuracy as its
crystal. The frequency servo is shown in Fig. 6.

There are two paths through the servo circuit, phase and fre-
quency. The phase error, leading or lagging, is derived from the
comparison of the local disk address with the remote disk address.
The result of this comparison is stored in a flip-flop which, if set,
speeds up the oscillator, and if reset, slows down the oscillator. The
oscillator controls the speed of the motor, hence, that of the disk.
The position of the disk is repeatedly read and compared with the
other disk, thus closing the loop (see Fig. 7).

In the second loop, the *bit* clock on the disk is compared with a
reference crystal. The comparison results in an output which will
speed up the oscillator if the bit clock is at a lower frequency than
the crystal or will slow down the oscillator if the bit clock is at a
higher frequency.
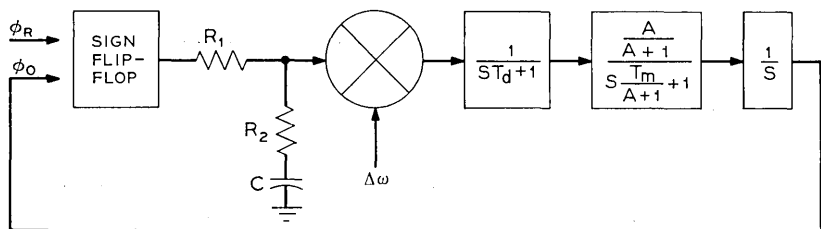
SIGN FLIP-FLOP RESPONDS TO SIGN OF DIFFERENCE IN PHASE
LAG NETWORK R_1, R_2 & C SHAPES PHASE ERROR

Fig. 7—Complete sign only servo.

Again, the oscillator output controls the speed of the motor, therefore, the frequency of the bit clock. The frequency loop always opposes the phase loop or any other force trying to change the speed of the disk.

Considering initial response, the oscillator, inverter, and disk can be represented by a simple time constant T_m.

$$T_m = \frac{I_m}{K_m} \quad \begin{array}{l} \text{(total system inertia)} \\ \text{(slope torque-slip curve)}. \end{array}$$

Figure 8 shows a servo structure without rate feedback. This simple servo would not meet the phase accuracy objectives.

To meet the phase accuracy objective, the phase loop gain must be high, which will cause the circuit to oscillate. Rate feedback is added to stabilize the circuit but must also meet severe requirements. Some of the problems are a result of the methods of powering the disks. Each phase of the power inverter produces a square wave output and is Y connected to produce the 3-phase power for the motor. Because the oscillator frequency is not changed until a transition occurs in the square wave output, a 0 to 2.77-ms delay is introduced into the loop. There are other small delays which are lumped into one time constant T_d.

Figure 9 shows rate or frequency feedback added to the servo loop illustrated in Fig. 8. For $A \gg 1$ and $T_m \gg T_d$ and $T_m/A \gg T_d$, an appropriate equivalent circuit can be derived. Starting with a 2-pole system, the closed loop response is:

$$(F_s)/(1 + F_s) = (A/[(ST_m + 1)(ST_d + 1)])$$

$$\div (1 + [A/(ST_m + 1)(ST_d + 1)])$$

$$(F_s)/(1 + F_s) \approx (1/\{[(ST_m)/A] + 1\}) \times (1/(ST_d + 1)).$$

Figure 10 shows the equivalent circuit as suggested by these equations.

Instead of proportional phase error, the actual servo uses sign only feedback, equivalent to infinite gain. For the slightest possible phase error, full output is obtained from the sign flip-flop. To reduce the frequency loop gain to a stable level, a lag network is added in the phase loop. The final servo design is shown in Fig. 11. Having infinite gain, the servo will continually oscillate with a predictable small phase error.

The frequency loop has two severe requirements: it must introduce

α = PHASE GAIN
ϕ_R = PHASE OF REMOTE DISK
ϕ_O = PHASE OF OWN DISK
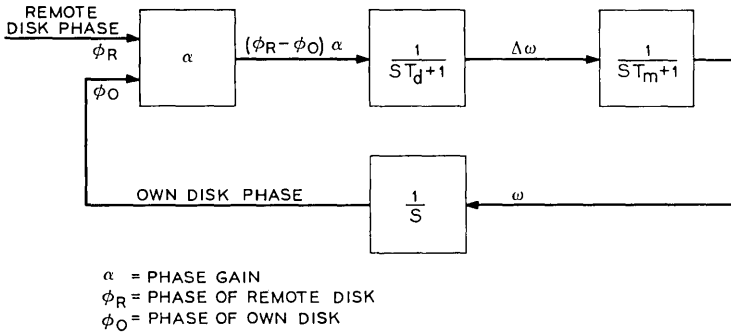
Fig. 8—Servo without rate feedback.

as small an open loop offset as possible, and it must derive the rate feedback with as small a delay as possible. For a gain A, of 100 and $T_d > 7.5$ ms, the frequency loop alone is unstable. T_d includes the 0 to 2.77-ms delay caused by the square wave drive to the motor. The requirements are an offset less than one part in 2×10^4, and effective delay of one millisecond or less, and a gain of 100.

The disk clock (1.1 MHz) is compared to a crystal oscillator. In one millisecond there are 1100 transitions, but the required accuracy is one part in 20,000. Thus, a counting type measurement of the relative frequency of the disk and clock is not practical. The comparison circuit must give high accuracy, low offset, and be immune to noise or variation in bit clock timing of single bits. The circuit developed uses a phase-locked oscillator as a frequency measuring device. An

α = PHASE GAIN
A = FREQUENCY GAIN
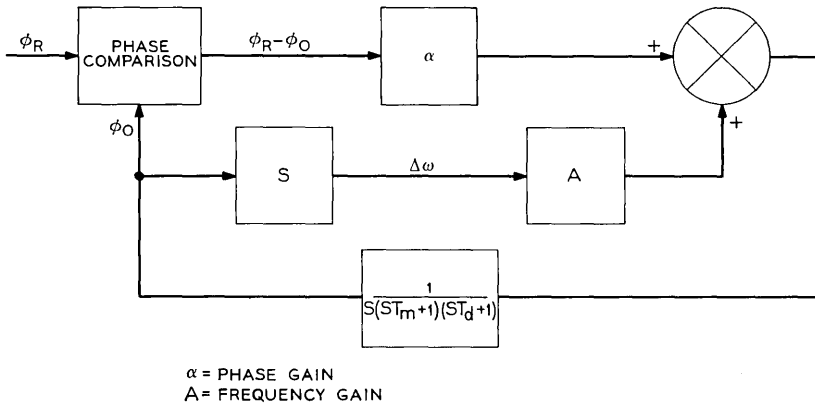
Fig. 9—Servo with rate feedback.

RATE FEEDBACK DECREASES MOTOR-DISK TIME
CONSTANT AND DECREASES DISTURBANCES
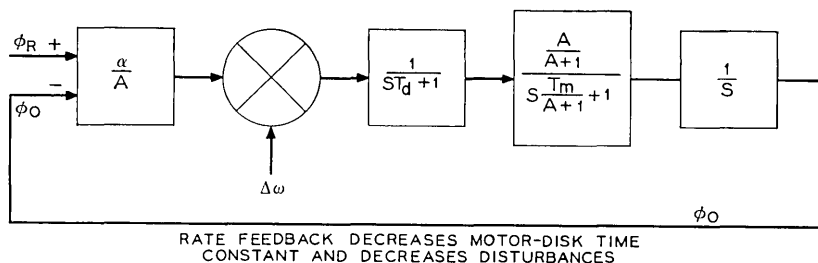
Fig. 10—Effect of rate feedback on servo.

RC oscillator is voltage controlled by the output of a phase discriminator.

This voltage-controlled oscillator is phase locked first to the local crystal, then to the disk clock. The change in the oscillator control voltage is proportional to the difference in frequency of the disk clock and the crystal. Since only the change in voltage is used, there is no offset error caused by changes in the voltage-controlled oscillator components or supply voltages. The change in voltage is amplified and rectified by a phase sensitive circuit. To provide a continuous output proportional to disk speed, two circuits are used. One circuit is locked to the crystal oscillator while the other is locked to the disk clock, thus producing the desired output.

Figure 6 is a simplified schematic of the frequency compensation circuit. The voltage-controlled oscillator can have any phase relationship when switched from crystal to disk or vice-versa. This large phase error appears as a pulse on the control voltage lead. This pulse is gated from a holding circuit so it cannot upset the actual error signal derived from the control voltage.

Figure 11 shows a two-disk system and the servo circuit. When first turned on, the two disks may have any phase relationship and differing speeds. For large phase errors, the error voltage is increased to decrease the time necessary to approach the same phase. Large phase error signals are derived from comparison of the more significant bits of the address. When the phase error is small, the comparison is switched to least significant bits, and the error voltage is decreased. All sequencing is automatic.

When the disks are servoed, the final speed is proportional to the average of the two crystal frequencies if there were no other offsets. Since there are other offsets, the final frequency is near the crystal

FROM OTHER DISK

INCOMING
PHASE
(POSITION)

LOCAL
PHASE
REFERENCE

LOCAL
FREQUENCY   ──── CRYSTAL

READ
OUT

| PHASE COMPARISON | FREQUENCY COMPARISON |

SUMMER

ERROR VOLTAGE = EV

VOLTAGE CONTROLLED
360 cps OSCILLATOR

DISK

NOMINALLY
360 ∼

VARIABLE
FREQUENCY
DRIVE

INVERTER
dc TO ac

MOTOR
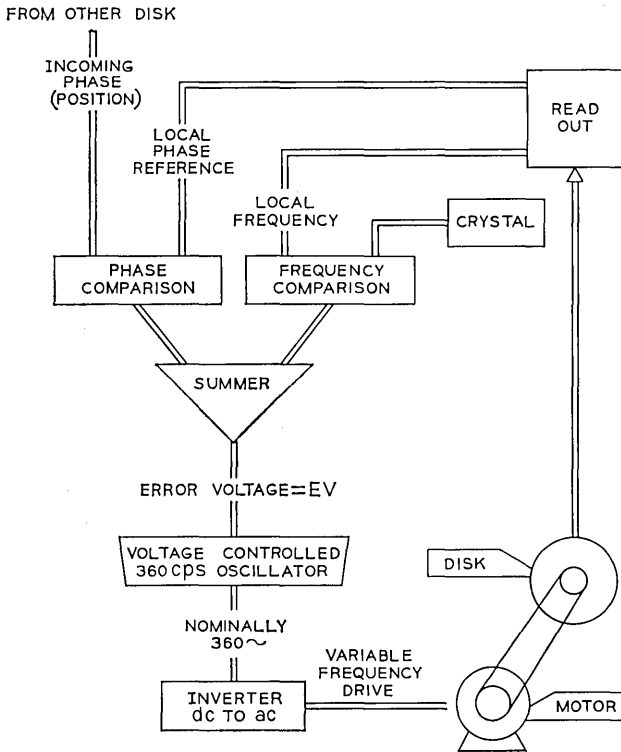
Fig. 11—Basic servo system.

frequencies but is not necessarily the average. If one servo stops working, and the disk continues to rotate, the other disk will stay phase locked to the unservoed disk. By deriving the phase error from the sector address and the rate feedback from the bit clock, the disk is affected with the same clock requirements as it would be without the servo.

The disks have been servoed for periods of months with continuous recording of phase errors with no error in excess of four microseconds.

VI. READ/WRITE CIRCUITS

Accessing a track to read or write is accomplished by selecting a head in the 96 by 12 head matrix (Fig. 12). Figure 13 shows a typical head group and its associated selection circuitry. In the writing mode, the emitter of Q2 is raised +12 volts to select the desired center tap.

A current regulator, Q5, is switched on, causing 130 milliamperes to flow in the ground clamp, Q6. Then the write switches for the select head group are enabled to write the data pattern.

The recording method used on the disk file is nonreturn-to-zero, where the disk surface is magnetically saturated in one direction for a "1" and the other direction for a "0". This recording method is schematically shown in Fig. 14. If switches Q3 and Q4 of Fig. 13 are alternately switched on and off, the 130 milliamperes will flow in one side of the head or the other. The fringing field in the gap region of the head (Fig. 14) will magnetize the disk surface in relation to the direction of current flow in the head windings.

While signals are being read from the disk surface, the write switches and current regulator are turned off. One of the 96 center taps is returned to +12 volts through transistor Q2, and read switch, and transistor Q7 is grounded, to access the desired head group. This causes a small current to flow from the center tap of a select head through the 1K ohms read-select resistors to ground. This forward biases all diodes between the desired head and the read amplifier, providing the low impedance path for the millivolt head signals to reach the amplifier. Because of the heads' poor low-frequency response, the read signals are differentiated pulses whose peaks correspond to the magnetic transitions on the disk surface. These signals are amplified and detected to obtain the nonreturn-to-zero representation.
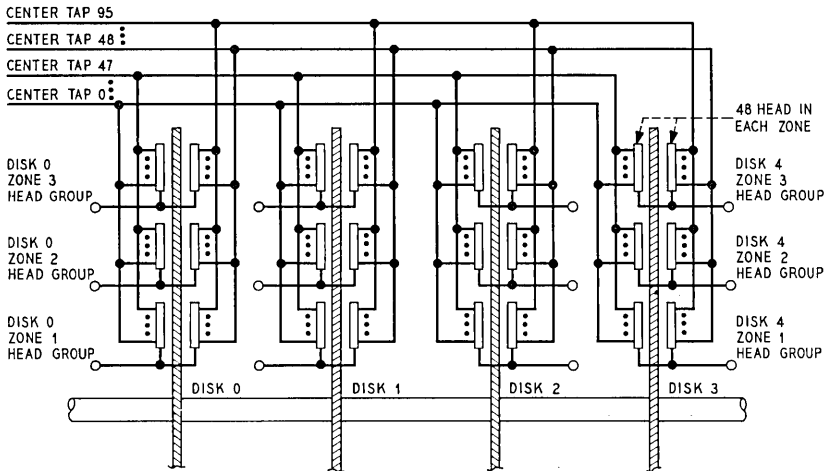
Fig. 12—Data track selection in a fixed head/track disk store.
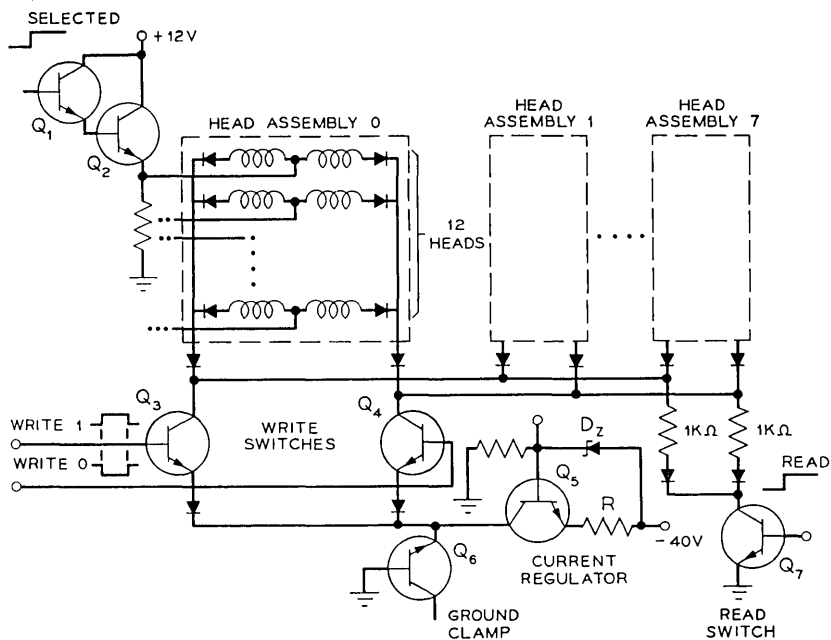
Fig. 13—Head selection read/write.
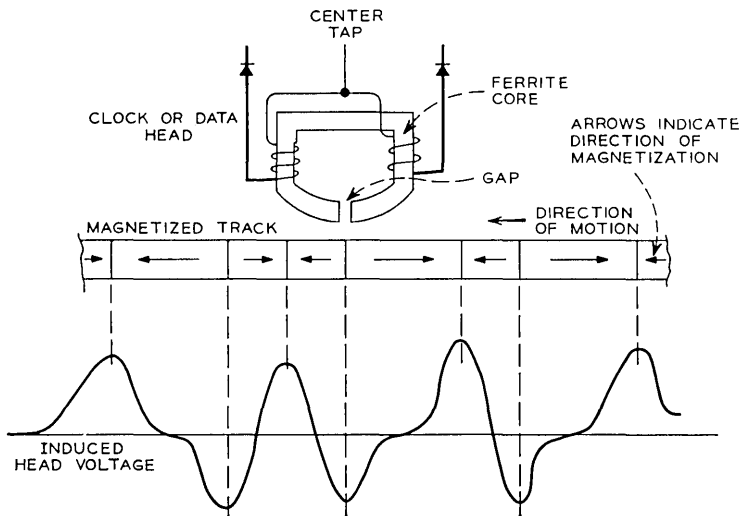
Fig. 14—NRZ recording.

6.1 *Disk Readout Detector*

In the readout detector of Fig. 15a, the analog signals are converted to a digital pattern similar to the nonreturn-to-zero writing current. Because peaks of the analog signal most accurately define the flux transitions of the disk surface, the peak detector is an important part of the detector. When the voltage level of the signal exceeds a fixed threshold, the threshold detector's output goes to a "1" state. When the peak detector senses the peak, its output becomes "1" and provides the necessary conditions to activate the following NAND gate. The output of the NAND gate then sets or resets the flip-flop depending upon which output is being considered. Hence, the data
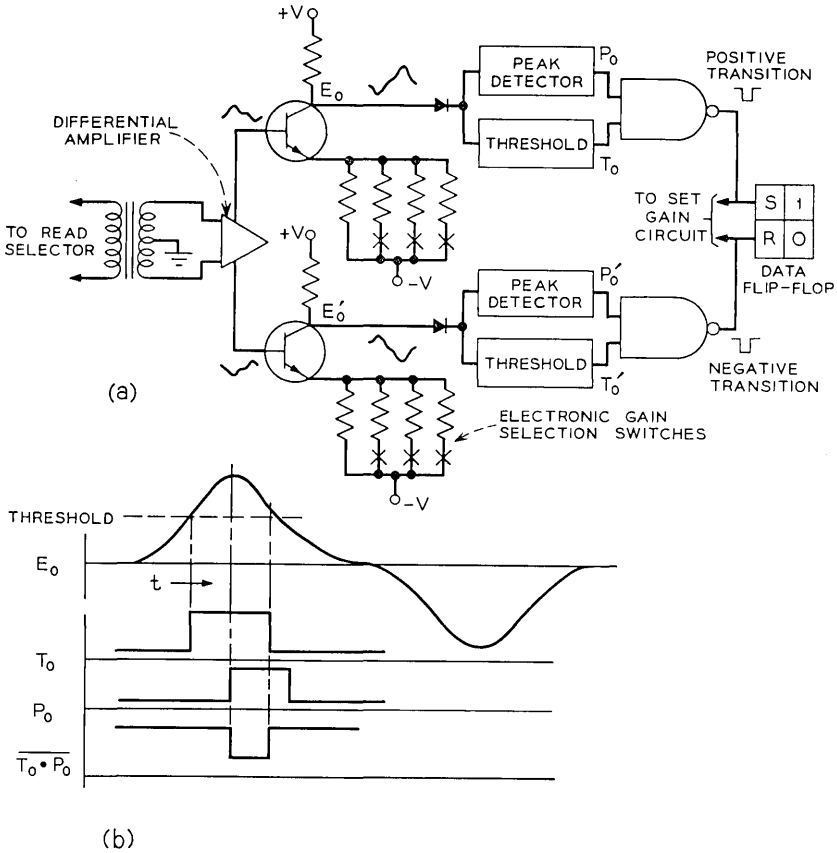
Fig. 15—Readout detection scheme: (a) detector circuits, and (b) detector circuit wave forms.

flip-flop is set for a positive peak and reset for a negative peak (see Fig. 15b for timing). The resultant output of the data cell is a non-return-to-zero representation of the data on the disk surface. This output must then be shifted, bit by bit, into the data shift register with the *bit* clock until an entire serial word is assembled and is available in parallel form.

6.2 *Adaptive Gain Circuit*

To accommodate the 5-to-1 signal amplitude variations from the disk heads, an adaptive gain amplifier is used in the readout circuit. The final stage of the readout amplifier incorporates a fixed collector resistance with a switchable emitter resistance which can vary the final stage gain from 2.5 to 10.7.

In the *control* portion of each sector on every track, there are 21 permanent check bits (see Fig. 16). These bits characterize the signal in the following data portion of that sector. They provide the phase and gain information necessary to read that sector track address accurately. The first two transitions of the 21 check bits are used in setting the gain of the amplifier. Approximately 60 microseconds preceding the gain check bits, the head access circuitry selects the desired head for the sector to be read. The 60-microsecond period is required for the amplifier to recover from the large head switching transient. The initial amplifier gain setting is such that the amplifier

Fig. 16—Adaptive write/read strobe timing relationship.

side used in detecting negative peaks has a final stage gain of 2.5, and the other side has a gain of 3.7. While the two check bits are passing through the detector, both the positive transition and the negative transition outputs are monitored (see Fig. 15a). If the first check bit causes the negative transition output to pulse, the gain of both sides of the amplifier will be set to a low gain of 4.8. If only the positive transition output pulses as a result of the second check bit, an intermediate gain of 7 will be set on both sides of the amplifier. If neither output pulses, a maximum gain of 10.7 is selected. Hence, the entire adaptive gain selection process is performed with only one gain change of the amplifier.

6.3 *Adaptive Data Strobe Circuit*

The remaining 19 check bits in the control sector characterize the unique write-to-read delays of each head-track combination. They further compensate for the changes in the clock and data timing relationships with temperature excursions. The clock heads and data heads are mounted on a large, aluminum casting. Because the two types of heads are located some distance apart, the temperature distortions in the casting move the data with respect to the reference bit clock. If the data is written with respect to the bit clock at the lowest ambient temperature and read back at this temperature, only a fixed delay need be added to the clock to obtain strobing pulses to read out the data. However, if the temperature is raised, distortion of the casting may move the data with respect to the clock; this total movement can exceed a data bit time. Hence, the data written at one temperature extreme cannot be retrieved at the other temperature extreme.

To accommodate the temperature range required in the No. 1 ESS ADF, two sets of check bits are written in each control sector before a disk is put into service. The first set of check bits (eight bits) is written with a fixed delay of 200 nanoseconds at a nominal temperature of 90°F. Due to circuit delays, this fixed 200- nanosecond delay places the first set of check bits roughly 400 nanoseconds away from the clock, as seen at the readout (see Fig. 16). However, this is true only at the check bit writing temperature. After the first check bits are written, they are read and variable delay is added to the clock in an amount necessary to equal the write-to-read delay plus the fixed 200-nanosecond writing delay. This places the delayed clock in alignment with the first set of check bits. With this delay in the clock, the

second set of check bits is written directly behind the first. These delay times are described in Fig. 16. After check bits have been written on all tracks, the disk is ready to be placed into service. Once written, the two sets of check bits are permanent and are rewritten only if destroyed or if the replacement of a circuit pack in the read/write path affects the write-to-read delay.

A method similar to the writing of the second set of check bits is used to write into a data sector. Upon receiving the write command, the strobe circuit reads the first set of check bits. It places a delay in the clock equal to the time difference between the first set of check bits and the reference bit clock. This delayed clock, called the write strobe, is then used to write the data in an interlace specified in the address register. Hence, the effect of adjusting the write strobe clock is to place the data in the same position relative to the first set of check bits, independent of the phase shifts in the reference bit clock with respect to the first set of check bits. Reading previously written data, the strobe circuit reads the second set of check bits, which are in the same relative position to the clock as the data, and chooses an optimum read strobe clock for reading the data. Therefore, the timing used in writing and reading of the data on a particular track depends only on phase information contained in the same track as the data and not on a phase varying clock located on a remote track. The range of the strobe circuit (± 450 nanoseconds) accommodates a maximum temperature shift of $\pm 30°F$ within the disk file enclosure.

Another advantage of the write strobe technique is that one set of check bits suffices for all interlaces in a track. Also, this technique requires only a minimum guard gap between each interlace because, independent of temperature, the data in an interlace is always written in the same position on the disk surface.

VII. DATA TRANSFER

A message store administration program, resident in the central processor, loads a disk instruction and a data block address into dedicated 2-word task hoppers in the buffer control call store (Fig. 17a, 17b). Data being transferred to disk is stored in a 32-word data block (Fig. 17c) beginning at the address specified by the contents of the buffer control call store data block address. No further action by the processor is required to carry out the disk data transfer.

The wired logic buffer control disk sequencer bids for and is granted use of the buffer control call store facility for one cycle. The sequencer
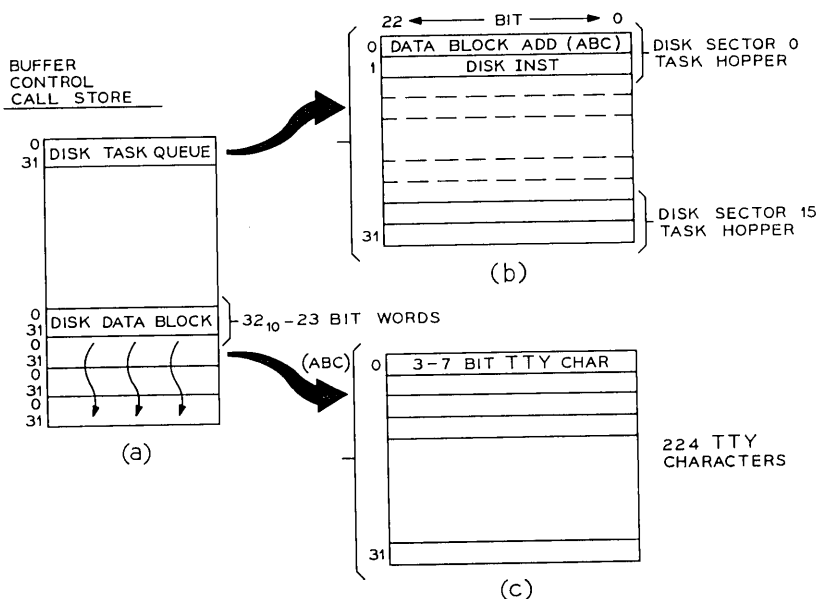
Fig. 17—Message store task queue and data block layout.

reads the 2-word task hopper (address specified by a wired machine constant plus the 4-bit queue counter) and transfers its contents to the disk sequencer address and data registers (Fig. 18). The first hopper word (buffer control call store data block address) is gated to the sequencer address register, the second (disk instruction) is gated to the sequencer data register. Once this preloading is completed, the disk sequencer waits for an instruction request from the message store. After receiving the instruction request and being granted permission to use the data store bus, the sequencer gates the contents of its data register (disk instruction) onto the data store write bus and into the message store address register. This instruction is decoded by the message store to select a unique data head and clock for transferring data.

The disk sequencer then preloads the first data word into its data register (disk write) and waits for the asynchronous data request from the message store. The address of the data stored in the buffer control call store is determined from the disk sequencer address register. When the data request is received by the disk sequencer, the contents of the sequencer data register (disk data) are gated onto the
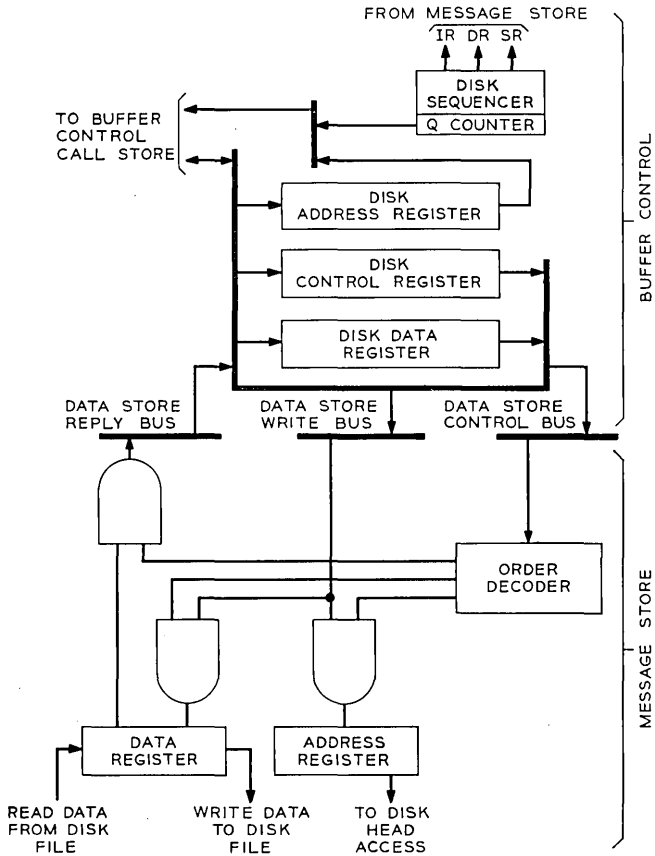
Fig. 18—Wired logic sequencer register access.

bus and into the message store data register. From there, the 24-bit word is shifted, one bit at a time, onto the disk memory surface. This sequence is repeated 32 times to complete the transfer of the 32-word buffer control call store data block into a disk sector. Reading disk data is accomplished in a similar fashion.

Upon completion of the block transfer, the disk sequencer waits for the asynchronous status request from the message store. Once the status request is detected, the sequencer reads the contents of the message store status register (Fig. 19) into its data register. If the status report indicates success, a no-op (all ones code) is written in the buffer control call store task hopper to indicate successful comple-

tion of the transfer. If a failure is detected by the message store or disk sequencer, the entire operation is left in the hopper for one additional retry during the next disk revolution. Failure on the retry causes the buffer control call store task queue instruction to be over-written with a failure status report. The failure status report serves a dual function. First, it alerts the client program requesting the disk data transfer that the transfer was unsuccessful. It is also used as an input to a maintenance fault recognition program which must isolate the faulty unit and prepare it for programmed diagnostic testing.

7.1 *Synchronization*

Sector synchronization must be maintained between the message store and the wired logic disk sequencer. Thus, when the disk is pre-pared to process sector n (determined by the disk position), the sequencer is synchronized to read the current instruction for sector n from the buffer control call store task queue. The sequencer contains a 4-bit binary queue counter (0 through 15 disk sectors) synchronized to the disk sector. Each status report read from the message store specifies the 4-bit sector address of the disk sector just transferred. To synchronize the queue counter to the current message store sector, the sequencer is programmed into a startup mode. When the first status report is read by the sequencer, the 4-bit message store sector address is extracted and written into the queue counter. The queue counter is then used to read the proper task hoppers associated with the nth sector. The counter is incremented by the status request from the message store from 0 to 15 and recycles to 0 at the start of each new job cycle. Once synchronization is achieved, no further initializa-tion is required.
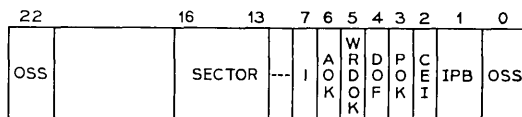
OSS = OVERALL STATUS SUMMARY
IPB = I AM PROBABLY BAD
CEI = COMMUNICATION ERROR INTERNAL
POK = DATA PARITY FAILURE
DOF = DATA OVERFLOW
WRDOK = WORD OK
AOK = ACCESS OK

Fig. 19—Status report exchanged between message store and wired sequencer.

VIII. READ/WRITE FAULT DETECTION

A large portion of the message store circuitry detects faults in the active read/write circuits. In the read mode, with normal system configuration, matching data with the duplicate disk controller provides a powerful fault detection tool. In addition, every 24-bit word has an odd parity bit which is checked as the data is read serially from the disk. When an entire word plus parity is in the 24-bit data shift register, the register is tested for at least a one. This check, plus parity, will detect most faults in the readout if the disk communities are not running in a duplicate matching mode. It also will help identify the faulty unit when a mismatch occurs in the match mode.

The write mode is more thoroughly monitored because, once the data leaves the wired logic sequencer in the buffer control, it is no longer matched. Each of the message store frames writes its data independently. The buffer control sends odd parity with each word to be written on the disk; as data is shifted onto the disk, parity is checked. After the 24-bit word is shifted out of the register, it is tested for all zeros. In addition, for the 2-bit times preceding each word, the write current is turned on and tested for 130 milliamperes ±10 percent. Also, the head selection matrix is tested for exactly one center tap selection and one group selection. Having passed all the above tests, the probability of writing bad data on the disk is minimized.

IX. MAINTENANCE

The message store includes maintenance hardware to check for the proper responses to read/write commands, and to verify proper data handling. This hardware, along with maintenance programs located within the central processor, alert the system to all irregularities encountered during disk data storage or retrieval. The maintenance functions are as follows:

(i) Hardware checks are performed by the message store, and failures cause system interrupts (immediate action).

(ii) Hardware checks are performed by the message store, and failures cause bad status reports to be written in the task queue (deferred action).

(iii) Routine programmed diagnostic tests are run to exercise all maintenance associated hardware.

All data transfers between the buffer control call store and the mes-

sage store are protected by an odd parity bit and an acknowledgment (all-seems-well) signal. Bad parity or the lack of an all-seems-well response will cause an immediate maintenance interrupt in the central processor (Fig. 20). Maintenance interrupts also occur when data read from disk by the duplicated sequencer fails to match.

A maintenance interrupt detected in the central processor stops normal program flow and transfers to an interrupt filter program. Before proceeding to isolate the faulty unit, all processor registers are saved so that the normal program flow can be reentered once the faulty unit is removed. The filter program interrogates unique hardware error indicators to determine the failing subsystem. Control is then transferred to the fault recognition test program for the failing subsystem. The failing unit is removed from service and its simplex mate configured to respond for both itself and the faulty unit. Once the faulty unit has been removed, and an automatic diagnostic test requested, normal processing is resumed at the interrupted program. Diagnostic requests are honored routinely and are run in a segmented mode with normal processing. Diagnostic test programs resolve the fault to within a few replaceable plug-in circuit boards and print out a trouble number associated with this location. After the faulty circuit pack is replaced, a diagnostic test is run to verify the subsystem. The message store is then updated from the active unit and returned to normal service.

The second maintenance function causes no immediate interrupts. A status report is read by the wired logic sequencer at the end of each disk sector. The message store replies with a data word indicating the results of hardware checks performed during the previous block transfer. If the sequencer detects a status report failure, and the data transfer is a second attempt, it overwrites the corresponding instruction word in the buffer control call store task hopper with the failure report. Thus, a failing data transfer is automatically abandoned by the hardware, and notification given to the client administration program by the failure status report in the task queue. The presence of a failure status report in the task queue causes the administration program to enter the fault recognition program at the block repeat entry (see Fig. 20). Tests performed at the fault recognition level are sufficient to isolate and remove the faulty message store half and request diagnostic testing.

Maintenance circuitry is tested by routine daily diagnostics performed on the message store. The diagnostic tests exercise all message store hardware and read/write circuits.
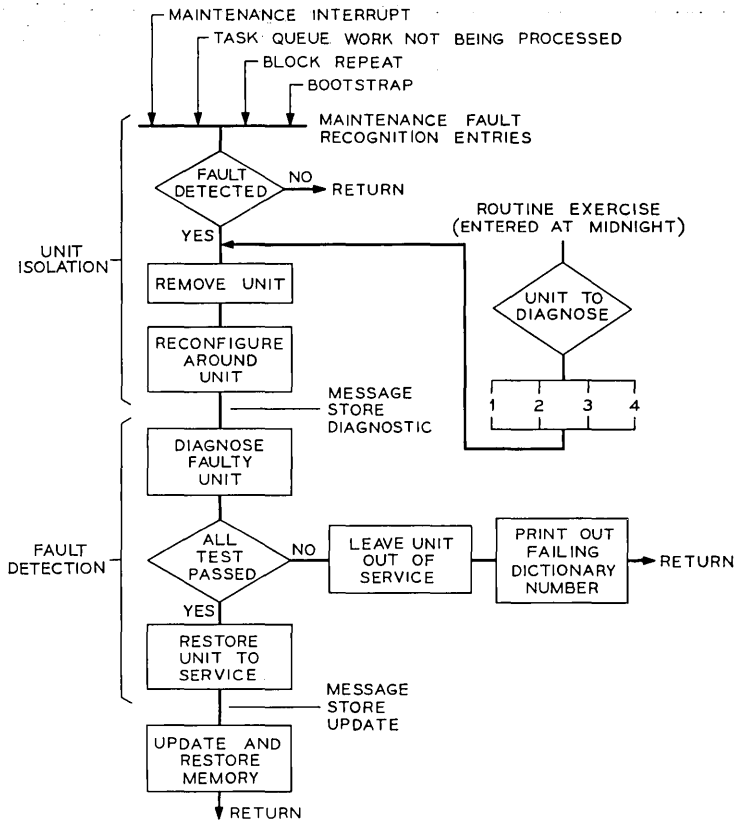
Fig. 20—Maintenance software flow diagram.

X. CONCLUSION

The message store is a 60-megabit disk file storage system. The cost per bit of this storage is two orders of magnitude less than the cost for the ferrite sheet call store employed in the No. 1 ESS. The disk memory system is run in full duplex to enable matching of all data by a duplicated wired logic sequencer. The integrity of the data is insured by internal hardware checks, all-seems-well, data parity, and cross matching of all bus transmissions.

Disk operation is asynchronous with the fixed 5.5-microsecond processing cycle. The disk cycle is determined by a set of programmed clocks permanently recorded on the surface. Two pairs of disks are held in frequency and phase synchronism by a closed loop servo.

The hardware is backed up by maintenance software programs to detect, isolate, and diagnose failing circuits. Operating experience, since system cutover on February 3, 1969, indicates that the primary message store meets expected reliability requirements. Faulty circuits have been detected and isolated by the field craftsman using a fault dictionary. Faulty data heads within the disk file have also been isolated by a special write/read failure matrix test provided as part of the diagnostic program.

XI. ACKNOWLEDGMENTS

REFERENCES

1. Genke, R. M., Harding, P. A., and Staehler, R. E., "No. 1 ESS Call Store," B.S.T.J., 43, No. 5 (September 1964), pp. 2147–2191.
2. Cagle, W. B., Menne, R. S., Skinner, R. S., Staehler, R. E., and Underwood, M. D., "No. 1 ESS Logic Circuits and Their Application to Design of the Central Control," B.S.T.J., 43, No. 5 (September 1964), pp. 2055–2095.
3. Aitcheson, E. J., and Cook, R. F., "No. 1 ESS ADF: Maintenance Plan," B.S.T.J., this issue, pp. 2831–2856.

No. 1 ESS ADF:

Magnetic Tape Subsystem

By J. L. POTTER, MRS. F. B. STREBENDT and J. R. WILLIAMS

(Manuscript received May 18, 1970)

The No. 1 ESS ADF magnetic tape subsystem provides a high-volume, low-cost storage medium in which user messages are filed, and from which current message copies can be automatically retrieved upon user request. Message handling and statistical information concerning all message switching transactions are also filed on tape, providing a journal record for the switching center. Emphasis on maintenance and defensive programming techniques enables the tape subsystem to provide continuous and excellent service. This article describes the tape equipment and program package.

I. INTRODUCTION

Message switching systems designed for nationwide use must provide a high degree of transmission reliability and protection against loss of messages. Extensive efforts have been made in the design of the No. 1 ESS ADF to insure these system objectives. Once the system has accepted a message from its originator, the ADF system has the sole responsibility for a correct and guaranteed delivery. Nevertheless, messages received at a user's terminal may be lost by an attendant, may be garbled because of transmission difficulties such as fades, distortion, and line hits, or may be lost as a result of a paper jam in the receiving terminal. To counter such occurrences, the ADF system has a message retrieval feature, whereby a missing or questionable message may be retrieved upon request. Magnetic tape is the vehicle used for long term message storage and retrieval in the ADF system.

Considerations which led to the selection of magnetic tape as the message recording medium included storage volume, access speed, and frequency of access. A copy of every message handled by the ADF system is saved for days in this message file, called the permanent file. This represents an exceptionally large storage requirement be-

cause of the high message handling capacity of the ADF system.[1] The use of magnetic tape was indicated by this need. Speed of access was found to be reasonably noncritical; however, the expected volume of retrieval requests was estimated at 1 percent of messages terminated. These considerations economically precluded the use of manual retrieval techniques and led to the use of automatic on-line magnetic tape storage and retrieval.

A key to the message retrieval process is a cross-reference file maintained in the message store.[2] The storage locations of all retrievable messages for each user who has the message retrieval feature are maintained in this file.

A second ADF system information filing requirement which is fulfilled by the magnetic tape subsystem is the recording of journal records for all message switching transactions involving the ADF system. These records, placed on magnetic tapes called journal file tape reels, provide a means for evaluating overall system performance. Further, the journal files enable telephone company retrieval center attendants to assist the ADF user with special retrieval requests. To gain access to the journal file, a journal file retrieval process has been included in the No. 1 ESS ADF magnetic tape subsystem.

The magnetic tape subsystem, consisting of tape equipment and a complete tape program package, is integrally provided for each ADF installation. Two tape unit controls with dedicated buffer control[1] tape hardware sequencers provide electrical access to and control of up to 16 tape units.[1] Maintenance software and hardware is provided to promptly detect hardware anomalies or failures.[3,4] This software provides for (*i*) quick recognition of and removal of faulty units without adversely affecting operational performance, (*ii*) diagnosis and isolation of faults to basic elements, and (*iii*) routine exercise of all tape equipment to insure functional integrity.

II. TAPE EQUIPMENT

Message and journal data storage and retrieval operations are performed under software control on tape equipment illustrated in Fig. 1. The tape control hardware is divided into two independent communities, each having a tape instruction queue in buffer control call store, a tape sequencer in each buffer control, and a tape unit control. Each community can gain access to each of the tape units in the office, provided that both communities are not simultaneously competing for the same tape unit.
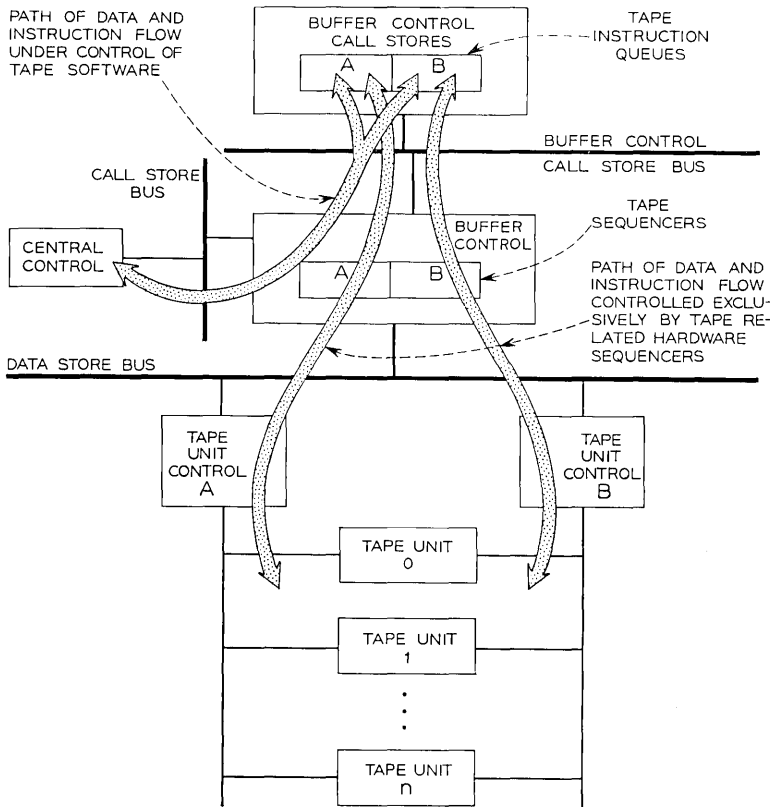
Fig. 1—Block diagram illustrating hardware units, buses, and paths of data flow involved in tape operations.

Duplication of tape related hardware units is limited to the buffer control tape sequencers and the tape queues in buffer control call store. Each buffer control has a sequencer for each tape community and thus an equipment outage will not impair tape system operation. Tape queue duplication is a consequence of buffer control call store duplication. Each tape unit control operates as a simplex unit, communicating with the buffer control tape sequencer appropriate to the configuration.

Tape instructions are loaded in the tape queues by software orders and are independently executed by the hardware. At the conclusion of the operation, completion (status) reports are placed in the tape queues for software appraisal and action.

Functionally, tape filing operations are processed on the tape unit control designated by software as primary, and retrieval operations are executed on the tape unit control designated secondary. Tape filing operations are nondeferrable, since prompt release of message store and call store facilities following final message termination is necessary to achieve the high capacity of the No. 1 ESS ADF.[5] Thus whenever a tape unit control must be removed from service because of faults, the other tape unit control is assigned the primary function. Retrievals are deferred until both tape unit controls are in service.

Each of the tape units is automatically assigned a function appropriate to the status of the tape mounted on it. Two tape units are used for permanent message filing, the active permanent file, and the standby permanent file. Filing is automatically switched between these tape units to allow prompt retrieval of messages on the active unit. In essence, filing is accomplished on the active unit, while the standby unit is available for retrieval work. When the standby is inactive, its tape is positioned just after the last record written, ready to return to the active state if a retrieval is requested from the active permanent file.

When an active permanent file tape unit reaches the end of a tape, a replacement is assigned from a pool of ready tape units, that is, tape units with clean reels of tape positioned at the beginning of the tape, loaded and in service. Then the replaced active permanent file is rewound and placed in the pool of on-line tape units. These tape units provide the large bulk storage of messages which is accessed by tape software in response to message retrieval requests.

The ready tape unit pool is maintained at a minimum of two tape units. If this pool becomes deficient, the on-line permanent file tape unit with the oldest reel is automatically removed from service, and an order is sent to retrieval center personnel to install a clean tape. When the new tape is loaded, the tape unit is placed in the ready pool. The old reel then is placed in off-line storage.

Journal records are written on a single tape unit assigned for that task. When a journal file reel has been completely written, it is removed and placed in off-line storage. This allows no retrievals from the active journal file, but such retrievals are seldom required before the complete tape reel is written. If required, provision is made for terminating the filing assignment on the current active jounal file reel.

Retrievals from journal file and permanent file reels that are not mounted on tape units are accomplished by the assignment of an off-

line retrieval tape unit. Retrieval center personnel are directed by the program via teletypewriter to make required reel changes during the execution of these retrievals. When off-line work is completed, the off-line retrieval tape unit assignment is terminated, and the tape unit rejoins the ready pool.

Tape hardware configuration and control is automatically accomplished by tape software. Retrieval center personnel are required to change reels only as directed by the system and to handle special cases of nonretrievable messages, reported by the system.

To provide the features discussed, the tape hardware is capable of executing the following operational orders:

(i) Read—forward direction,

(ii) Search—permanent file forward and reverse (based on matching the first word in record), journal file forward (based on matching on three-word groups within journal file records),

(iii) Write—forward direction of variable length records,

(iv) Backspace—up to 15 records per command,

(v) Advance—up to 15 records per command, and

(vi) Rewind—at 225 inches per second.

2.1 *Tape Unit*

Each tape unit in the No. 1 ESS ADF magnetic tape subsystem includes a special design Ampex Corporation transport with associated transport servo electronics. Read amplifiers and logic for transport control and communications with the tape unit control were designed by Bell Laboratories. Nine-channel, 800-bits per inch, nonreturn to zero operation at 56.8 inches per second is provided for normal forward and reverse operation. Rewinding at 225 inches per second enables a 2400-foot reel to be rewound in two minutes. Figure 2 shows a typical tape unit installation.

The tape unit responds to several commands from a tape unit control in order to execute the various instructions: write, read, forward, reverse, move, stop, rewind, select and unselect. Data is written on tape in records of 1 to 15 blocks with 3/4 inch gaps between records. Each block consists of 31 call store words of data plus one link word. A word is 24 bits long, with 23 information bits and one parity bit. On tape, each word is written in three segments called bytes. A byte is nine bits, eight information bits and one parity bit.

To insure integrity of data written on tape, each tape unit is
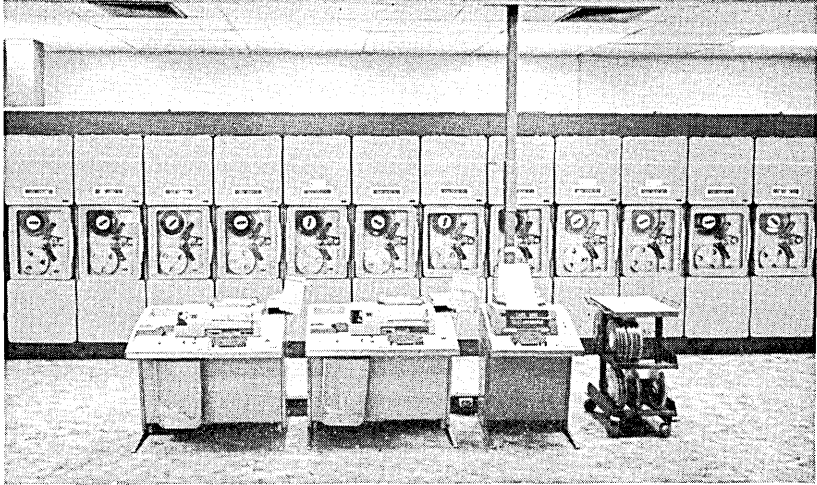
Fig. 2—A typical tape unit installation.

equipped for read after write, thus enabling the tape unit control to verify each write operation.

2.2 *Tape Unit Control*

The tape unit control is a consolidation of common tape control equipment that interfaces the buffer control and tape unit. Extensive circuitry is provided for error detection in tape operations. Figure 3 pictures the tape unit control.

The tape unit control receives tape instructions from the buffer control, stores each instruction in its instruction register, decodes the instructions, and controls tape unit operation during execution of the instruction. Data transfers between tape and call store pass through the tape unit control's 24-bit data register, where word-to-byte transformations are accomplished.

Read, write, and search control circuits coordinate data transfers between tape units and the buffer control, and control operation of the selected tape unit.

2.2.1 *Operation*

Upon receipt of an instruction, the tape unit control decodes the order and selects the proper tape unit. For a write operation, the tape unit control orders the selected tape unit into the write, forward, and
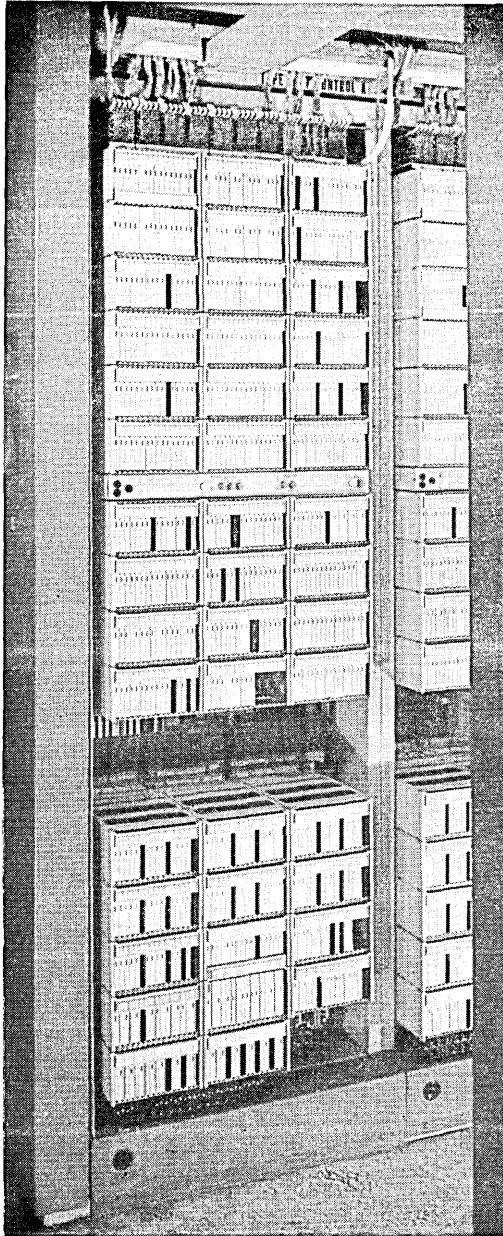
Fig. 3—Tape unit control.

move states. Data is transmitted to the tape unit, a byte at a time, at a rate necessary to achieve 800-bit-per-inch density at 56.8 inches per second. When the order is completed, the tape unit control stops and releases the tape unit, and reports status to the buffer control. Read instructions are similarly handled by the tape unit control.

At the beginning of the search operation, search data received from the buffer control is saved in the tape unit control. Next, the tape unit is selected and controlled as in a read operation. Data read from tape is not sent to the buffer control; instead, the search control circuitry matches search data within each record obtained from tape against the search data received from the buffer control. When an exact match is obtained, the search operation is terminated, the tape unit is stopped and released, and successful search status is reported to the buffer control.

Two types of search instructions are provided. The journal file search instruction searches the first five blocks of every record to match 9-byte search entries. Section 4.4 of this article describes the layout of the journal file record. The permanent file search matches a 24-bit search number against the first word (three bytes) of every record passed on the tape. Permanent files can be searched in either forward or reverse.

2.2.2 *Integrity*

The tape unit control contains several mechanisms for guaranteeing the integrity of data transmission and storage. One method used is the all-seems-well response[6] to buffer control transmissions. Absence of an all-seems-well response causes a system interrupt, which results in immediate program action towards solving the difficulty.

Communications between the tape unit and its control are monitored via status registers in each tape unit control. Failures or data errors cause unique status bit combinations in the status register. At the conclusion of every operation the contents of the status register are transferred to the tape queue in buffer control call store. Program detection of a bad status triggers the tape system fault recognition software.

Transverse and longitudinal parity checks insure the integrity of data stored on tape. As a part of the write operation, the tape unit control generates parity over each 8-bit byte and stores it as the ninth bit of the byte. During a read operation, after parity has been checked, the tape unit control strips off this parity bit. A parity

failure causes the tape character error counter in the status register to be incremented and the tape character error indicator to be set.

As data is being written, the tape unit control also computes longitudinal parity over the entire record. At the end of the record, this byte is written on tape. This serves as an additional error detection device used in read operations.

The tape unit sends data to the tape unit control in an amplified analog form over a common bus. For reliability, the tape unit control has two signal detectors, the high and the low amplitude threshold read registers, listening to this bus. If both registers agree, the information is accepted as valid. However, if the low register's parity passes and the high register's parity does not, the data is judged in error and the tape character error indicator in the tape unit control is set. Nevertheless, the entire record is read. This enables the No. 1 ESS ADF to retrieve the best copy of the requested data, if after multiple attempts parity errors continue to occur.

The status report, initiated by the tape unit control upon completion of an operation, enables the software to determine the success or failure of the order. Decisions to try again, remove equipment from service, or continue normal operational sequence are heavily based on the status report. The layout of the tape status report (Table I) provides insight into the error detecting and reporting scheme of the tape unit control.

2.3 *Buffer Control Tape Sequencers*

Within each buffer control, dedicated tape sequencers are provided for each of the two tape unit controls. These sequencers control the flow of information between the corresponding tape unit control and its dedicated tape instruction queue in buffer control call store.

Each tape instruction queue consists of 32 buffer control call store words. Tape programs place the instructions, in two-word instruction sets, in the desired queue. The first word of the instruction contains the address of the buffer control call store link list of data blocks to be used for the data transfer. The second word contains the operation code (read, write, backspace, and so on), the tape unit involved, and other pertinent data.

The sequencer is responsible for obtaining the data from (or storing data in) buffer control call store and sending it to (or receiving it from) the tape unit control upon request. Data in buffer control call store is stored in 32-word block link lists. The sequencer inter-

TABLE I—LAYOUT OF TAPE STATUS WORD

Bit	Description
0 1 2 3	Beginning of tape End of tape Rewinding Unsuccessful search
4 5 6 7	Tape unit failure in setup Tape unit failure in run No character response Data time-out
8 9 10 11	Parity error Word count error Parity generator error Longitudinal parity mismatch
12 13 14 15	Write longitudinal parity mismatch Longitudinal parity error Tape character error Counter overflow
16 17 18 19	Word error counter
20 21 22	Buffer control invalid sequence Successful operation Successful operation

prets the last word of each block as the link address of the next block of data. Each transfer of data from the buffer control to the tape unit control is verified by all-seems-well techniques.

The tape sequencer responds to instruction, data, and status requests from the tape unit control. In the idle state, a tape unit control continuously sends instruction requests until the sequencer finds work to do. When an instruction is found in the tape instruction queue, the sequencer sends the instruction to the tape unit control. If a transfer of data to tape is involved, the tape unit control sends a data request to the tape sequencer, in response to which the sequencer obtains a data word from buffer control call store and transfers it to the tape unit control. When this data has been transferred to tape, the tape unit control sends another data request to the sequencer. The next word is sent to the tape unit control, and the process continues until all data has been transferred to tape.

When the tape unit control has completed execution of the instruction, it sends a status request to the buffer control tape sequencer.

In reply to this, the tape sequencer fetches the execution status from the tape unit control and stores it in the queue for later software examination. Unsuccessful status freezes the sequencer, thus retaining tape unit control and sequencer information for fault recognition processing. After a successful operation, the sequencer proceeds to the next instruction in the tape queue.

Checks are made within the tape sequencer as well as the tape unit control to insure that the correct number of data words have been transmitted. If errors are detected, appropriate status bits are set and the sequencer is stopped as for other errors.

III. MAINTENANCE CONCEPT

The objective of the tape subsystem maintenance scheme is to provide a high degree of dependability and maintainability, consistent with ESS standards. To meet this objective, sophisticated techniques are used in fault detection and recognition, diagnosis, error analysis, and routine exercise.

3.1 *Environment*

Although duplicated hardware matching is a standard technique used in high reliability systems, it is not used in all segments of the tape subsystem. Notably, the tape units and tape unit controls used in the No. 1 ESS ADF magnetic tape subsystem are operated in a simplex mode. Reliable simplex operation of these units is provided by extensive maintenance hardware and software for detecting, reporting, and analyzing errors. The simple fact that two tape unit controls and several tape units are installed in the No. 1 ESS ADF switching center does mean that functional redundancy effectively exists. All critical filing operations are continued, for example, when a tape unit control is out of service. For such an occurrence, the retrieval function is simply deferred until the second tape unit control is restored. The powerful ESS diagnostic approach used for tape subsystem equipment insures that equipment downtime is minimized.

Tape maintenance procedures must deal with several types of faults and abnormalities. The hard logic failure is easily detected and isolated, but intermittent logic and mechanical problems are a serious challenge. Numerous retries of failing instructions and error analysis methods are used to overcome the latter problems.

3.2 *Fault Recognition*

The tape subsystem fault recognition software:

(*i*) Determines action required when tape subsystem errors are detected.

(*ii*) Identifies true faults from superficial failures such as minor tape imperfections.

(*iii*) Isolates true faults to the defective equipment unit.

(*iv*) Removes faulty equipment from service for diagnosis and repair.

Fault recognition processing begins upon detection of tape hardware interrupts or failing status reports.[3,4] Faults which cause interrupts are isolated by the program to the faulty tape unit control, buffer control, or data store bus. The faulty unit is then removed from service and diagnosed.

When fault recognition is stimulated by an unsuccessful status report in the tape instruction queue, the failing instruction is repeated up to four times. If any of these four attempts passes, no further action is taken. After four consecutive failures the instruction is tried on the other tape unit control. A success implies that the original tape unit control was at fault. If no attempt succeeds, the tape unit is assumed to be at fault since it cannot work with either tape unit control.

This simple algorithm has some complications resulting from the uniqueness of the tape system. First, it is difficult to tell if the tape unit control, the tape unit, or the magnetic tape is at fault. For example, a longitudinal parity failure during a tape operation may be caused by defective tape or a bad parity circuit in the tape unit control. A simplification is made by assuming that tape imperfections are minute and isolated to a small portion of tape. For a failing write instruction, before trying the instruction on the other tape unit control, up to six inches of tape is skipped and the write process repeated. If this fails, the failure is assumed to be an electronic fault. Success implies a tape imperfection, and no further action is taken.

When a tape unit is removed from service, the tape is placed in off-line storage. However, when a tape unit control is removed from service, the tape unit it was addressing is not removed. Incorrect or incomplete records are erased from the tape using the in-service tape unit control. Correct data is then written on the tape using the in-service tape unit control.

3.3 *Error Analysis*

Error analysis provides the means by which tape system fault recognition copes with marginal electronic circuitry and mechanical ab-

normalities. Both problems are usually intermittent and not consistently reproducible. Therefore records are kept in call store of all failures. When any one type of failure is excessive the unit is removed from service and diagnosed.

3.4 Diagnostic

The tape subsystem diagnostic software provides a thorough test of all tape-related hardware equipment. The objective, of course, is to isolate failures to individual replaceable circuit elements such as circuit packs. Diagnostic procedures are provided for the tape unit controls, tape units, tape sequencers, and connecting buses.

The diagnostic software reports results via the maintenance teletypewriter to maintenance personnel. If an error is detected, a unique 12-digit trouble number is reported which, when checked in a trouble locating manual, identifies the faulty circuit.

Mechanical faults are difficult to diagnose. Inconsistent diagnostic results often occur, yet optional raw data diagnostic printouts assist maintenance personnel in locating faulty components.

3.5 Exercise

The tape subsystem exercise routinely checks idle hardware to insure that operationally undetectable faults do not remain undetected. This is accomplished by periodically running all equipment diagnostics on those units which have not been diagnosed within a given time. This technique guards against faults in the maintenance failure detecting circuitry which could mask other faults, and thus prevent error detection.

Each tape unit control is routinely exercised, provided that one tape unit control remains in service at all times. All ready tape units are exercised. However, exercises of tape units assigned to an active filing, retrieving, or on-line storage pool are deferred until termination of the functional assignment.

If an exercise passes, the equipment unit is returned to service. If a fault is detected, a diagnostic report is sent to the maintenance personnel, and the equipment unit is left out of service.

IV. OPERATIONAL FEATURES

The primary purpose of the magnetic tape subsystem is message retrieval. The permanent filing feature exists mainly to support message retrieval. As Section II discusses, on-line permanent file tape

units provide for the bulk storage of customer messages. Assuming a complement of 16 tape units, 10 would normally be in the on-line pool. Using 2400-foot reels and average message statistics, the on-line retrieval capacity is approximately 125,000 messages. Older messages are retrievable semi-automatically from off-line reels.

The configuration, control, and use of the tape subsystem equipment is totally automatic. The system maintains in the cross-reference file the storage locations of all retrievable messages handled by the system. Further, the location of every tape reel is maintained in a call store table. Both of these are routinely audited to insure integrity. Retrieval center personnel need change tape reels only as indicated by the system. For permanent and journal filing as well as on-line retrieval operations, no manual intervention or tape handling is required.

Retrieval center personnel have access to the journal records via the journal file retrieval process. Information filed on the journal tape gives a history of message switching transactions but does not include any message text. Through off-line analysis of the journal tapes, telephone company personnel can ascertain the quality of total user service, operating and traffic statistics, and switching transactions. The journal file retrieval capability also enables retrieval center personnel to assist the user with special problems for any given message.

4.1 *Permanent Filing*

Each message handled by the No. 1 ESS ADF is recorded on permanent file tape after all terminations for the message have been completed.

For program and tape packing economies, messages are divided into three categories, based on total file length (that is, the total number of message heading, text, and message processing blocks.) Messages of five blocks or less (short messages) are packed two or three to a record, provided that sufficient messages are terminated with this characteristic within a fixed time. Since the maximum tape record is 15 blocks, messages greater than 15 blocks (long messages) are filed in several records. Messages from 6 to 15 blocks long (normal messages) are not packed and thus record size in blocks is equivalent to message length.

In the first word of each tape record is placed a unique 23-bit search number. As Fig. 4 shows, this number consists of a base search number, and long and short message indexes. The short message index is set to zero in the tape record. After a message has been filed on

BITS 16-22	BITS 2-15	BITS 0-1

LONG MESSAGE   BASE SEARCH   SHORT MESSAGE
INDEX   NUMBER   INDEX
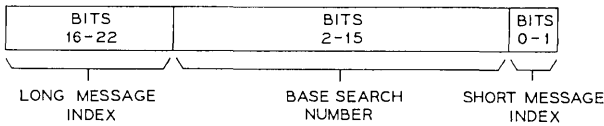
Fig. 4—Breakdown of search number used to locate a message, or part of a message, on tape.

tape, the base search number (and short message index if applicable) and tape reel number are placed in the cross-reference file forming the basis for searching if a retrieval is requested. When permanent filing is started on a new tape, the base search number is set to one. For subsequent records that are not long message residue segments, the base search number is increased by one, providing the uniqueness.

For a normal length message, the long and short message indexes are always zero. For a long message, the long message index is zero for the first record and is increased by one for each subsequent record of the same message. The base search number remains the same for all records of a given long message. For short messages, one base search number applies to all messages within a record. The short message index which is placed in the cross-reference file entry identifies where the message is packed within the record. This index is zero in the search slot at the beginning of the tape record.

Permanent file strategy does not require that all records of the long message be written consecutively. High efficiency is obtained by involving the program is concurrent processing of numerous messages. Figure 5 shows an example of final tape data packing.

A tape-to-tape link is provided when the end of tape is reached when partially recorded long messages exist. The balance of these messages are written on the replacement permanent file tape, and a link record is written on the first tape, linking each such message to its new tape address. Thus a tape can be completely used without sacrificing retrieval capabilities. At the same time, call store and primary message store space can be released when each long message segment is written on tape.

The permanent file message tape copy contains a complete copy of the message, its heading, and associated message processing blocks.[7] Thus when a message is retrieved from tape, the retrieval program readily verifies correctness and completeness, reconstructs the message, and nominates it for delivery.
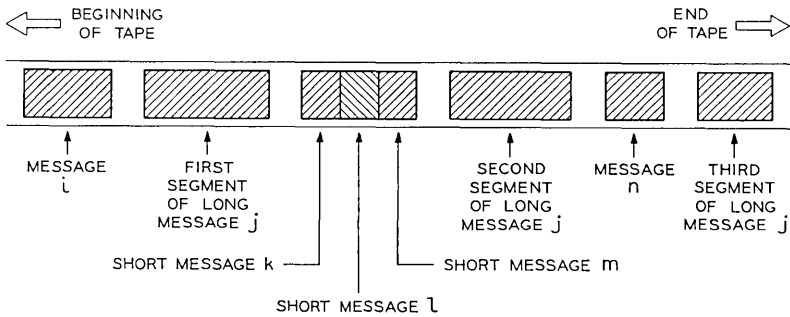
Fig. 5—An example of data packing on permanent file tape.

4.2 Cross-Reference File

A vital link in the message retrieval process is the cross-reference file, which is a station-by-station record in the message store of messages sent or received. The cross-reference file is the basis for message retrieval since it contains the location or retrievability status of messages handled by the No. 1 ESS ADF. The fixed cross-reference file area for a particular station can be found in that station's translation information. From this, the entry concerning a given message can be obtained via an indexing scheme using the message number.

The cross-reference file organization consists of four different kinds of blocks, each 32 words long. (See Fig. 6.) Recycling blocks point to summary blocks, and summary blocks point to file blocks. Accumulating blocks accumulate cross-reference data, and when full, their contents are transferred into file blocks.

Within each station's fixed area of cross-reference file are two recycling blocks, one for terminated messages and one for originated messages. A recycling block is an overall record of received or sent messages. The first 21 words of the recycling block are divided into seven sections of three words each. Each section represents a group of 1000 message numbers called a cycle. Message numbers for a station generally proceed from 1 to 999 and then go to 1 again, although stations may reset their message number counter before normally recycling. The newest cycle is at the top of the recycling block and the oldest at the bottom. Since the seventh cycle is needed for updating, each station's cross-reference file can contain information for a maximum of 6000 messages each for send and receive message numbers.

For each cycle there is a summary block address, the date and time

of the first message number in that cycle, and the message number upon which the cycle ended. The last two words of the recycling block contain the oldest valid message number cycle and the oldest message number therein for which cross-reference file information has not yet been released.

Summary blocks and file blocks are seized from a pool common to all stations. The number of these blocks that a station may seize is a subscriber feature directly affecting retrieval capability. Thus, a station may elect to be able to retrieve from none to the latest 6000 messages. A summary block contains the addresses of all the file blocks pertaining to a given cycle within a recycling block. Each of the 32
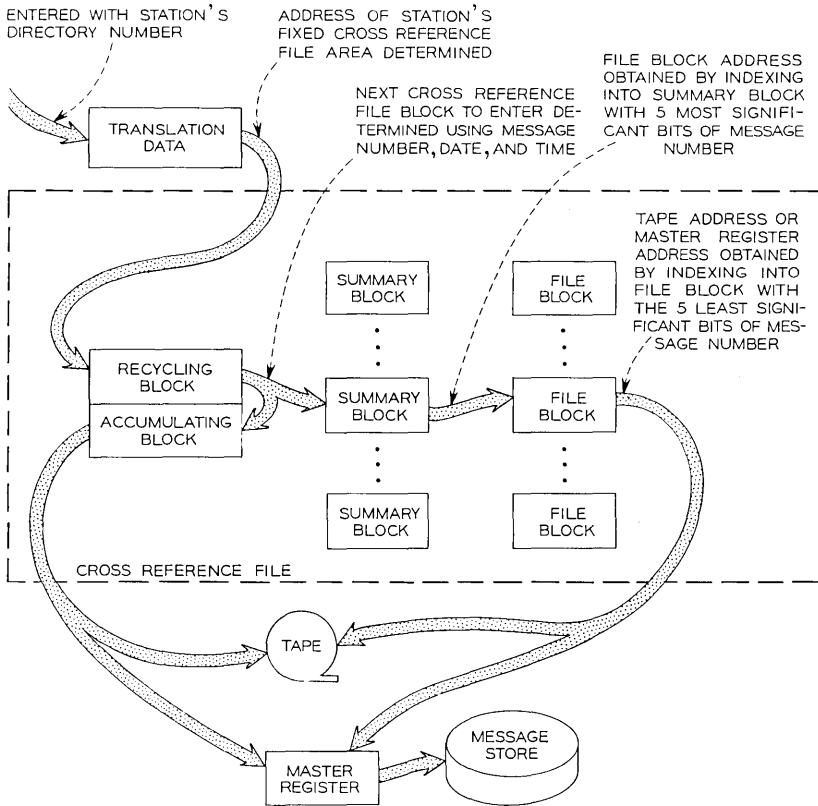
Fig. 6—Layout of cross reference file, illustrating indexing scheme for locating a message on tape or in primary message store.

words of a summary block points to a file block if this is a full cycle of 1000 messages. A file block contains the location or status of a group of 32 messages such as those for message numbers 1 to 32, 33 to 64, and so on.

An accumulating block collects entries for the most current group of 32 messages. When a message number which is a multiple of 32 is reached, the contents of the entire accumulating block are transferred to a file block. The appropriate slot within the summary block is also updated with this file block address. Accumulating blocks, like re-cycling blocks, are part of each station's fixed cross-reference file area. There is one accumulating block for receive message numbers and one accumulating block for send message numbers.

As soon as a message enters the No. 1 ESS ADF, an initial entry in the cross-reference file is made for the originating station's send message number. As each message termination is completed, an initial cross-reference file entry is made for that addressee's receive message number. In these initial entries, a master register[7] address is inserted into the appropriate message number slots in the corresponding station's accumulating blocks. At this time the master register points to the address of the message in the message store. After all deliveries are made and the message is permanently filed on magnetic tape, the appropriate cross-reference file slots for the originator and all termi-nators are updated with the tape address of the message. If the mes-sage was not completely filed, if it was an action request, or if it was aborted in origination or delivery, the cross-reference file entries are marked with a non-retrievable code.

4.3 *Message Retrieval*

The message retrieval service feature permits user stations to retrieve their messages in the event of garbled copy, message numbers out of sequence, a station out of paper, or the like. The location of each mes-sage requested is obtained via the cross-reference file. Each message is found, processed, and sent to the retrieval destination. Examples of a garbled message, a message retrieval request for that message, and the message retrieval output are shown in Fig. 7.

4.3.1 *Requesting Message Retrieval*

The input stimulus for message retrieval is an input message called a resend action request, whereby a user station requests the No. 1 ESS ADF to resend one or more messages. Basically, the only items

R237 01/30 1211 CST ← _ _ _ _ — HEADING GENERATED BY
                                  NO. 1 ESS ADF

AN EXAMPLE OF -A MESSAGE THAT WAS GARBLED

ATATEDNRSRETRINGMESSIAS ERRGRUISIDE HANDLING ⎱— TEXT

A LINE FEED CHARACTER.

(a)

ACTION
REQUEST   S119 01/30 1522 CST ← _ _ _ _ — HEADING GENERATED BY
MNEMONIC   NO. 1 ESS ADF
           ⟶ ACT-REQ   _ _ _ _ _ — MNEMONIC OF STATION
                                            WHOSE MESSAGE IS
             /RESEND/BTL100   TO BE RETRIEVED

             R237 01/30 1215 CST ← _ _ _ _ — MESSAGE NUMBER PLUS
                                             OPTIONAL DATE AND TIME

             /TO/BTL101
                         ↖ _ _ _ _ _ _ _ _ — MNEMONIC OF STATION
                                            TO RECEIVE RETRIEVAL COPY

(b)

                                         _ _ — HEADING GENERATED BY
                                              NO. 1 ESS ADF
             R354 01/30 1525 CST ←   _ — MNEMONIC OF STATION WHOSE
,DATE/TIME   _ _ _ _   MESSAGE WAS RETRIEVED
'OF RETRIEVAL ...05-03...COPY R237 _ _
REQUEST   _ _ _ _ — MNEMONIC OF STATION
             01/30 1522 CST BTL100 BTL101 BTL101   REQUESTING THIS
                                                    RETRIEVAL
           ⎧ R237 01/30 1211 CST   ↖ _ _ — MNEMONIC OF
           │   RETRIEVAL COPY
           │ AN EXAMPLE OF A MESSAGE THAT WAS GARBLED DESTINATION
RETRIEVED ─┤
 MESSAGE   │ AT THE RECEIVING END AS A RESULT OF A
           │
           │ STATION OR TRANSMISSION ERROR WHILE HANDLING
           │
           ⎩ A LINE FEED CHARACTER.

(c)

Fig. 7—(a) A garbled message received at station BTL100. (b) Message retrieval request originated at station BTL101. (c) Retrieved message copy received at station BTL101.

that must be specified are the identity of the station whose messages are to be retrieved and the message numbers desired. (A maximum of 100 message may be requested in a single action request.) Unless dates and times of the messages requested are also included, it is assumed that these messages are in the most recent cycle of 1000 messages. A variation in format permits retrieval from a given message number through the most recent message number for that station. The telephone company retrieval center has the further capability of requesting a message whose tape address is already known. In all cases, the retrieval destination, unless otherwise specified, is the action request originator.

4.3.2 *Locating Messages Requested*

When a message retrieval action request is received, the cross-reference file of the station whose message is to be retrieved is interrogated for each message number desired. First it is determined whether the message is retrievable; if it is, its location is obtained.

It could happen that a desired message is too old, that is, its pertinent cross-reference file area has been released. In this event, a service message is sent to the retrieval center, where the attendant may, through journal file retrieval, obtain the tape address, which then is used for the retrieval request.

A message number may be nonexistent or nonretrievable. An example is a requested message number larger than the most recently used message number as indicated by the message number counter. This results in a service message to the action request originator.

If a retrieval is requested for a message which is being filed, a delay mechanism causes the cross-reference file to be interrogated later.

4.3.3 *Queuing Tape Search Requests*

In most cases the cross-reference file entry for a message number contains a permanent file tape address, consisting of a tape reel number and a search number. Because of the time-consuming nature of tape operations and the fact that only one search can be performed at a time, this tape address is used to make an entry on a tape search queue. A tape search queue is a call-store-linked list of retrieval requests for a given tape reel. Each tape search queue entry consists of a four-word call store facility called a retrieval register. Each retrieval register contains the message number requested, the tape address of the desired message, and a reference to the action request (which is stored in primary message store until the message has been found).

Several tape search queues are involved in message retrieval. For on-line tape reels there is a queue per tape unit. For off-line tape reels there is only one queue. Entries on any tape search queue are ordered according to ascending search number. In the case of the off-line tape search queue, entries are grouped first-in first-out, according to tape reel number. Within each off-line tape reel number group, entries are further arranged in order of search number. Tape searching normally proceeds in the direction of increasing search numbers.

4.3.4 *Scheduling Tape Searches*

All tape search queue entries are serviced in reasonable time because

of a tape search priority scheduling scheme. New tape search operations are started on the highest priority scheduled tape search queue which has entries.

The four categories of tape search queues, listed in order of priority, are:

(*i*) *Ready-On-Way:* For a tape unit with the oldest completed on-line permanent file tape scheduled for reel change on completion of search.

(*ii*) *Timed Out:* For an on-line tape search queue that has not been serviced for an excessive time.

(*iii*) *Off-Line:* For search requests for either permanent or journal file off-line tapes. Off-line permanent file searches are scheduled to alternate with journal file retrieval searches.

(*iv*) *On Timing:* When an on-line tape search queue is initially set up, it is given this lowest priority. A timer is also set so that the priority can be upgraded if this tape search queue is not processed within a reasonable time. If a search is to be performed from this schedule, the longest queue is selected first.

4.3.5 *Searching A Permanent File Tape*

The prerequisites for starting a search are that (*i*) another search is not in progress, (*ii*) the secondary tape unit control is available, and (*iii*) the tape unit (corresponding to the tape search queue scheduled for processing) is available for searching.

If the tape selected for searching is on the active permanent file tape unit, writing of the permanent file is first switched to the standby permanent file tape unit. The former active permanent file tape unit becomes the standby, and its tape is rewound and made available for searching. When retrieval is completed on the standby, its tape is positioned just after the last record written so that it may immediately resume permanent file writing when needed.

If an off-line permanent file tape is scheduled for searching, an off-line tape unit is assigned. The tape attendant is instructed to replace its tape reel with the desired off-line permanent file tape. When this is accomplished, the tape unit is made available for searching. When retrieval is completed from this off-line tape reel, the tape attendant is requested to put the next tape on.

Searching a particular permanent file tape begins with processing the first entry on the corresponding tape search queue. A forward search is started for the search number contained in the first entry. When the proper record is found, and the tape backspaced for read-

ing, buffer control call store tape blocks are seized for receiving the data, and a read is ordered. When the read is accomplished, the message is verified for station directory number and message number. A new message is built, using the message found and an indication that this is a retrieval copy. This message is then sent to the retrieval destination.

If the message requested is long (stored in more than one record) the search-backspace-read sequence is repeated for each segment until the entire message is found. If it is a short message packed in a record with one or two other short messages, the desired message is extracted from the record and processed.

These tape operations and the subsequent processing of the message found are repeated for each entry on the tape search queue of the tape being searched. Since retrieval queue entries are grouped according to ascending search number, all entries on a tape search queue can be processed in one efficient pass along the tape. At the conclusion of the search, the tape unit is returned to the proper state and search scheduling logic is entered to select the next tape search queue for processing.

4.3.6 *Retrieving From Message Store*

A master register address in the cross-reference file slot for a requested message number implies that the message is recent. It has not yet been filed on magnetic tape and is still located in the message store. The message is immediately retrieved from message store and sent to the retrieval destination.

4.4 *Journal File*

Selected data from each message is filed in journal file records on tape. Originating and terminating directory numbers, delivery status, dates, times, tape address of permanent file message copy, and message numbers are examples of this selected information.

All journal tape records are 15 blocks long, and each record is divided into 50 entries. Each entry is composed of a three-word search slot and a six-word text slot as illustrated in Fig. 8. A three-word search slot with its corresponding six-word text slot is used for each message originator. Similarly, each message terminator is assigned search and text slots. Thus a message sent to three addresses requires a total of nine words for the originator and 27 words for the terminators.
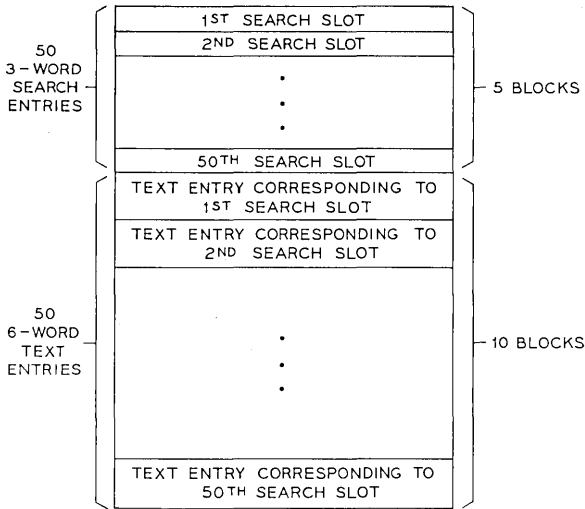
Fig. 8—Layout of data in the journal file tape record.

All information packed into the journal file record is obtained from the message processing block complex for each message. Message data is accumulated in a 15-block journal file buffer in buffer control call store. Data from individual messages is packed into the journal file buffer. When the buffer is full, its contents are written as a single record on the journal file tape.

The journal file record, unlike the permanent file record, in no way has a one-to-one relationship with the individual message. Rather, the journal tape record usually contains entries for many messages, one or two of which may overlap into adjacent records. The originator and terminators of an individual message always occupy contiguous slots in the record.

The search slot for each message originator and terminator contains the directory number, message number, and originator or terminator flag applicable to the message sender or receiver. The information contained in the search slot is unique to this message and thus forms the basis for search in a journal file retrieval. The corresponding text slot contains statistical information concerning the handling of the message by the No. 1 ESS ADF, such as delivery status, date, and time.

No. 1 ESS ADF journal records are useful because:

(*i*) They contain tape addresses of all permanent file message copies, and thus a message may be retrieved even after the cross-reference file entry has been overwritten in the message store. A journal file retrieval process is provided for this purpose.

(*ii*) Extracts of message processing records can be retrieved from the journal file tape by telephone company retrieval center attendants for assisting users and operating center personnel with special problems.

(*iii*) Statistical studies of operating performance, traffic levels, and specialized problems are reduced to final form by analyzing the journal records on commercial data processing systems. A journal file analysis program geared to a commercial computer is operational and is frequently used for this purpose.

4.5 *Journal File Retrieval*

The journal file retrieval service feature enables retrieval center personnel to obtain origination, termination, and processing information concerning messages handled by the No. 1 ESS ADF.

4.5.1 *Requesting Journal File Retrieval*

A journal file search is initiated by an input message called a journal file resend action request. This action request specifies the directory number of the station that sent or received the message, the message number, the journal file tape reel number, and the date and time that the recording was started on the journal file tape. The journal file tape reel to be searched must be off-line. Each journal file search action request implies a search over the entire reel for all entries pertaining to the message specified.

4.5.2 *Queuing Journal File Search Requests*

Since journal file retrieval requests cannot be processed immediately and since each search consumes considerable time, a journal file search queue entry is made for each search request. As in message retrieval, a four-word call store facility called a retrieval register is used. A new entry is added to the end of the queue, the queue consisting of a one-way link list of retrieval registers.

4.5.3 *Scheduling Journal File Tape Searches*

Journal file search scheduling is performed with the message retrieval tape search scheduling; journal file searches are alternated

with permanent file off-line searches. The same conditions and proce-
dures are followed as in setting up a permanent file off-line search.

4.5.4 *Searching A Journal File Tape*

When a journal file search is next in the search priority, a journal
file forward search is set up. Unlike a message retrieval search on one
word (the search number), a journal file search operation is based on
three words. The hardware matches on three-word entries within the
search section (the first five blocks) of each 15-block record on the
journal file tape. The first two words include the originator or termi-
nator flag and a ten-digit directory number. The third word is the
message number.

Processing the journal file record that is found involves locating the
search slots for all the entries associated with the entry for which the
search was made. The originator entry and all terminator entries for
this message are found. Since the entries for a given message may
extend beyond one record, additional tape operations may be required.
The contents of the text slot corresponding to each search slot is then
put into output format, one entry at a time, starting with the origina-
tor entry. Using this newly formed text, a journal file retrieval output
message is built and sent to the retrieval center.

Such a message contains data concerning origination, termination,
and handling of the message. Originator information includes the
originator's directory number, the message number, the date and time
of the origination, the originating message customer identity, and the
input message status. The termination data for each terminator con-
sists of the directory number, the message number, the terminating
date and time, the delivery status, and the delivery precedence. The
tape address of the complete message copy and the total message
length are included as handling information.

After a match has been made and the record processed for output,
another search is set up for the given directory number and message
number. The search-read-and-process sequence is repeated until all
entries have been found on the tape for the given directory number
and message number combination. When the end of the tape is reached,
the tape is rewound and reel change ordered, as appropriate for re-
maining off-line retrieval work.

V. ACKNOWLEDGMENTS

The authors wish to acknowledge the dedicated efforts and tech-

nical expertise of the many individuals at Bell Laboratories who have contributed to the development of the magnetic tape subsystem. The mutual cooperation and achievements of the numerous departments involved in the design of circuits and equipment, maintenance methods and programs, and operational software enabled the successful development of this system. Particularly noteworthy are the design achievements of Messrs. W. G. Reichert, Jr. and G. A. Van Dine for the tape unit control and tape unit hardware circuits, respectively.

REFERENCES

1. Ewin, J. C., and Giloth, P. K., "No. 1 ESS ADF: System Organization and Objectives," B.S.T.J., this issue, pp. 2733–2752.
2. Aitcheson, E. J., Ault, C. F., and Spencer, R. G., "No. 1 ESS ADF: Message Store—A Disk Memory System," B.S.T.J., this issue, pp. 2887–2913.
3. Aitcheson, E. J., and Cook, R. F., "No. 1 ESS ADF: Maintenance Plan," B.S.T.J., this issue, pp. 2831–2856.
4. Downing, R. W., Nowak, J. S., and Tuomenoksa, L. S., "No. 1 ESS Maintenance Plan," B.S.T.J., *43*, No. 5, part 1 (September 1964), pp. 1961–2019.
5. Genke, R. M., Harding, P. A., and Staehler, R. E., "No. 1 ESS Call Store— A 0.2-Megabit Ferrite Sheet Memory," B.S.T.J., *43,* No. 5, Part 1 (September 1964), pp. 2147–2191.
6. Connell, J. B., Hussey, L. W., and Ketchledge, R. W., "No. 1 ESS Bus System," B.S.T.J., *43,* No. 5, Part 1 (September 1964), pp. 2021–2054.
7. Kienzle, H. G., Nicodemus, K. L., Smith, M. T., Jr., Weber, E. W., and Zydney, H. M., "No. 1 ESS ADF: Message Processing Program Organization," B.S.T.J., this issue, pp. 2753–2829.

No. 1 ESS ADF:

Teletypewriter Stations and Transmission Facilities

By A. C. CARNEY, S. M. FITCH and G. PARKER

A solid-state electronic station controller, small enough to be housed within Model 33, 35 and 37 teletypewriters, administers station on-line procedures. To realize automatic message reception and transmission at the station, the controller samples all traffic exchanged between the line and the station teletypewriter, detects message-and device-control characters and generates and detects station-identity codes. In addition, the controller monitors the status of the station teletypewriter and provides appropriate alarms which are displayed on an attendant unit.

Stations are linked to the No. 1 ESS ADF office via the Bell System's private line telegraph network. New ac and dc data sets have been developed to accommodate speeds up to 300 bauds with lower distortion and crosstalk generation than previously attainable with former transmission apparatus.

I. INTRODUCTION

1.1 *Facilities*

With the advent of the No. 1 Electronic Switching System, Arranged with Data Features (No. 1 ESS ADF), the art of store-and-forward switching has been significantly advanced in the areas of reliability, capacity and feature capability. Interconnection between the newly designed stations and the message switcher is provided by existing Bell System and Independent Company private-line telegraph facilities. Only minor modification was required to modernize these facilities for use with the higher data rates of No. 1 ESS ADF.

The private-line telegraph system contains a large number of centrally located offices which provide both transmission and maintenance facilities. These locations are interconnected by a variety of carrier systems, utilizing channels derived from frequency division of voice channels shared with other subscribers. This sharing of facilities pro-

vides the subscriber with significant savings. At each location, the data signals of a channel are demodulated to baseband. Any number of individual baseband "legs" can be combined by an interconnecting circuit called a "hub." The basic function of a hub is to permit all outgoing legs to transmit what has been received on an incoming leg. As shown in Fig. 1, hubs can be interconnected to permit multistation arrangements or a hub may simply be used to interconnect two carrier channels, or to connect a final channel to the local loop of a station. In this way, hubs are well suited to the formation of party lines and broadcast systems.

Each hub has one additional leg, providing alarm, monitor and test circuits at the local service board. This expedites trouble locating and at times even makes it possible to anticipate service failures by the early detection of deteriorating transmission margins. Rerouting of circuits is often possible, ensuring prompt re-establishment of service. Due to these features, the private-line telegraph system has achieved high reliability and continuity of service. Over the years, a large network of shared hub-to-hub links, readily capable of meeting the growing needs of subscribers, has been established throughout the United States.

At times, the most efficient trunk utilization between two stations may require quite a circuitous route, i.e., many hub-to-hub links may be involved. To compensate for the accumulating distortion of the
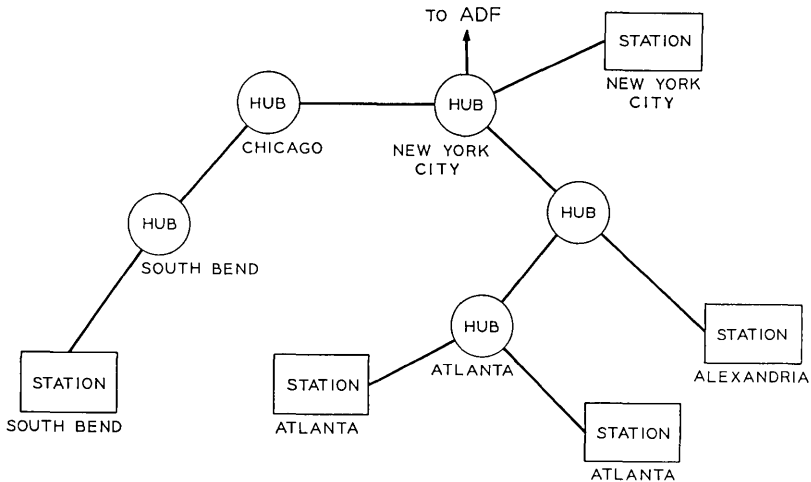
Fig. 1—Typical network illustrating use of hubs for a multistation arrangement.

many modulations and demodulations of the data signals, data-stream regenerators may be added at appropriate hub points. These regenerators restrict the network to a limited number of speeds and code structures.[1] The No. 1 ESS ADF code structures are compatible with such restrictions.

Although the system presently operates at a maximum of 150 bauds, new data sets have been developed for this system to provide data transmission at speeds up to 300 bauds between the hub and the station, anticipating the need for future increases in speed. At the same time these new data sets produce considerably less electrical noise on the loops than previous arrangements which operate at voltages compatible with vacuum-tube circuits.[2]

One office, near the No. 1 ESS ADF office, is designated as the controlling serving test center. Special data sets have been developed to provide a lower-cost arrangement for the link between this center and the No. 1 ESS ADF office, taking advantage of the fact that this link is expected never to exceed one mile.

1.2 Stations

Both half-duplex and full-duplex teletypewriter stations have been developed for this system. Half-duplex stations may send and receive message traffic sequentially, but not simultaneously, while full-duplex stations may send and receive messages simultaneously. Messages are transmitted from a station from punched paper tape and are received at a station as printed-page copy and/or punched-paper tape. The stations described in this paper use the USA Standard Code for Information Interchange (ASCII)[3] and operate at either 100 or 150 words per minute.*

1.2.1 Station Configuration

The functional form of the half-duplex station is shown in Fig. 2. The station controller conducts the control dialogue with the No. 1 ESS ADF processor (ADF)[4], directs the flow of data within the station and controls the operation of the teletypewriter. The teletypewriter sends and receives all message traffic and the attendant unit, consisting of lamps and keys, permits the station attendant to initiate actions at the station and observe the status of the station.

The attendant unit and controller are physically mounted within the

---

* Presently all stations operate at either 100 or 150 words per minute and use ASCII. However, ADF is arranged to accommodate stations that operate at 60, 75 and 100 words per minute and use the Baudot code.
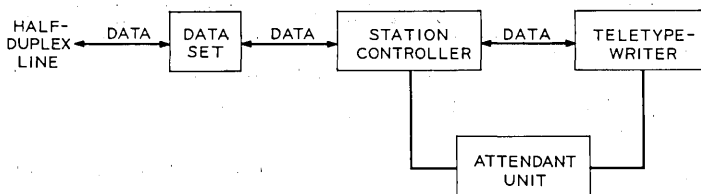
Fig. 2—Functional form of half-duplex station.

teletypewriter. Figure 3, for example, shows the front view of a half-duplex station equipped with a Model 33 automatic send-receive (ASR) and an optional Model 33 receive-only (RO) teletypewriter; this station many send and receive message traffic. The attendant unit is mounted to the right of the keyboard of the ASR. The RO is an optional secondary receiver which allows messages or any portion of them to be received at either or both the ASR or RO, thus permitting separation of received traffic into message categories. All message traffic originated at the station is printed as local copy on the ASR. In Fig. 4 the station controller is shown mounted in the pedestal of a Model 33 ASR.

The functional form of the full-duplex station is similar to the half-duplex station and is shown in Fig. 5. The station controller performs all of the functions mentioned previously but in addition permits the station to send and receive messages simultaneously. Because of this simultaneous operation, two teletypewriters (one send and one receive) are provided, whereas in the half-duplex station only one teletypewriter (send-receive) is necessary.

A typical send-receive full-duplex station equipped with Model 35 teletypewriters is shown in Fig. 6. The station controller is physically mounted in the ASR along with the send attendant unit. The RO has the receive attendant unit mounted in it and is shown equipped with an optional secondary receiver, a receive-only typing-reperforator (ROTR). A received message or any portion of it can be delivered to either or both the RO or ROTR while the local copy of traffic originated at the station is printed on the ASR.

1.2.2 *Controller*

The functional form of the half-duplex station controller is shown in Fig. 7. Data may be received from either of two sources, the half-duplex line or the teletypewriter, and since the station is half duplex,

these sources are never active simultaneously. The data is passed to
the character storage, detection and generation circuits where each
character received by the station is temporarily stored. Those char-
acters which require some action on the part of the station are detected
and the resulting information is passed to the logic circuits for process-
ing. Suppose, for example, ADF is "polling" the station, i.e., asking
the question "Do you have traffic to send?" This is part of the control
dialogue and is accomplished by ADF sending a prescribed character
sequence to the station. The characters are detected in turn by the
controller detection circuits and an appropriate response is de-
termined by the logic circuits based on the information available from
the attendant unit and teletypewriter. With the response determined,
the response character is generated and passed to the half-duplex line.

In addition to the detection and generation of control characters,
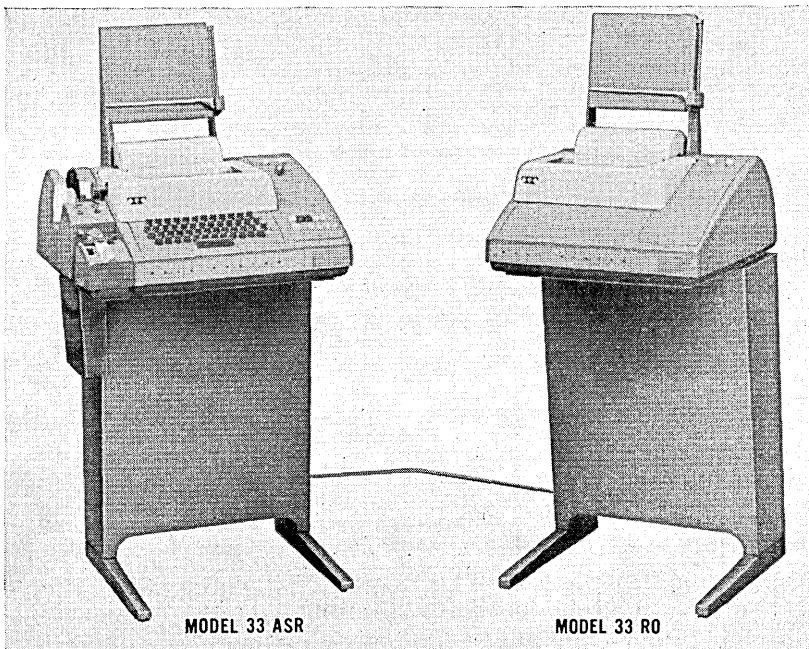
Fig. 3—Half-duplex station equipped with Model 33 teletypewriters.
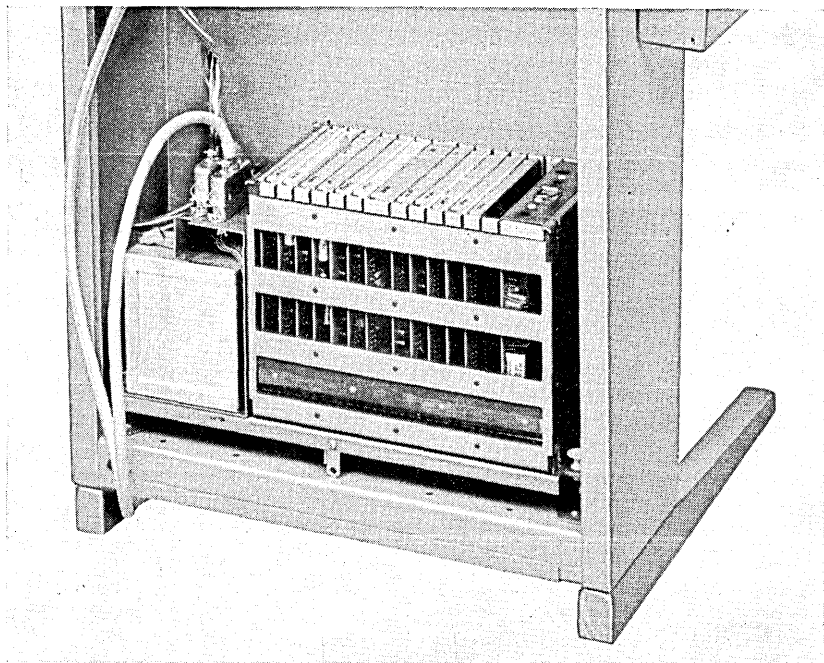
Fig. 4—Half-duplex station controller mounted in pedestal of Model 33 ASR teletypewriter.

the controller must also direct message traffic within the station. Consider, for example, a message at ADF which is intended for delivery to the station. ADF "selects" this station as a receiver using another prescribed control sequence and then delivers the message to the station. At the station, the logic directs this message data from the half-duplex line, through storage, to the teletypewriter. A more complete discussion of controller operation, including these functions, is presented in Section III.

The full-duplex station controller can be described in terms similar to those above except, as noted earlier, the controller must handle both message transmission and reception simultaneously. This means that while some of the functional blocks of the half-duplex controller are provided in duplicate in the full-duplex controller, the logical operation is similar.

1.2.3 *Attendant Unit*

The attendant unit is an assembly of lamps and keys which displays the status of the station and permits the station attendant to

control the operation of the station. By means of operating these keys, an attendant may, for example, make a request to send a message. The lamps indicate such states as selected to send, selected to receive, alarm conditions, etc.

1.2.4 *Teletypewriters*

Teletypewriters of the Model 33, 35 and 37 product lines of the Teletype Corporation were adapted for use in the stations. In all, eight different types of teletypewriters are available: two Model 33s, the ASR and RO; three Model 35s, the ASR, RO and ROTR; and three Model 37s, the ASR, RO and ROTR. The Model 33 and 35 teletypewriters are shown in Figs. 3 and 6, respectively. The Model 37 teletypewriters are shown in Figs. 8, 9 and 10. The Model 33 and 35 teletypewriters operate at 100 words per minute and the Model 37s at 150 words per minute and all use ASCII. The Model 33s are intended for limited usage while the Model 35s and 37s may be used continuously. Selection of the type of teletypewriter for use at a station depends on such items as station usage, machine features, cost objectives, etc.

II. TERMINATION TRANSMISSION LINKS

2.1 *Station Links*

ADF has a direct and permanently connected link to every station in its system. For the sake of economy, the channelized trunk routes between hubbing locations can be shared with other subscribers of the private-line telegraph system, and stations on multipoint lines of the No. 1 ESS ADF system can share a single channel, being connected
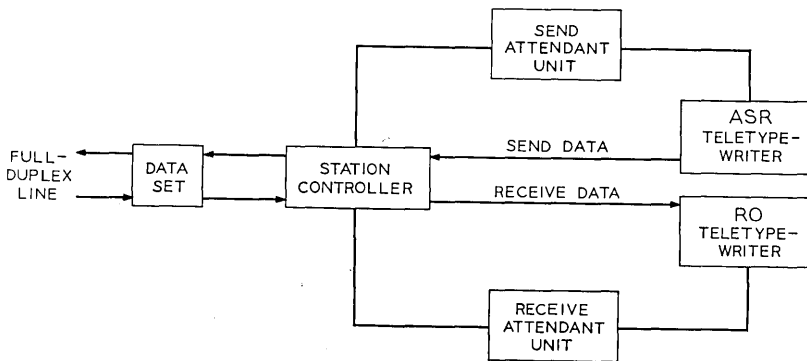
Fig. 5—Functional form of full-duplex station.
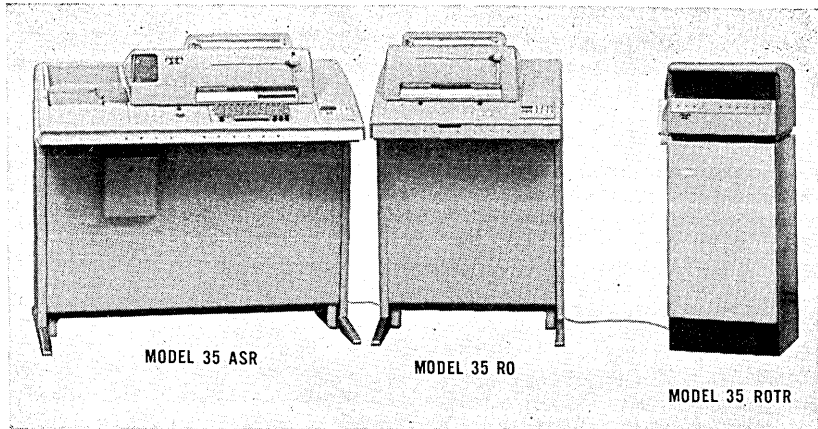
Fig. 6—Full-duplex station equipped with Model 35 teletypewriters.

to ADF on a "party line" basis. In contrast the last transmission link, between the station and its nearest telegraph office, is always individual to the station.

As previously mentioned, new end-link data sets were designed. The basic objectives of these new designs were: higher-speed capabilities than heretofore available for end links, less electric interference into other circuits on the same cable than some existing equipment, and simplicity of equipment to keep costs comparable to previous services.

Two new types of data sets for transmission links were introduced: one uses ac (voice frequency) transmission, the other polar dc.

2.1.1 DC Data Set

One new type of data set developed for the system, coded 109, uses polar dc transmission combining reliability of operation with low cost. The limitations of this method are the need for metallic cable pairs and the restricted range of operation, namely, 2500 ohm loop resistance, ranging between 6 and 15 miles depending on the gauge of wire used. Yet, because of the strategic distribution of service locations, the probabilities are high that a sizable portion of the stations of any system can be served by these dc facilities.

The functional form of the data set 109 is shown in Fig. 11. It contains a transmitter which acts as a low-impedance voltage source. The data set applies 4 volts between the wires of the line for a mark, and

12 volts of opposite polarity for a space. The data set at the other end of the line is similar, but is connected with opposite polarity to the metallic conductors. Thus, the voltage around the loop, including both data sets, is 8 volts for marking signals generated at both ends. If either end sends a spacing signal, the voltage around the loop changes to 8 volts in the opposite sense. Resistance padding is used to keep the total of line-plus-pad resistance constant. Thus, the loop current for two marking data sets is about 3 mA, while the loop current for one marking and one spacing data set results in current flow of 3 mA in the opposite direction around the loop.

Each set contains a pair of monitor resistors, one inserted in the "tip" and one in the "ring" leg of the transmission line. Under favorable conditions, the polarity of the voltage drop across either resistor indicates whether both data sets are marking, (i.e., current flowing in one direction) or whether one set is marking and the other spacing (current flowing in the opposite direction). In practice, longitudinal currents may flow along the line due to a ground potential difference between the two ends. As this potential difference may exceed the source voltage of the transmitters, the current in either monitor resistor may drop to zero or even reverse direction. On the other hand, by checking the voltage drop in both monitor resistances simultaneously with a differential circuit, the voltage drop due to the loop current can be found independently of the longitudinal current. This information then yields the data condition of the data sets.
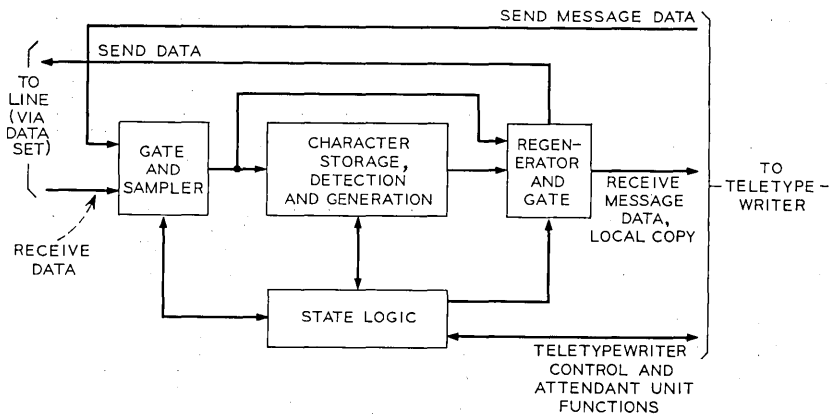
Fig. 7—Functional form of half-duplex station controller.
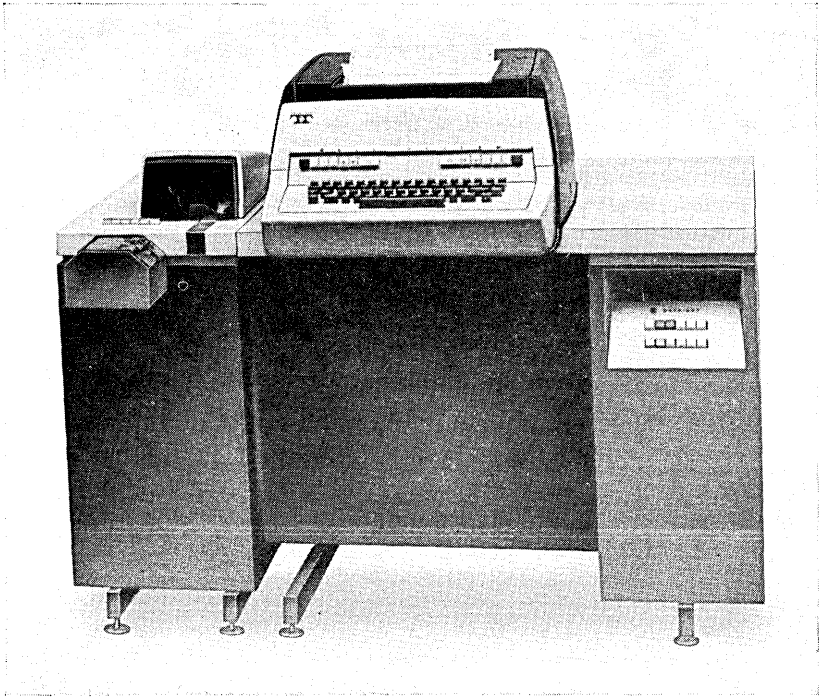
Fig. 8—Model 37 ASR teletypewriter.

Other circuitry detects loss-of-loop current, a condition which indicates a failure of the local portion of the transmission facilities. There is also provision to suppress "copy" of the transmitted signal in the receive lead of the same set. The entire circuit is contained on a single printed circuit board for easy installation at the station and in the telegraph office.

Extensive tests of data set 109 have yielded very satisfactory results. The set operates with less than 5 percent peak distortion on lines up to 2500 ohm loop resistance and under 1 μF capacitance at 150 bauds. It will work with proportional increases in distortion up to 300 bauds. The set can tolerate up to 20 volts dc and ac ground potential difference. The electrical interference generated by the set is sufficiently low not to adversely affect other circuits.

The data set 109 just described is restricted to half-duplex operation; however, a full-duplex version of this data set, operating over 2-wire

lines, is now in preparation and will be applicable to the No. 1 ESS
ADF system.

2.1.2 *AC Data Set*

Whenever full-duplex service is required, or whenever the distance
between the last hubbing location and the station is such that the
limits for dc transmission would be exceeded, ac (voice frequency)
transmission is used. A new type of data set was designed to permit
operation at the speeds of the No. 1 ESS ADF system.

Fig. 9—Model 37 RO teletypewriter.
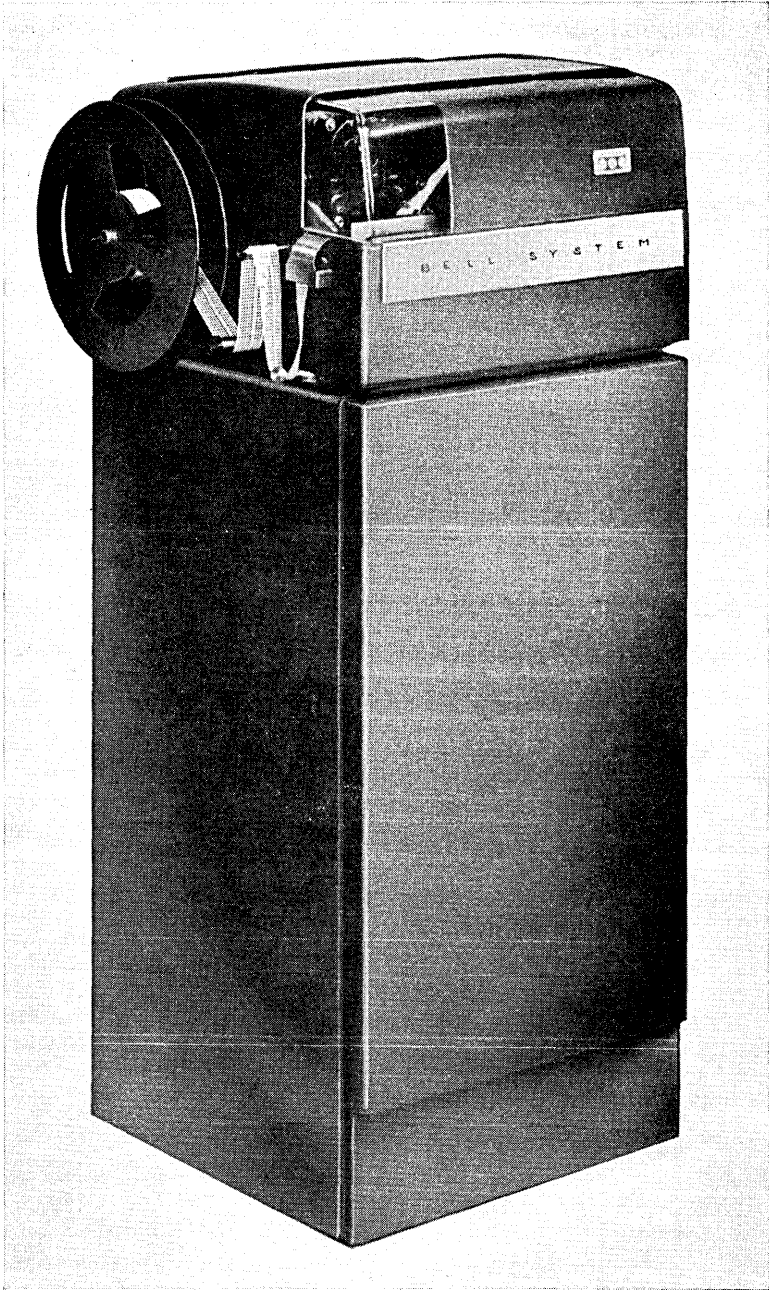
Fig. 10—Model 37 ROTR.
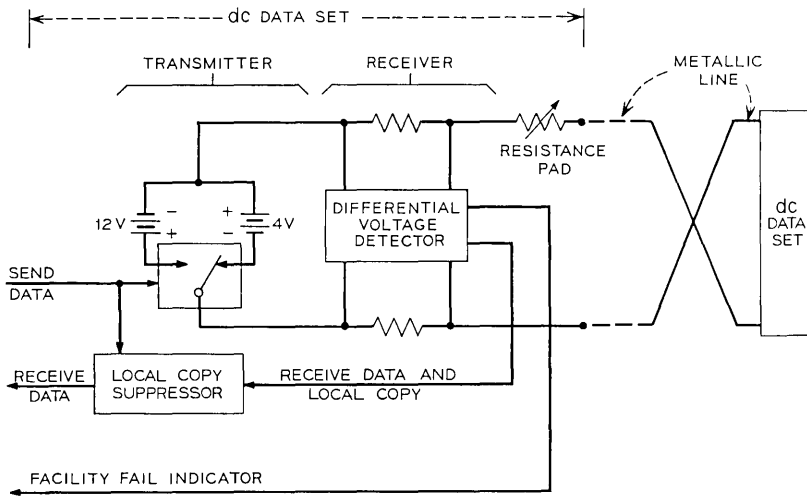
Fig. 11—Functional form of dc data set.

This data set, coded 108, contains a frequency-shift modulator, generating 2125 Hz ±100 Hz and 1170 Hz ±100 Hz at the station and at the hub, respectively. These frequencies are used in Teletype-writer Exchange (TWX) and DATA-PHONE® services and have been maintained for private-line telegraph end links, so that the test facilities in central offices can be shared. Receive circuitry employs conventional FSK demodulating techniques. The send-and-receive frequencies of the station set match the receive-and-send frequencies of the central-office set. The data-set circuitry is contained on a single printed circuit board, which is physically interchangeable with data set 109. Data set 108 is shown in Fig. 12. The peak distortion of data set 108 at 150 bauds is less than 10 percent over transmission lines with up to 30 dB end-to-end loss at a signal-to-noise ratio of 10 dB. With a proportional increase in distortion, it will work up to 300 bauds.

2.1.3 *High-Voltage Circuits*

The data sets described above have input and output circuits in the typical low-voltage range of transistor circuits: under 24 volts. At the hubbing location, data sets 108 and 109 must connect to the existing high-voltage hub circuits which were designed for vacuum-tube circuits with a +130 volt plate supply. An auxiliary circuit has been designed to provide the proper voltage conversion between hub and data set circuits. As data sets 108 and 109 have the same characteristics at
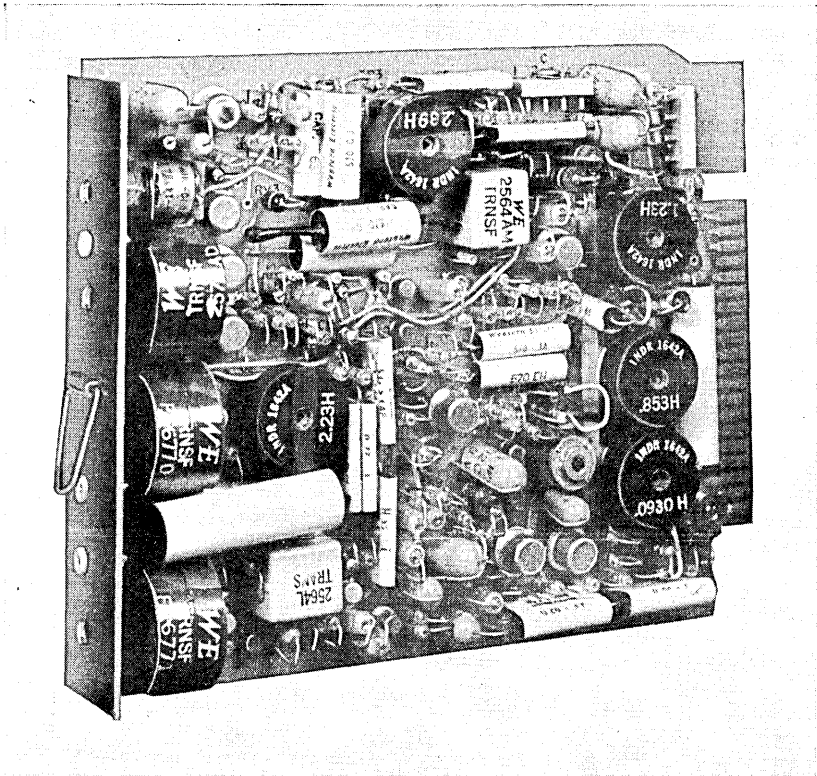
Fig. 12—Data Set 108 type.

their baseband interface, the same auxiliary circuit is used for either set.

This circuit, coded data auxiliary set 811C, contains multistage conversion circuits to effect the necessary step-up and step-down in voltage. The auxiliary set also contains the directional control for use at half-duplex hubs to prevent copy received from its data set from being sent back to the same data set by the hub.

2.2 Office Link

The ADF office is always located close to the first hubbing location which is its serving transmission center, usually in the same building. It was therefore possible to utilize a dc transmission scheme without ground potential compensation. To accommodate full-duplex stations,

the transmission on this link has to be full-duplex also. This is achieved by the use of 4-wire facilities between the serving office and the ADF office, keeping data-set circuits as simple as possible.

At the serving office a single circuit board, containing the dc data-set circuit as well as the low-voltage to high-voltage conversion circuits, is used. At the ADF office only a low-voltage interface is needed and the conversion circuits are not required. Therefore, the dc data set may be mounted on one of the smaller No. 1 ESS circuit boards which permits 256 data sets to be accommodated on a single frame.[5,6] Facility interruption is indicated at both ends of the link by detection of the loss of dc line current.

III. STATION OPERATION

3.1 *Character Format*

As in most teletypewriter systems, asynchronous character timing is employed. In this mode of operation, referred to as start-stop, a fixed-timing pattern is used for each group of bits representing a character. Each group is preceded by a start element which serves to
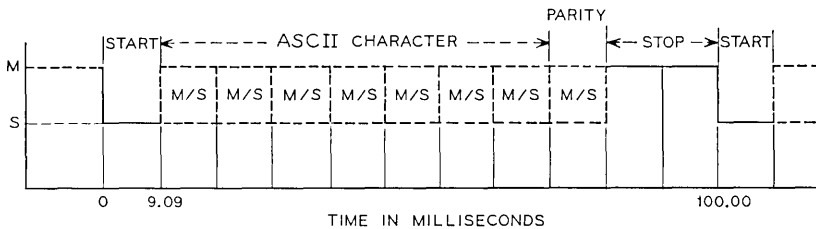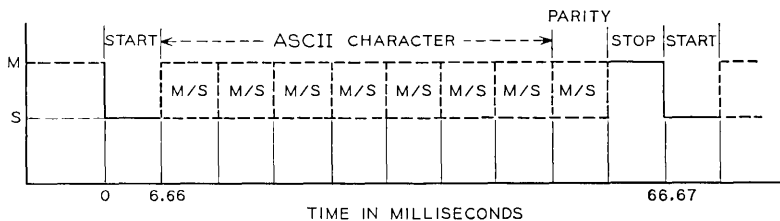
Fig. 13—Character format of teletypewriter signals. (a) Format at 100 words per minute. (b) Format at 150 words per minute. M = mark, S = space, M/S = mark or space.

indicate when a fixed pattern is to start. The character format for 100 words per minute (10 characters per second) and 150 words per minute (15 characters per second) is shown in Fig. 13. Each character consists of a spacing start element, a seven-bit ASCII character, a parity bit, and two (100 words per minute) or one (150 words per minute) marking-stop element(s). The parity bit is chosen (mark or space) so that the eight-bit character consisting of the ASCII character and the parity bit, contains an even number of marking bits.

The station controller is automatically sequenced through message handling operations by selected control characters, a subset of the

TABLE I—USA STANDARD CODE FOR
INFORMATION INTERCHANGE (ASCII, USAS x3.4—1967)

Bit Value				$b_7 \longrightarrow$ $b_6 \longrightarrow$ $b_5 \rightarrow$	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b_4 ↓	b_3 ↓	b_2 ↓	b_1 ↓	Column ⟶ Row ↓	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	`	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	,,	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	10	LF	SUB	*	:	J	Z	j	z
1	0	1	1	11	VT	ESC	+	;	K	[	k	{
1	1	0	0	12	FF	FS	,	<	L	\	l	¦
1	1	0	1	13	CR	GS	—	=	M	]	m	}
1	1	1	0	14	SO	RS	.	>	N	∩	n	~
1	1	1	1	15	SI	US	/	?	O	__	o	DEL

ASCII alphabet. Some of these characters are part of the normal message format and some are transmitted out of message context, for line-control purposes. The complete ASCII character set is given in Table I. Characters listed under columns 0 and 1 in the table are defined to be control characters.

3.2 *Sending from the Station*

Messages for transmission from the station are prepared on paper tape using the teletypewriter keyboard and tape punch. Messages must possess the following format:

$$\cdots \begin{matrix} DS \\ EO \\ LH \end{matrix} \quad \text{Heading} \quad \begin{matrix} S \\ T \\ X \end{matrix} \quad \text{Message Text} \quad \begin{matrix} ED \\ TE \\ XL \end{matrix} \cdots$$

The message heading, separated from the message text by the control character STX (Start of Text), includes the addresses of those stations that are to receive the message text. The control characters SOH (Start of Heading) and ETX (End of Text), denoting the beginning and end of the message, together with STX, are detected by ADF and the station controller to initiate message handling sequences. To provide a tape leader that facilitates introduction of the prepared tape into the teletypewriter tape reader, a number of DEL (Delete) characters are punched in the tape prior to SOH. Several messages may be sent by the station during a single transmission. In this case, additional messages with the prescribed format are prepared on the same tape with DEL characters used to physically separate the messages. An EOT (End of Transmission) control character is punched in the tape following the ETX of last message to denote termination of the transmission.*

To initiate the sending sequence, the tape containing the prepared message is inserted into the tape reader, and the tape reader is started by the controller when a key (BID) on the station attendant set is operated. Alternatively, at stations equipped with Model 35 or 37 teletypewriters, tape may be fed directly into the tape reader from the punch and the reader started automatically when the EOT character is punched in the tape. In this case, a differential counter in the station controller is incremented each time an EOT is punched during

---

* Since certain control characters automatically initiate message control sequences, both at ADF and at the station, they may never appear in the message heading or text. These control characters are SOH, STX, ETX, EOT, ENQ (Enquiry) and DLE (Data Link Escape).

message preparation, and decremented each time an EOT is detected by the controller during actual transmission to the line. The advantages of this feature will be noted in subsequent sections.

The controller discards the DEL characters and stops the tape reader upon detection of the SOH character. At this point the controller logic is now conditioned to generate and transmit control characters indicating that the station has traffic to send upon interrogation by ADF.

3.2.1 *Polling*

ADF periodically interrogates each station on a line in order to determine if the station sender has traffic to originate and if the station receiver is ready to receive traffic. The polling cycle begins when ADF transmits the control character DLE (Data Link Escape). This character causes all station controllers on the line to assume the polling state. Programmable character detection circuits in the station controller permit each station on the line to be assigned a unique station-polling code which is selected from the set of ASCII printing characters.* Upon receipt of the appropriate polling code from ADF, the controller generates and transmits a 1- or 2-character sequence, depending on the status of the station. The possible responses and their significance are as follows:

CAN (Cancel) : this character indicates that the station has no traffic to send, but is ready to receive.

NAK (Negative Acknowledge) : NAK is transmitted if the station has no traffic to send and is not ready to receive. The station is defined as "not ready" for any of the following reasons:

(*i*) The receiving† teletypewriter does not have sufficient paper (or tape, in the case of an ROTR).

(*ii*) The receiving teletypewriter is in the process of feeding form‡ or tape, a process that may have been initiated as the station returned to the idle state after receiving a previous message.

---

* Characters included in Columns 2 through 7 of the ASCII code, except characters SP and DEL, are printing characters. The characters P, R, W, X and + have special significance in the operation of the controller and therefore are not usable as station-polling codes.

† Most of the operations described are common to both full- and half-duplex stations; where operations are not the same in both cases, the differences will be noted. When reference is made to the sending or receiving teletypewriter it should be understood that in the case of a send-receive half-duplex station, these teletypewriters are one and the same.

‡ ASR and RO teletypewriters may be equipped with either a typing unit that uses paper on rolls or one that uses fan-folded paper forms.

(*iii*) The station has been put out-of-service using a key on the attendant set.

R ACK (Regular Acknowledge): this two-character sequence is generated if the station has regular (nonpriority) traffic to send and is ready to receive.

P ACK (Priority Acknowledge): the meaning of this sequence is similar to that for R ACK, except the available traffic in this case is of priority level. The priority level is assigned by the station attendant using a key on the attendant unit.

R NAK (Regular Negative Acknowledge): these characters indicate regular traffic available but the receiving teletypewriter is not ready to receive.

P NAK (Priority Negative Acknowledge): similar to R NAK, except that traffic is of priority level.

On half-duplex lines, if none of the stations indicate traffic available, ADF can return all stations to idle by transmitting a control character. If all stations on a full-duplex line respond "no traffic available," ADF transmits a control character which causes the sending logic of the controller to assume the "cocked" state. A programmable character-generation circuit in the controller permits each station on a line to be assigned a unique station-identity code, selected from the set of printing ASCII characters, with the same exceptions as noted for the station-polling code. When the full-duplex controller is in the cocked state, it will automatically generate and transmit this code when an SOH character is detected from the station's tape reader. ADF detects the activity on the line and interprets this as a polling request. Ordinarily, ADF would not discover that a station on the line had traffic to send until the next periodic polling cycle. Polling subtracts from the time available to ADF for message delivery, therefore, the cocking feature increases the efficiency of full-duplex lines. The efficiencies of the cocking feature cannot be realized on half-duplex lines since interference would result at the half-duplex hub if a station attempted to send at the same time ADF was sending.

On both half- and full-duplex lines, when a station indicates traffic available, ADF will normally proceed to the message pick-up operation, as described in the next section.

3.2.2 *Message Pick-up by ADF*

The programmable character-detection circuits in the station controller also permit each station on the line to be assigned a different call-enquiry code chosen from the ASCII printing characters, with the

same exceptions as noted for the station-polling code. After identifying those stations with traffic available during the polling cycle, ADF begins the traffic pick-up sequence by transmitting the control character ENQ (Enquiry) followed by the call-enquiry code of the station to be selected as a sender. When the controller detects this, it automatically generates and transmits one of two possible control characters. If SOH has been detected from tape, as described in Section 3.2.1, the controller is in the traffic-available state and will regenerate and transmit SOH, causing the station to be selected as a sender and a lamp (TRAN) to be lighted on the attendant unit of the sending teletypewriter. However, if the traffic available state has been cancelled, by removing the tape from the tape reader for example, the controller responds with the control character NAK. If traffic is still available, i.e., the station responds with SOH, ADF transmits appropriate control characters causing the controller's data-gating logic to be conditioned to pass data received from the line to the typing unit of the sending teletypewriter. ADF may now transmit an originating message number for message identification as well as the time and date. At this point it is necessary to differentiate between half- and full-duplex operation.

On a half-duplex line, traffic can originate from only one source at any given time, either from ADF or from a sending station. This being the case, in addition to messages being forwarded by ADF, direct station-to-station communication or intra-line delivery under control of ADF is permitted. On a full-duplex line, however, both a station and ADF may originate traffic simultaneously; therefore, to prevent interference, messages are handled on an interline basis only, i.e. all messages are stored and forwarded by ADF. The remainder of the description of message pick-up will be divided into two parts. Half-duplex operation will be discussed first.

3.2.2.1 *Half-duplex.* After transmitting the time, date and message number, ADF sends the go-ahead control character STX. The controller detects STX and starts the station tape reader. The controller's data-gating logic applies the output of the tape reader to the station data set for transmission to the line. The controller also unblinds the station printer so that local copy of the message is obtained. The controller monitors the transmission and stops the tape reader when STX, the control character that follows the message heading, is detected. Stopping the reader at this point in the sequence provides an opportunity for ADF to determine if any stations on the same line as the sending station are designated as addressees of the message. If this is the case, ADF will call in these stations, i.e., select these stations

as receivers, (see Section 3.3.1) provided of course they have indicated that they were ready to receive during a previous polling cycle. ADF then restarts the sending station's tape reader and the message text is transmitted. Upon detection of ETX, the control character denoting the end of the message, the controller again stops the tape reader. The called-in intraline stations receive the message directly from the sending station, and ADF stores the message for future delivery to the interline addressees. ADF "roll-calls" (see Section 3.3.2) all called-in receive stations to determine if they have received the message properly and then unselects them as receivers. ADF again transmits the go-ahead signal, and the tape reader is turned on once again at the selected-to-send station. If the station has additional messages to transmit, the reader continues to advance the tape until the controller detects and transmits the SOH character denoting the beginning of a new message. ADF will restart the tape reader and repeat the pick-up process described above. If no additional messages are available, ETX on the tape is followed by EOT. The EOT is transmitted by the controller and the station becomes unselected as a sender and returns to the idle state. If the controller's EOT counter indicates zero, the controller stops the tape reader, otherwise the tape is permitted to advance until the next SOH is detected. SOH stops the reader and cycles the controller logic to the traffic-available state.

3.2.2.2 *Full-duplex.* On a full-duplex line, after transmitting the originating time, date, and message number to the selected-to-send station, ADF transmits control character DC2 (Device Control 2). The controller detects DC2, starts the tape reader, and restores the receive portion of the line to the receive-only teletypewriter. Since intraline delivery is not possible on full-duplex lines, the controller, in contrast to half-duplex operation, does not stop the transmitter until both the heading and the message text are transmitted, i.e., until the controller detects ETX from the tape. Local copy is provided on the typing unit of the sending teletypewriter. Except for the different go-ahead control character, DC2 instead of STX, the remainder of the full-duplex pick-up sequence is identical to that for half-duplex. After message pick-up has been completed ADF polls a full-duplex line in order to determine if traffic has become available during the pick-up process.

3.2.3 *Hold Feature*

In the process of sending many messages during a single transmission it may be desirable to transmit a message of higher precedence

from a separate piece of prepared tape. The hold feature allows the station attendant to remove the in-progress tape and introduce a "torn-tape" message without affecting the station's status as a selected sender. When the HOLD key on the attendant unit is operated, the controller inhibits ADF from restarting the tape reader after it has been stopped following detection of ETX. When the tape reader stops, the HOLD key lamp lights indicating that the original tape may be removed and the new tape inserted. The station remains selected as a sender and the station attendant may restart the tape reader. To reintroduce the original tape, the hold process is repeated.

3.2.4 *Emergency Stop*

During transmission from the station, ADF checks for invalid address information and errors in message format. If a violation is detected, ADF initiates an emergency stop sequence. This procedure stops the station's tape reader and activates an alarm; ADF then delivers a service message to the offending station indicating the nature of the error.

ADF interrupts a half-duplex sender by transmitting a break signal, i.e., several character intervals of steady spacing followed by a sequence of control characters that unselects the station as a sender, activates an alarm and unblinds the station receiver. After delivery of the service message ADF restores the station to the idle state.

In the case of a full-duplex station, ADF transmits a sequence of control characters that interrupts the sending station and blinds all called-in receive stations on the line. After sending the service message, which is copied by the typing unit of the sending teletypewriter, ADF sends a sequence that unselects the sending station and unblinds any selected receivers and then resumes delivery of the message to the selected receivers.

3.3 *Receiving at the Station*

3.3.1 *Call-In and Message Delivery*

If ADF has a station marked ready to receive as a result of the polling cycle, the station is eligible for selection as a receiver, an operation referred to as "call-in." To select a station ADF transmits a two-character sequence, ENQ followed by the station's unique call-enquiry code. This sequence is identical to the sequence used to select a station as a sender but is distinguishable since the controller is not in the polling state.

As mentioned previously, full-duplex stations are arranged to send and receive traffic simultaneously. During the course of receiving a message, the station controller must generate and transmit control characters in response to interrogation by ADF. When the station is in the process of sending, the control character ENQ interrupts the transmission permitting the receive portion of the controller to access the sending logic in order to generate and transmit the necessary responses.

ENQ followed by the station's call-enquiry code elicits a two-character response from the controller, the station-identity code followed by ACK or NAK. The station-identity code is used by ADF to verify that the response originated from the station that was called. NAK indicates that the station is not ready to receive for reasons previously described in Section 3.2.1. Further attempts to call-in a station responding with NAK will not be made until the station is found to be ready during a subsequent polling cycle. When the controller generates the NAK response, a lamp (CALL) lights on the attendant unit of the receive teletypewriter to alert the station attendant to the call-in attempt.

The station response ACK, i.e., ready to receive, automatically selects the station as a receiver, causing a lamp (REC) on the attendant unit of the receive teletypewriter to light. ADF may now transmit any per-station heading information such as a receive-message number. If additional stations on the same line are designated as recipients of the message, ADF transmits ENQ followed by the call-enquiry code of the second station. ENQ blinds the receiver of the first station ensuring private delivery of per-station information to the second station. To call in additional stations the process described above is repeated. When call-in has been completed, ADF transmits the sequence ENQ DC2 and unblinds all called-in stations. DC2 also reactivates any sending full-duplex station that was interrupted when call-in was initiated. ADF now transmits the text of the message followed by the control character ETX.

3.3.2 *Roll-Call*

As stated previously, ADF unblinds all selected station receivers prior to message delivery by transmitting the sequence ENQ DC2. DC2 also enables that part of the controller logic that monitors "proper" reception of the message. This logic state is referred to as "roll-call." Specifically, any one of the following occurrences indicate improper reception:

(*i*) A character received by the controller did not reach the receiving teletypewriter.

(*ii*) A receiving teletypewriter that utilizes paper forms ran out of forms before the end of the message.

(*iii*) The control character ETX was not detected or was received with a parity error.

After transmitting ETX, ADF interrogates each station in turn using the sequence ENQ followed by the station's call-enquiry code which, since the controller is in the roll-call state, is interpreted as the beginning of the roll-call sequence rather than either a call-in or select-to-send sequence. The controller generates and transmits a 2-character response, the station-identity code followed by either CAN or NAK. CAN indicates proper reception; NAK indicates improper reception and activates an alarm on the attendant unit. After all stations are roll-called, ADF transmits EOT which restores all stations to the idle state. In the event a station responds NAK to roll-call, ADF makes another attempt to deliver the message. If the second attempt is not successful ADF passes the message to a network control station for appropriate action.[4] On full-duplex lines, DC2 is sent after EOT to reactivate an interrupted sending station, if any.

3.3.3 *Receive Message Abort*

This feature allows ADF to interrupt message delivery to a station and activate a station alarm if an irregularity is detected in the message or if some other system anomaly occurs.

On a full-duplex line ADF transmits a sequence of control characters that interrupts any selected sender and activates an alarm at selected receive stations. ADF then transmits a service message indicating the reason for the delivery interruption, then idles all selected receive stations and restarts the interrupted sending station.

On half-duplex lines ADF sends a sequence of control characters that activates an alarm at the selected receive station and in the case of an intraline delivery, also interrupts the tape reader at the sending station and activates an alarm (see Section 3.2.4). After sending the service message, ADF restores the sending station, if any, and all selected receive stations to idle.

3.4 *Controller Operation*

3.4.1 *Character Processing*

All data exchanged between the station teletypewriter and the line

is monitored, temporarily stored, and then regenerated by the station controller. The character-processing circuits perform the required timing, storage, control character detection and response generation. The functional form of this circuit for the half-duplex controller is shown in Fig. 14. To facilitate simultaneous reception and transmission, the full-duplex controller contains two character-processing circuits.

Each character, from either the teletypewriter tape reader or the receive-data lead of the data set, initiates a new timing cycle and is stored in a 10-stage magnetic core shift register. Special windings threaded through the cores of the shift register permit detection and generation of the necessary control characters. During message transmission, characters are read into the shift register and applied to the line simultaneously. However, the shift register is in series with the received data stream; therefore, a character received from the line is stored in the register until shifted out by the timing cycle initiated by the following character. By introducing a delay of one character interval during reception, sufficient time is gained to permit the use of gating circuits to prevent sequence-control characters transmitted by ADF from reaching the station teletypewriter.

Pulses from the character-detection circuits sequence the controller's state logic through the various operational modes. Status signals from the station teletypewriter and attendant unit are monitored by the state logic and exercise control over the writing and gating circuits.

3.4.1.1 *Timing*. Three timing signals are necessary for character processing. One, derived by the character timer, defines the character interval; the second, provided by the bit clock, defines the element or bit* interval. The third will be discussed in Section 3.4.1.2.

In the idle state, the input-data lead is marking. A mark-to-space transition, denoting the beginning of a character, initiates the timing cycle by starting the character timer, the bit clock, and the clock-guard timer. The output of the bit-clock circuit is a square wave; the period of one cycle of the square wave is equivalent to a bit interval. The bit clock is permitted to run for ten cycles before being squelched by the character timer. Data sampling and shifting operations are timed by the bit clock.

The shift register is divided into two parallel 10-stage information stores. One stores marking bits and the other stores spacing bits. A

---

* In the strict sense, not all the elements of the teletypewriter character may be defined as "bits". For convenience, however, the term will be used in this discussion.
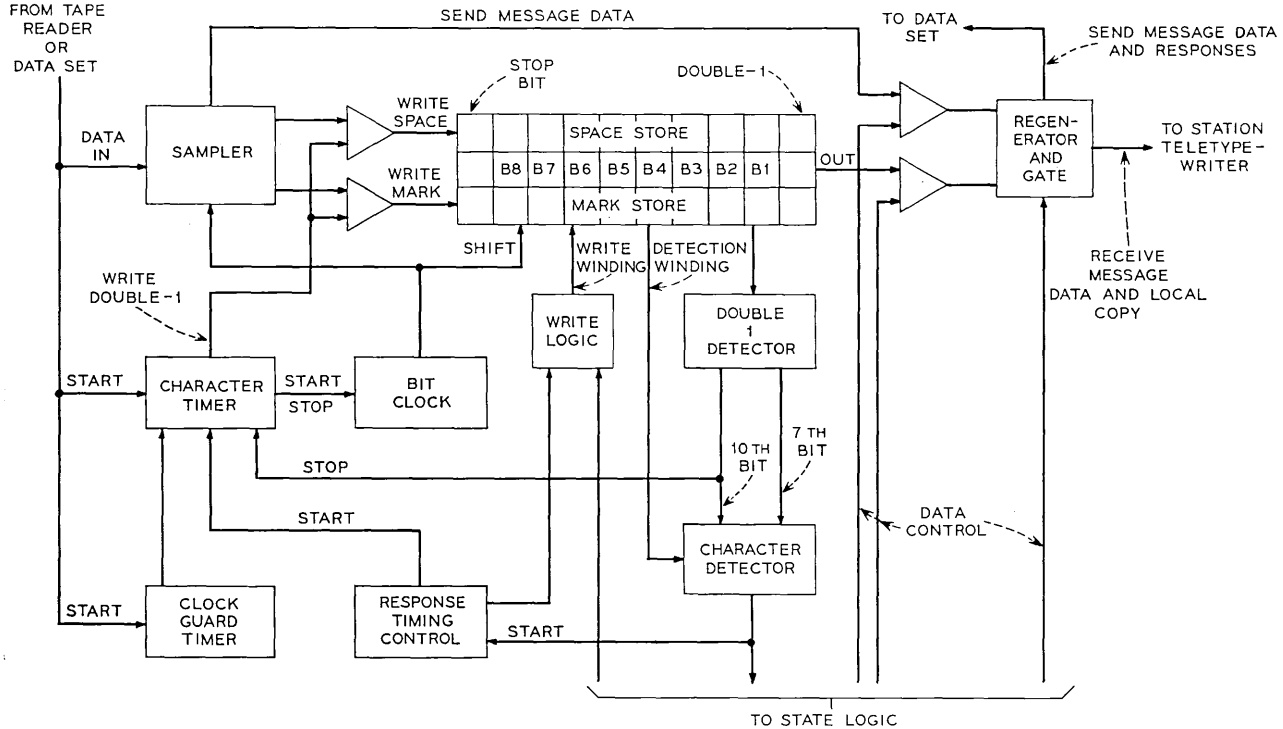
Fig. 14—Functional form of half-duplex character processing circuit.

logical 1 is entered into the appropriate store as each bit following the start bit is sampled. Coincident with the start bit, however, a "double-1" is entered into the register; i.e., a 1 is entered into both the mark and space stores. As each of the remaining nine bits (seven-bit ASCII character, parity bit, and one-stop bit) is sampled and entered into the register, the double-1 propogates through the register and is detected as it enters the tenth position, coincident with the center of tenth-timing cycle. When the double-1 is detected, a pulse is fed back to the character timer and terminates the timing cycle. The clock guard timer, which is reset at the beginning of each timing cycle, prevents the bit clock from free-running in the unlikely event a double-1 pulse is not detected or is mutilated.

The possibility exists that the bit clock may be falsely started, by a hit due to line noise for example. The write and detect double-1 sequence still occurs and the normal timing interval is maintained. Unless certain repeated characters are received causing the timing circuit to synchronize on some mark-to-space transition other than the normal-start transition, synchronization is quickly regained. In order to reduce the number of incorrect characters that are printed or punched while the receiver is recovering synchronization, the received data is sampled at the end of the timing interval, coincident with what would normally be a marking stop bit. If the sample is found to be spacing, the received character is changed to an "underline" (_) character by the write circuit before it is passed to the teletypewriter. In addition, characters received from the line are also checked for parity. Those having incorrect parity are also changed to an underline.

3.4.1.2 *Character Detection and Generation.* A separate wire, which acts as a detection winding, is threaded through the cores of the shift register for each character to be detected. Each wire is threaded in a unique manner, governed by the mark-space bit pattern of the character to be detected. Except for the SOH, ETX and STX windings, the wires are arranged so that the characters are detected as the stop pulse is written into the shift register. As described in Sections 3.2.1 and 3.2.2, the controller is required to stop the station's tape reader when message-format characters are detected from the tape; these stop characters are SOH, ETX and, in the case of half-duplex operation, STX. The station's tape reader must be stopped before reading the character following the stop character. To gain time, therefore, the windings for the stop characters are arranged to permit detection as the seventh ASCII bit is written into the register, without regard for the parity bit. The presence of a character that agrees with the

winding pattern of a detection winding is indicated by a negative pulse on the winding output, provided it occurs at the appropriate point in the timing cycle. Other characters yield positive pulses. In order to sample the detection windings of the stop characters, a third timing signal is derived by an additional double-1 detector which is arranged to detect the double-1 coincident with the center of the seventh ASCII bit; the other detection windings are sampled by the same double-1 pulse employed to terminate the timing cycle. Pulses from the character-detection circuits are applied to the controller's state logic to initiate automatic sequencing of the various operational modes.

When the state logic is conditioned to the appropriate mode, the detection of certain control characters will initiate a response sequence. The response-timing control circuit is triggered by a signal from the state logic; the state logic also indicates whether a 1- or 2-character response timing cycle is required.

A separate write winding is threaded through the shift register cores for each of the characters to be generated. A current driver is associated with each write winding and the desired character is written into the register by applying a narrow current pulse to the appropriate winding. To do this, the response-timing control circuit applies a pulse to the write logic. The pulse is gated under control of the state logic to the appropriate current driver and the character is written into the shift register. After a suitable delay, the response-timing control circuit applies a pulse to the character timer and a timing sequence is initiated. The character residing in the register is shifted out, passed to the data set and transmitted. If the state logic indicates that a second character is required, the response timing control is enabled by the character timer at the end of the timing cycle and the write and shift process is repeated.

3.4.2 Alarms

A number of alarm conditions are detected by the controller. Lamps on the attendant unit indicate the nature of the alarm condition; an audible alarm is also provided by a loudspeaker housed in the attendant unit. A list of the alarms and their meaning is given below.

(i) TAPE: This alarm monitors the status of tape in the teletypewriter tape reader while the station is selected as a sender. The alarm is activated if the tape reader runs out of tape or if the tape becomes taut. The latter condition is likely to occur when tape is fed directly into the tape reader from the punch with insufficient

slack. The alarm also occurs if the tape is removed from the reader or if the switch on the reader is moved from the RUN position. This alarm circuit is disabled when the hold feature described in Section 3.2.3 is in effect.

(*ii*) EMERGENCY STOP: This alarm occurs when ADF initiates the emergency-stop sequence described in Section 3.2.4.

(*iii*) PAPER LOW: For teletypewriters that use paper on rolls, the alarm condition is indicated when the diameter of the paper supply roll is reduced to a prescribed dimension. In the case of teletypewriters that use paper forms, the alarm is not given until forms are depleted.

(*iv*) TAPE LOW: This alarm is provided to indicate a low tape-supply condition when the station is equipped with an ROTR.

(*v*) MESSAGE RECEPTION: A message-reception alarm is given whenever a station responds NAK to roll-call indicating "improper" receipt of a message. The conditions that elicit this response are described in Section 3.3.2. In addition, this alarm is triggered at a selected receive station if ADF interrupts message delivery in order to transmit a service message.

(*vi*) ERROR: This alarm occurs if a parity error is detected or if the normally marking stop element is found to be spacing in a character received from the line. In addition to generating the alarm, the controller changes the affected character to an underline character with incorrect parity before it is gated to the teletypewriter.

The alarm lamps are located under translucent nonlocking keys on the attendant unit; by depressing the lighted key the alarm lamp and audible tone may be turned off. The PAPER LOW and TAPE LOW alarms, however, cannot be canceled until the paper or tape supply is replenished. A separate locking key is provided to allow the attendant to disable the audible alarm.

3.5 *Maintenance Features*

3.5.1 *Local Tests*

Model 33, 35 and 37 ASR teletypewriters are equipped with a mode switch. Using this switch the teletypwriter may be switched off-line, isolating the machine from the station controller. When off-line, the typing unit and punch operate from either the keyboard or tape reader allowing the operator to check basic machine functions. When the station is polled while in the off-line mode, the full-duplex controller will respond "no traffic available;" half-duplex stations equipped

with an ASR teletypewriter will, in addition, respond "not ready to receive." If an attempt is made to select a half-duplex station as a receiver while the ASR is off-line, the CALL lamp on the attendant unit will light, alerting the station attendant to the call-in attempt.

3.5.2 Remote Tests

When the station is equipped with an ac data set, a switch (MTCE) located on the controller may be used to interconnect the send and receive data leads of the data set. Thus, with the remainder of the controller and the station teletypewriter isolated, tests of the transmission facility and data set may be performed from a remote test center.

Remote tests of many controller functions may be performed using an automatic loopback feature provided by the controller. If the controller is equipped with the ac data set, it may be conditioned to the automatic loopback mode from a remote test facility by transmitting a prescribed sequence of control characters. The sequence forces the station to the "ready" mode and permits the station to be selected as a receiver using the normal call-in process. The controller's state logic now causes all data that would normally be passed from the data regeneration and gating circuits to the station teletypewriter to be looped back to the data set and transmitted to the line. Since much of the controller logic is included in the loopback path, many controller functions may be verified merely by monitoring the returned data at the test facility. For example, the ability of the controller to detect control characters used to blind and unblind the station receiver can easily be tested since the printer control logic will control the looped-back data. The obvious advantage of the automatic loopback feature is that maintenance personnel need not be dispatched to the station; nor do any switches have to be operated by the station attendant. By transmitting EOT the test facility returns the station to the idle state.

IV. CONTROLLER APPARATUS

The station controller consists of a wired nest equipped with a power supply and 13 printed circuit boards, and measures 15.5 inches long by 7.5 inches wide by 6 inches high (see Fig. 15). Different brackets are available to permit installation of the controller within various types of teletypewriters. Cords are used to interconnect the controller to the attendant unit and teletypewriter.

The data set is contained on one of the printed circuit boards; the other boards provide all of the controller logic. Discrete components
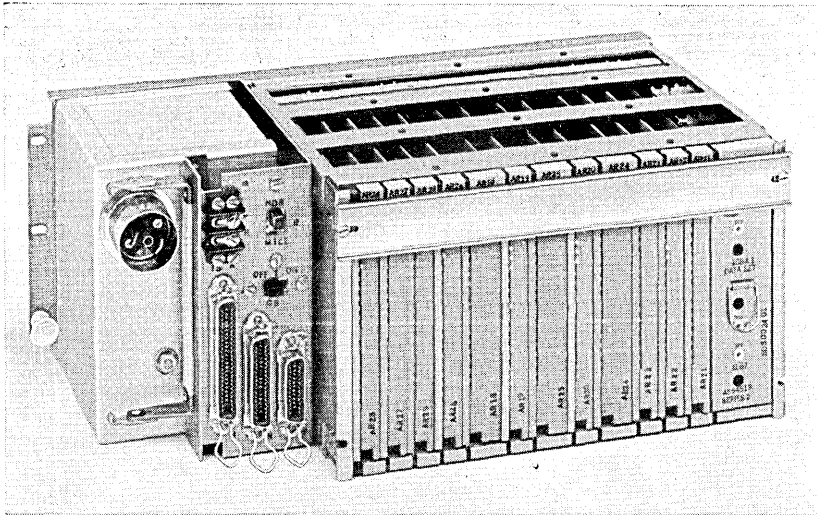
Fig. 15—Full-duplex station controller equipped with data set 108 type.

are used; each logic board contains, on the average, 13 capacitors, 12 diodes, 95 resistors and 26 switching transistors. The controllers may be equipped with either of two clock circuit boards, one for 100-words-per-minute operation, the other for 150-words-per-minute operation.

The power supply converts standard 117 volts, 60 Hz ac power to +24, +12 and −12 volts dc for use in the controller. The controller dissipates 50 watts of power making forced ventilation within the teletypewriter pedestal unnecessary in environments at normal room temperature.

The attendant unit for a half-duplex send-receive station is shown in Fig. 16. The loud speaker for the audible alarm is clearly visible. The attendant unit mounts under a cutout of the Model 33 and 35 teletypewriter cover as shown in Figs. 3 and 6. In the case of Model 37 teletypewriters, the attendant unit is installed in a vertical door panel as shown in Fig. 8.

V. CONCLUSION

The marriage of the electronic controller and the teletypewriter has been successful. Since system cutover on February 3, 1969, the stations have operated with a high degree of reliability.[7] This performance, coupled with the ability of ADF to frequently check the
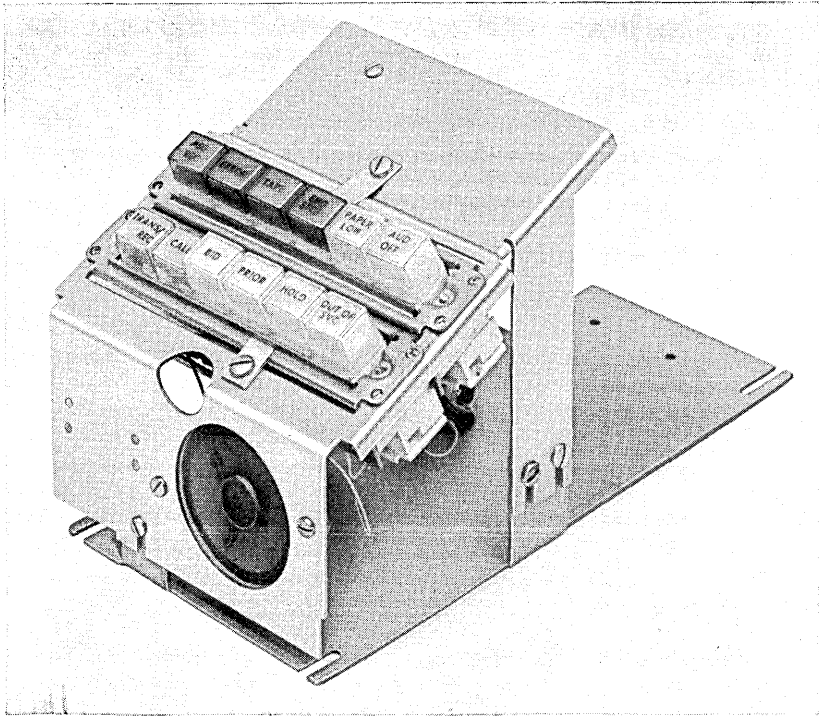
Fig. 16—Attendant unit for half-duplex send-receive station.

status of all stations and the inherent reliability of the private line telegraph network, has contributed significantly to the users' confidence in the No. 1 ESS ADF system.

VI. ACKNOWLEDGMENTS

REFERENCES

1. Bell System Data Communications Technical Reference, "150 Baud Private Line Channels, Interface Specification," February 1968, American Telephone and Telegraph Company, New York.

2. Davey, J. R., Hanley, F. H. and Purvis, M. R., "A New Telegraph Service-board Using Electronic Circuits," AIEE Trans. Commun. and Elec., *74*, Part I (March 1955), pp. 30–37.
3. USA Standard Code for Information Interchange, USAS X3.4-1967, United States of America Standards Institute, New York.
4. Kienzle, H. G., Weber, E. W., Zydney, H. M., Nicodemus, K. L., and Smith, M. T., Jr., "No. 1 ESS ADF: Message Processing Program Organization," B.S.T.J., this issue, pp. 2753–2829.
5. Chevalier, J. G., and Eisenhart, R. K., "No. 1 ESS Circuit Packs and Connectors," B.S.T.J., *43*, No. 5, Part 2 (September 1964), pp. 2441–2444.
6. Ferguson, J. G., Grutzner, W. E., Koehler, D. C., Skinner, R. S., Skubiak, M. T., and Wetherell, D. H., "No. 1 ESS Apparatus and Equipment," B.S.T.J., *43*, No. 5, Part 2 (September 1964), pp. 2355–2439.
7. Barney, D. R., Giloth, P. K., and Kienzle, B. E., "No. 1 ESS ADF: System Testing and Early Field Operation Experience," B.S.T.J., this issue, pp. 2975–3004.

No. 1 ESS ADF:

System Testing and Early Field Operation Experience

By D. R. BARNEY, P. K. GILOTH and H. G. KIENZLE

(Manuscript received August 14, 1970)

This article describes the equipment and test procedures used for system testing of a large store and forward system. It discusses a novel programmed computer load test facility for determining system operational characteristics in overload, system traffic handling capacity, and the adequacy of operational call register design. It presents early field experience with No. 1 ESS ADF controlling a very large nationwide data network for the Long Lines Department.

I. INTRODUCTION

A No. 1 Electronic Switching System Arranged with Data Features (No. 1 ESS ADF) was installed in New York City to operate and control the Long Lines Department's nationwide Administrative-Data Network (ADNet).[1] The new system was cut into service February 3, 1969, and connects some 720 Long Lines, Operating Company, and Western Electric Company locations to the No. 1 ESS ADF switching center through 400 circuits which are terminated by 1,250 four-row teletypewriters that use the American Standard Code for Information Interchange (ASCII). Connection is also made to computers at two Long Lines data processing centers so that computer-generated data can be distributed by the network, or field data can be assembled for computer center processing.

Large traffic handling capacity was needed for the ADNet. Therefore, a near maximum-sized system has been installed. Two autonomous data scanner-distributors terminate 1,024 full duplex or half duplex data lines which operate at speeds up to 150 words per minute. Call stores capable of handling 159,744 24-bit words, duplicated, are provided. Duplicated program stores of 327,680 44-bit words are pro-

vided for storing the 250,000 word program, along with translation and parameter data for 1,024 lines.

Two duplicated disk communities, each capable of storing 2,359,296 24-bit words, are used for in-transit storage. There are 12 magnetic tape units, so that journal filing, permanent filing, and retrieval can be handled with six tape units in the on-line pool for retrieval. This provides an on-line retrieval capacity of approximately 75,000 messages.

The various development testing phases of a message switching system of this complexity added up to a sizable undertaking. New testing techniques were developed to yield thorough hardware and software testing in minimum time.

II. FUNCTION TESTING

When testing any large system, it has been conventional to divide the work into two main phases, discrete function testing, and system operation and maintenance testing. However, it is quite difficult to define exactly where function testing for either hardware or software ceases and system testing starts. This article uses a somewhat arbitrarily-defined point: that point where all program words had been initially debugged (see Fig. 1) and all hardware testing had been completed in a fashion similar to a conventional No. 1 ESS system.[2] Hardware testing is not described here. Function and system tests are separated to present certain of the testing phases which were somewhat uniquely handled in the overall testing program.

The complete computer program required by the No. 1 ESS ADF system contains about 250,000 words. A considerable amount of program testing was required over a period of three years before the system was considered to be operating satisfactorily. The bulk of the testing was performed in the test model system laboratory at Bell Telephone Laboratories, Naperville, Illinois. The function testing required 22 months while system tests started seven months before system cutover and continued for eight months after. The progress in program testing, indicated by the amount of program words debugged during this period, is shown in Fig. 1.

During function testing the individual programs were tested as independently of other system functions as possible. A test plan was devised that would achieve the maximum (i) rate of program integration, (ii) number of programmers able to use the system effectively, and (iii) number of programmers on day shift. To implement this
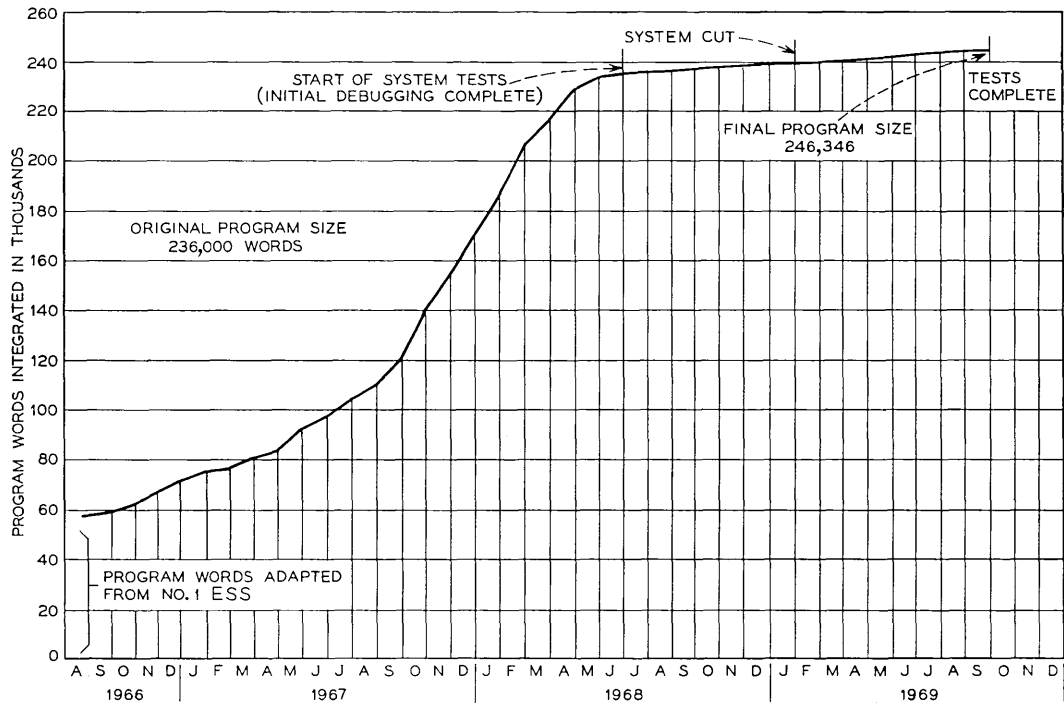
Fig. 1—No. 1 ESS ADF program integration.

plan, special utility programs were developed, new laboratory operating procedures were established, and a new program test console was designed. The console, as shown in Fig. 2, contained some 6,000 lamps indicating status in all parts of the system and the information stored in the many system registers. It was an expansion of previous ESS test console designs with their capabilities, but several important new features were added.[2] Facilities were installed for controlling and monitoring the frames unique to No. 1 ESS ADF, along with additional program and call store address matchers which could be set electronically under program control from card input data. The entire console was duplicated, but so interconnected that a duplex system could be controlled from either half, or the system could be split into two simplex systems each independently controlled by one half of the console and each half capable of independent operation. This feature doubled the machine testing capacity for those programs not requiring a duplex system.

The general concept for the plan of operation was to run the system laboratory as a computation center where programmers would work on a single day shift while there would be sufficient console operators to complete all the day's batch work in either two or three shifts, as required. A programmer could request to be present while his work was being run, to observe the system operation and make limited changes in his test procedure. All work, however, was under control of the console operator. No time was allowed for problem study at the machine, on machine time. All problem study was done off-line with only completely defined tests being run using machine time.

If a programmer were not present, his work was called a "batch" run. If he were present, his work was called a "personal" run. No time limit was placed on the length of batch runs, but the longer runs were usually performed during night shifts. However, a personal run was limited to ten minutes because experience showed that few programmers could efficiently use more than ten minutes of time on a personal run without requiring at least one-half hour off-line time to analyze his results.

This general plan was adhered to during the period of function testing and well into the system test period. During the end of the system test period before cutover, the request for batch runs dropped off and the personal run time was allowed to become longer. After the New York system cutover the work became almost entirely personal runs with the run time limit extended to 30 minutes.
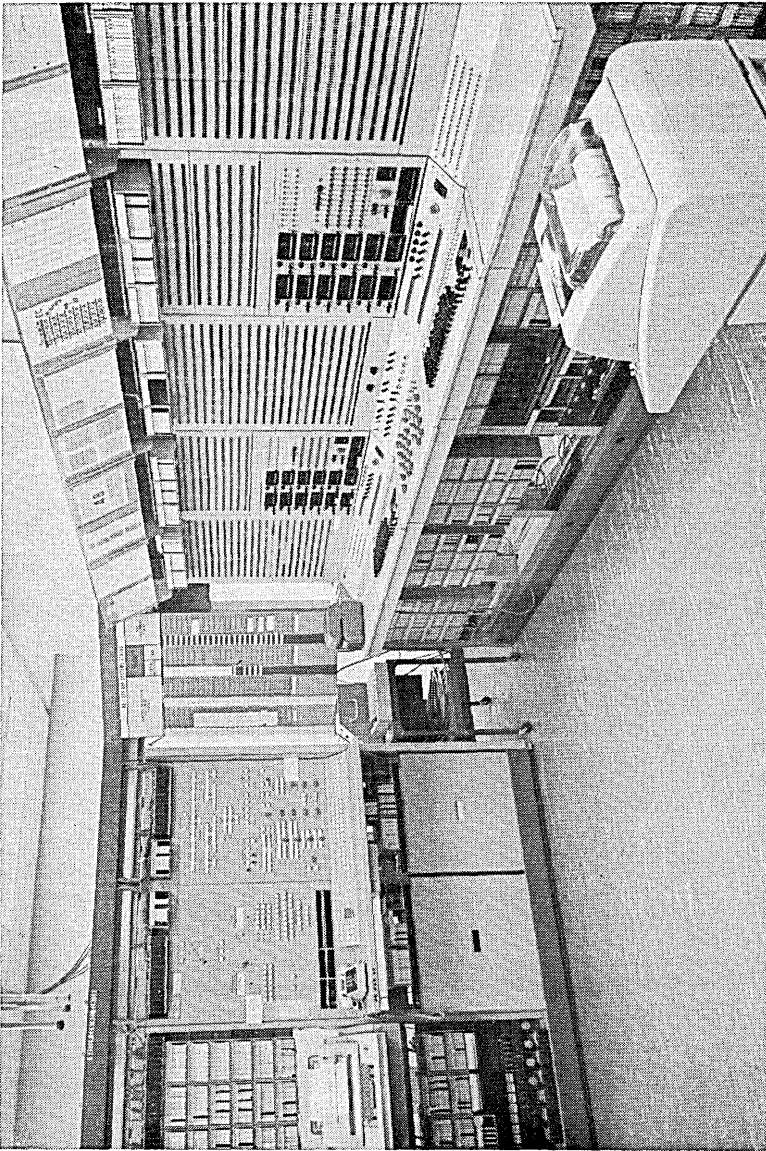
Fig. 2—No. 1 ESS ADF test control area.
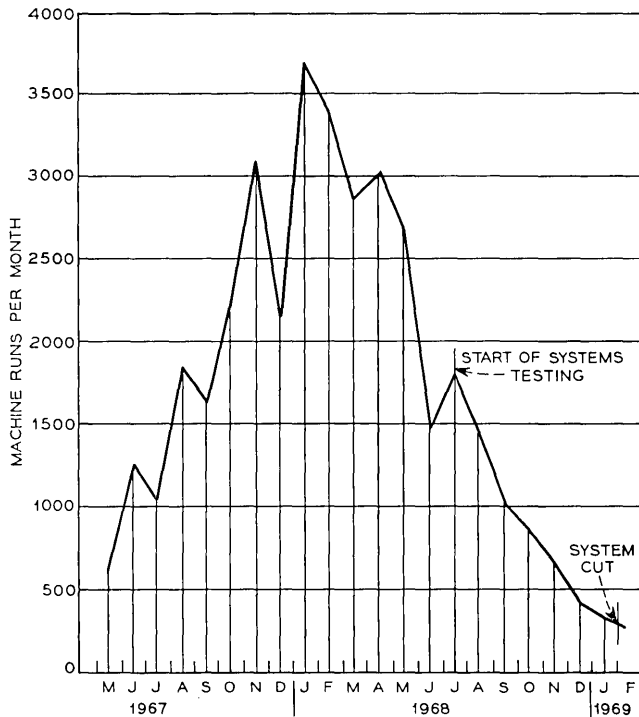
Fig. 3—Total batch and personal test runs per month.

At the start of program testing on a test model at Naperville, Illinois, not all the programs had been loaded. As more programs were loaded into the system, the quantity of test runs increased sharply. Figure 3 shows the total runs made per month from the start of integration until the New York system was cut over in February 1969.

Figure 4 shows the monthly percentage of the total runs which were batch runs during the entire test period. During the debugging of individual programs the percentage of batch runs was high, but as the testing became more of the system type, the percentage dropped off.

Figure 5 shows the average length of a test problem. The length of each test during the peak program debugging months was kept under five minutes. Of this, set-up and read-in of the instruction deck took less than one minute, the run itself averaged two minutes, and the high-speed printer dump used the remaining time. With the dual input console arrangement, the tests were made alternately on each half
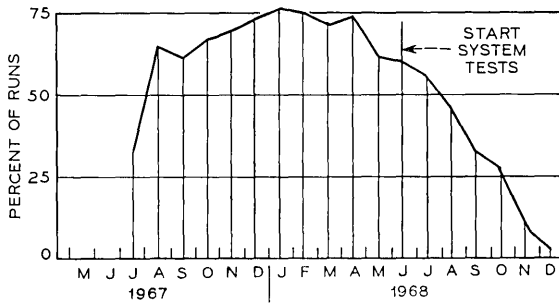
Fig. 4—Percent of batch runs to total runs.

with one half setting up the next run while the other half was making
a run. This feature greatly helped to keep running time to a minimum.

Figure 6 shows the monthly rate of program integration which
reaches a peak of over 18,000 words in a month. Figure 7 shows the
relationship between words integrated and tests made. The relation-
ship remained relatively constant throughout most of the integration
period, indicating that a definite number of test runs must be made by
a programmer to accomplish a given amount of program integration.
This seems to be independent of the length of run or the work load
being performed in the laboratory.

The relationship between the number of program words integrated
and hours of machine time is shown in Fig. 8. The overall average was
37 words integrated per hour of machine time based on the hours when
the machine was used only for program debugging. If all the hours of
machine time, that is, overwrite time and maintenance time, are in-
cluded, the average number of program words integrated per hour
drops to 30.
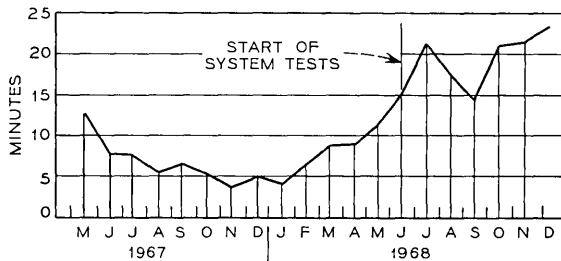
Fig. 5—Average length of run.
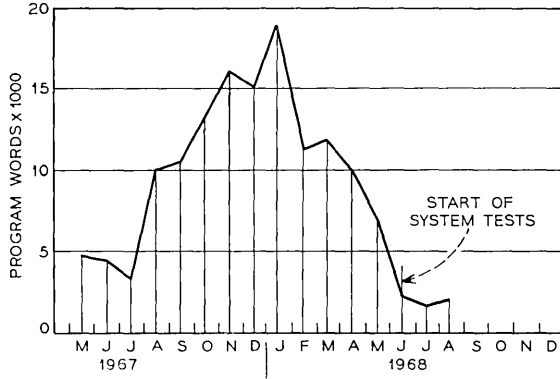
Fig. 6—Program words integrated per month.
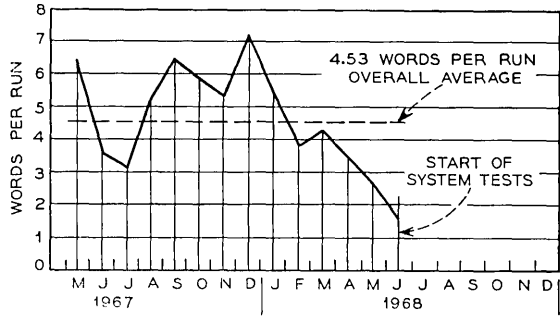
Fig. 7—Program words integrated per run.
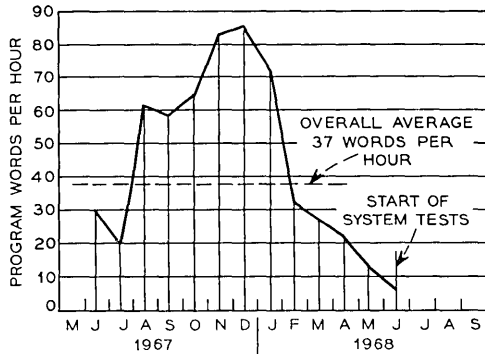
Fig. 8—Program words integrated per hour of machine time.

During the various phases of program testing at the Naperville Laboratory, 58,000 overwrite words of program were introduced while debugging 134,000 words of program. During the seven months of system testing, 22,000 overwrite words were made against the whole program of 242,000 words, that is, 9.4 percent of the program was changed. During the first eight-month period after system cutover, 9,700 overwrite words have been put into the system, many because of feature additions. Detailed information on the number of overwrites made is shown on Figs. 9 and 10.

III. SYSTEM TESTING

System testing was conducted primarily on the New York installation of the No. 1 ESS ADF system. This installation alone contained a full complement of hardware necessary to operate the system at full design capacity and to provide a nearly normal environment for testing operating procedures. This testing period started seven months
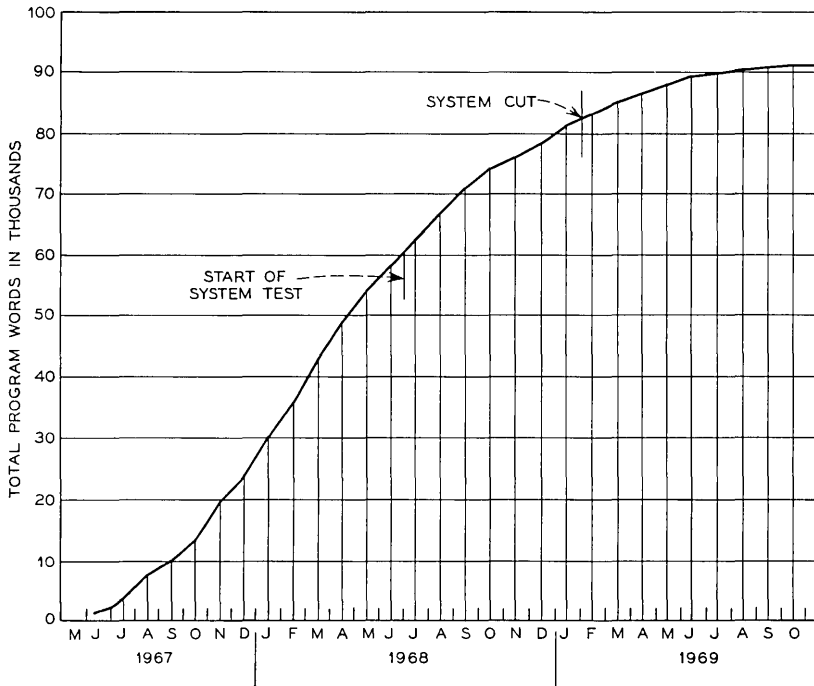
Fig. 9—Total overwrites installed in Indian Hill system during testing.

before cutover and continued for eight months after. The system tests consisted of three distinct but related areas of testing: feature tests, network tests, and load tests.

3.1 *Feature Testing*

A detailed test plan was prepared before the start of system testing. This plan listed all the tests to be conducted and the system conditions and configuration for each test. The tests included all operational and administrative features, maintenance test procedures, and system responses to abnormal conditions.

A test bed of stations, not part of the Long Lines network, were connected to the New York system for operational feature testing. This group of stations consisted of six eight-level ASCII stations and nine five-level Baudot stations. There were approximately 500 operational tests subdivided into major feature categories as shown in Table I. Each test required the preparation of station message tapes; and because of the limited number of stations in the test bed, recent change tapes had to be prepared for many of the tests to establish the proper station options for the given test. Several general purpose programs were written to aid in the preparation of these test tapes.

These tests were systematically run during the early months of the system test and were very effective in detecting program bugs. Many of the tests were performed while a large system load was being placed on the system by the load facilities which are discussed in Section 3.3. A subset of these tests were used until cutover to test the introduction of new program loads in the New York system.

3.2 *Network Testing*

During the seven months before the cutover the 400 circuits and approximately 1250 teletypewriters were installed and tested from manual
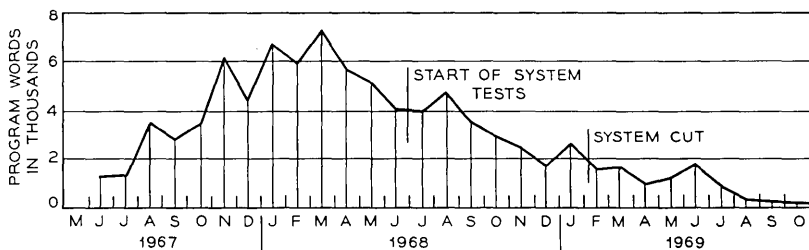
Fig. 10—Overwrite words installed per month.

TABLE I—OPERATIONAL FEATURE TESTS

Heading format	133
Addressing	8
Precedence	30
Privacy	4
Multiline hunt group	15
Message retrieval	48
Poll-delivering criteria	111
Undeliverable traffic	17
Miscellaneous customer services	126
Total	492

transmission test positions. Five months prior to cutover, the translation data for the total Long Lines network was installed in the New York system. During this period, one shift of system time was devoted to network testing which checked the ability of each station to transmit and receive traffic, and which verified the translation data. Message tapes were prepared by Bell Laboratories and mailed to station attendants for testing the transmission capability of each station. In addition, message tapes were prepared for a Long Lines location to send to each station for testing that station's receiving capability. Since stations continued to be installed throughout this period, special network testing programs and procedures were developed to allow the translation data to be progressively activated as the stations were installed.

These tests proved to be very effective, and the smooth cutover of the network was a direct result of the joint efforts and cooperation of Long Lines and Bell Laboratories personnel during this test period. The network tests also provided a vehicle for testing the No. 1 ESS ADF switching center. Any troubles reported during the tests were given the bookkeeping name of failure reports, and corrections were introduced into the program. Figure 11 shows the number of failure reports issued during system testing. Prior to cutover, there were 690 reports per month. While the network tests were responsible for a large number of these failure reports, load tests and feature tests which were conducted concurrently and the continued testing on the Naperville test model contributed significantly to these results. After cutover the rate dropped to an average of 135 reports per month for the next five months and then dropped to near zero. These failure reports were divided into three classes:

Class A, serious service-affecting troubles which were corrected immediately (less than five percent).
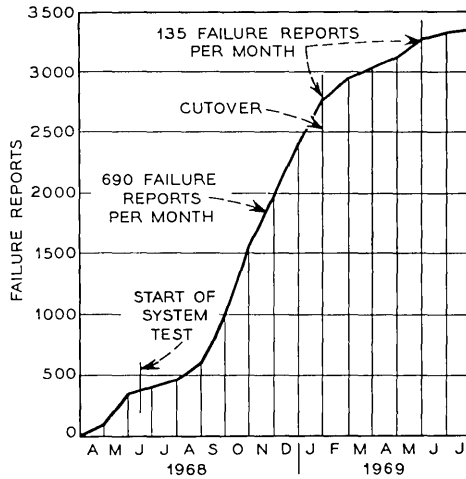
Fig. 11—Total number of failure reports.

Class B, refinements to service or maintenance which were corrected more routinely (over 80 percent).

Class C, corrected only if simple, otherwise deferred for later program versions.

3.3 Load Testing

A switching system cannot be considered to be fully tested until it is stressed to the limits of its traffic handling capability. However, the large traffic capacity of the No. 1 ESS ADF switching center makes live load testing difficult because of the problem of generating the large amounts of controlled traffic required to stress the system to capacity. To fully load the system with traffic on each line is impractical because the total cost of hardware becomes prohibitively great. Even if one could afford the hardware, it is virtually impossible to administer the generation of traffic to produce a controlled load which is reproducible. Therefore, load testing of the No. 1 ESS ADF system was accomplished through the use of two load boxes which were developed for this purpose. The test configuration for the load boxes, a five-level load box and a computer-controlled load box, is shown in Fig. 12.

3.3.1 Five-Level Load Box

The five-level load box was designed as a means of testing the data scanner distributor units under full load. This device consisted of ten

paper tape readers each of which could be connected to as many as 52 ports on the data scanner distributor units. The load box thus simulated 520 five-level full duplex stations. The five-level full duplex signalling sequence is such that after the initial handshaking with the station to establish the connection, a multimessage transmission can be continued indefinitely since no further signaling dialogue between the station and the switching center is required. The control teletypewriter shown in Fig. 12 was used to handle the initial signaling sequences for each of the paper tape readers. Once the connection was established, the control teletypewriter was then available for use with another tape reader. While this load box arrangement was designed primarily for testing the data scanner distributor units, it had some characteristics which produced useful program tests. Since a paper tape reader was connected to 52 lines, the traffic presented to these lines was not random, that is, all actions such as start and end of message occurred simultaneously. This in turn produced stresses on particular program functions as each action occurred. This load box was an effective testing device in the early stages of system testing, but had to be removed when the network tests started to make way for the Long Lines network.
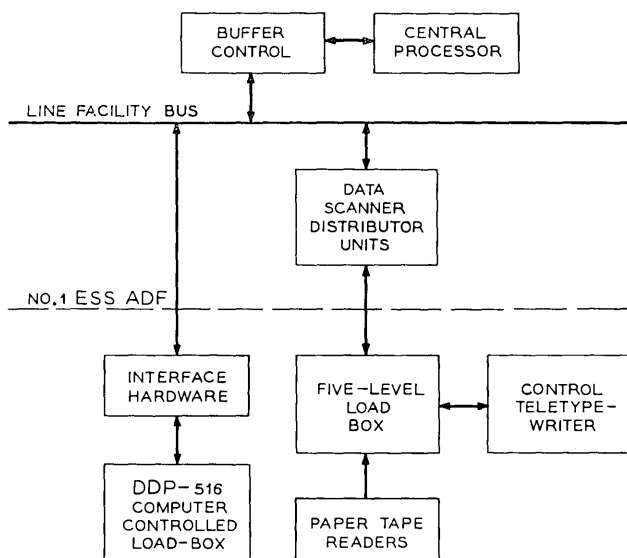
Fig. 12—Load box facilities used in testing the No. 1 ESS ADF system.

3.3.2 *Computer-Controlled Load Box*

A computer-controlled load test facility was developed for the No. 1 ESS ADF project. This load box consisted of a small computer, the Honeywell DDP-516, and hardware connecting it to the ADF system in such a way as to simulate a data scanner distributor unit with 512 lines and two stations per line. The program written for the DDP-516 computer provides a large degree of flexibility in controlling the traffic parameters used to specify the type of traffic load presented to the system.

These parameters are: (*i*) the number of active originating stations, (*ii*) the level of presented load called message presentation rate, (*iii*) the multiple address factor or number of addresses per message, and (*iv*) the text length of the message. The latter three parameters can be provided with constant, normal, or exponential distribution. Traffic statistics collected by the load box are periodically printed and have proven to be valuable in evaluating the No. 1 ESS ADF system operation.

The DDP-516 computer was so programmed that steady state or dynamic load testing could be performed. In a steady state test, the traffic parameters remain fixed during the test. This allows the system response to a steady load to be observed. In a dynamic test, the basic traffic parameters for the load box are varied with time according to a pattern. This allows the system response to varying traffic conditions to be observed.

The first use of the load box was to supply a background load while feature tests were performed on the system. Subjecting the No. 1 ESS ADF to traffic loads while making discrete functional tests of individual programs exposed program faults which otherwise would have remained undetected. Many of the program faults consisted of subtle interactions within the program which had a very low probability of occurring under light load. Such situations could be created by large volumns of traffic which would never have been produced manually. The ability to control the traffic and reproduce the tests made it possible to examine a given fault repeatedly.

3.3.3 *System Capacity Tests*

To build a foundation for future traffic engineering of the Long Lines network, it was necessary to make dynamic system load tests to gather information concerning the capacity limits of the system. In addition, system performance under heavy loads and overloads needed to be demonstrated. Since the system is now only partially

loaded from the Long Lines network, the use of the computer-controlled load box was the only means of obtaining the desired test results.

The large number of system problems encountered before cutover and in the early months after cutover, and indicated in Fig. 11, made it impossible to perform meaningful capacity tests. However, these problems were gradually fixed and by September 1969, all the known software and hardware problems had been eliminated. A week-long test of system capacity and overload performance was conducted during September 1969.

The No. 1 ESS ADF system uses the time required by the program to complete the main program cycle (called the E-to-E cycle time) as a measure of the load on the system and as a means of initiating load control procedures. To aid in collecting data on real-time capacity tests of the system, additional measuring equipment was provided with the load box. This consisted of an electronic counter which was triggered upon the completion of each main program cycle of the No. 1 ESS ADF processor. The counter was started on odd pulses and stopped on even pulses. The results of the counter were read by the load box, thus the load box collected measurements of every other main program cycle time. These data were printed periodically as a part of the load box traffic statistics with the maximum, the minimum, and the average main program cycle time for the preceding period.

The load box also predicted the capacity based on measurements taken in the preceding measurement period. The prediction was an estimate of 90 percent of the limiting capacity, that is, 90 percent of the capacity at which the E-to-E cycle time would be infinite. The 90 percent level was chosen as the best guess of the level of capacity which would be achieved with the current load control parameters. The prediction was based on this equation which was derived from theoretical analysis of the system capacity.

$$C(0.9) = \frac{0.9 E_A T}{E_A - E_N} \tag{1}$$

where:

$C(0.9)$ = 90 percent of the limiting capacity in characters per second.

E_A = Average E-to-E cycle time in ms.

E_N = No load E-to-E cycle time in ms.

T = Average capacity in characters per second during the measuring interval.

For some values of traffic characteristics, call store size and not real time, limits the system capacity; thus a prediction method was necessary to determine real-time capacity. Also, the load box prediction was useful during load box runs to establish changes in the input load box parameters to establish the desired load level. The data required for the calculation of equation (1) was measured by the load box during each test interval with the exception of E_N, the no load E-to-E cycle time. This quantity which was an input parameter to the load box was a constant for the purpose of the prediction calculation. The value used was the average E-to-E cycle time observed during long periods of operation with no load on the system and was about 50 ms. The variation in E_N during the test runs results in a prediction error in the calculation of equation (1). The value of the error decreases as the load increases and has been observed to be less than 3 percent for values of the capacity in excess of 50 percent of the real-time capacity.

The real-time capacity of the ADF system is a function of multiple address factor, message length, and statistical distribution of message length. Past estimates of capacity based on calculations rather than on measurements have indicated that the multiple address factor (M), the number of addresses per message, has a minor effect on capacity. To validate this, tests were made with $M = 2$ and $M = 3$ for the same average message length. The results indicated only a 5 percent increase in capacity for the higher multiple address factor. This confirmed the original assumption so that the major effort was directed at determining real-time capacity as a function of message length and its statistical distribution, and all other tests were made with a multiple address factor of 2.

Two test runs, fixed message length and exponential distribution of message length, were made for four average message lengths: 100, 500, 900, and 1,300 characters. The call store rather than real time was found to be the limiting factor on capacity for all average message lengths greater than about 300 characters. The call stores could support more capacity if the data rate on the lines were increased. For this reason, the load box was programmed to produce a line rate of 150 words per minute rather than the normal 100 words per minute for the tests made at the three longer message lengths. A load was applied to the system and allowed to stabilize. Recordings of capacity, predicted capacity, and minimum, maximum, and average E-to-E cycle were taken. The load was increased and the process repeated
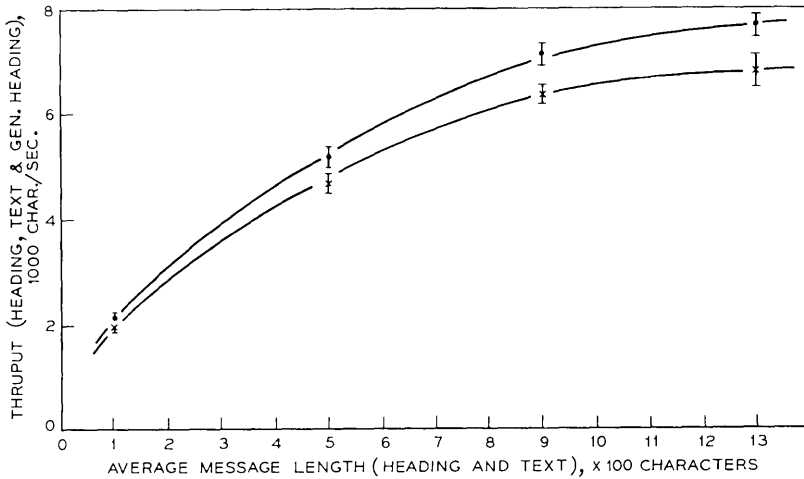
Fig. 13—Real time average and limits of capacity measurements for (●) fixed message length and for (×) exponential distribution of message length. The multiple address factor is 2.0.

until the real-time load control level was reached or the call store limited further increased in the load.

The predicted capacity, 90 percent of the limiting capacity, as a function of the average message length for the two types of message distributions, is given in Fig. 13. The prediction error should be low since capacity levels of 75 percent or higher were reached during each test run.

The tests were analyzed to determine whether the ADF system could operate at 90 percent capacity and, even further, whether it should. To understand this analysis, it is necessary to consider the functional relationship between the average E-to-E cycle time and the capacity, and how the capacity is limited by the load control parameters. The functional relationship between the average E-to-E cycle time and the capacity is a family of curves dependent on message length which tends to make the present considerations quite complex. To simplify the analysis, only the functional relationship between the average E-to-E cycle time and normalized load were considered, where normalized load is defined as the ratio of the amount of data that pass through the machine to the limiting capacity. This results in a single curve independent of message length. This relationship is given in Fig. 14 based on the load box data. At the load con-
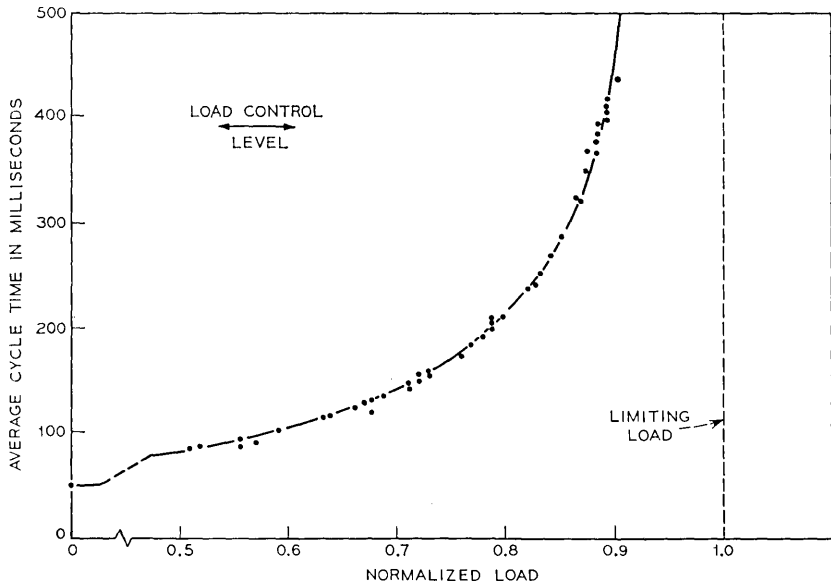
Fig. 14—Average E-to-E cycle time versus normalized load (ratio of data passing through the computer to its limiting capacity).

trol level, currently set in parameters as 400 ms, the capacity is kept by the load control procedures to approximately 88 percent of the limiting capacity, not quite 90 percent.

To determine if the load control level is set properly, the effects of peak rather than average E-to-E cycle time must be considered as two other actions are initiated as a result of parameter value comparisons with E-to-E cycle time. (i) If the E-to-E cycle time exceeds 3.24 seconds, a real-time overload will be declared and certain operational tasks will be suspended. (ii) If the E-to-E cycle time exceeds 4.32 seconds, phases of emergency action will be initiated.

During tests with the capacity at 85 to 90 percent, peak values of E-to-E cycle time went very high—about eight times the average E-to-E cycle time. However, the number of occurrences of real-time overload was small and it appeared that the recovery was rapid, that is, the real-time overload ended in the E-to-E cycle following the one in which it began. The overload appeared to have no effect on service; thus the current value of 400 ms for the load control level appears to be quite reasonable. With this control level, the system will operate at about 88 percent of the limiting capacity. The capacity curve of

Fig. 13 for the exponential distribution of message length appears to be a conservative representation of the real-time capacity of the ADF system for the Long Lines traffic characteristics.

To further confirm this conclusion, a load test of the No. 1 ESS ADF system was made during a normal traffic day. Throughout the test period, 8:00 a.m. to 6:00 p.m., the traffic data produced by the system was monitored. The load box parameters were adjusted to maintain a system traffic intensity that was three times that produced by the Long Lines network alone while maintaining the traffic profile, that is, the length of busy hours was unchanged. During the course of this demonstration, all the traffic was handled without an overload, although some of the call store facilities were in very heavy use. Toward the middle of the afternoon, it was decided that an overload should be forced on the system. Therefore, the load box traffic was sharply increased to force a call store overload. Load control was automatically called in and, after the overload was removed, the system recovered from the overload without incident or loss of messages. This test was a demonstration to establish performance credibility and was not used to collect capacity data.

3.3.4 *Queueing Tests*

A series of tests was designed to show that the queueing logic, imbedded throughout the operational program, operated properly for each and every call store facility. A method was devised, using a small number of program store overwrites, to artificially reduce the number of items assigned to any given call store facility. For the call store facility under test, the number of items available could be controlled and changed by setting the appropriate value in a call store location. With the load box providing a high level of load on the system, but with the load controlled to a level that produced no facility queueing, the number of items for the facility under test was reduced to where virtually continuous queueing existed. The system was operated for approximately one hour in this state to exercise as many points as possible in the program which involved queueing for the facility being tested. Audits were run periodically to detect errors, and the performance of the system was monitored. It should be obvious that there is no assurance that all points in the program which involve queueing will be tested by this process; however, those with a reasonable probability of occurrence will have been thoroughly exercised. All call store facilities were tested through this process.

In addition to the call store facility queueing tests, a test of the performance of the system during a message store overload was made. Load control procedures were initiated which allowed current inputs to continue until they were complete, but restricted the acceptance of any further input traffic until the output process could make storage space available.

3.3.5 *Call Store Tests*

Capacity of a store and forward system is generally defined in terms of the system's thruput capability, that is, the number of characters transmitted and received by the switching machine per unit of time. In line-switched electronic switching systems, capacity is primarily limited by the real-time capability of the central processor. However, in store and forward systems, the requirements for storage are much greater. The system is committed to accepting originating traffic without regard to whether the traffic can be delivered to the terminators. Large amounts of storage are required to hold or queue the traffic for delivery to the terminators in addition to the storage required for the bookkeeping associated with transmission to and from the switching machine. In the No. 1 ESS ADF system, the call store capacity, as well as the real time of the central processor, limit the system capacity under certain traffic characteristics. Storage space is assigned by two basic considerations: that which is associated with transmission—the greater the storage, the higher the thruput that can be obtained—and that which is associated with holding traffic in the system—the greater the storage, the longer the traffic may be held in the system—which, in turn, means that transmission facilities can be used more efficiently. Both of these functions are vying for the same pool of finite storage.

The current call store allocation for the New York system was based on a traffic survey made by the Long Lines Traffic Department. The recommended assignment of call store for all traffic-dependent areas based on this design load is shown in Table II. The five major items—input processing registers, output processing register, assembly-disassembly blocks, message processing blocks, and message queue registers—require 88 percent of the assignable area with message processing blocks requiring more storage than all other items. For this reason, the major traffic engineering effort was placed on these five items in the development of assignment rules.

To test the adequacy of the call store assignments, several load box

TABLE II—PRESENT CALL STORE ASSIGNMENT FOR TRAFFIC DEPEND-
ENT FACILITIES

Facility	Words per Register	Number Assigned	Total Number of Words	Percent
Assembly disassembly blocks	32	544	17,408	16.2
Input processing registers	16	150	2,400	2.2
Output processing registers	19	331	6,289	5.8
Message processing blocks	32	1645	52,640	48.7
Message queue registers	3	5500	16,500	15.2
Other items			2,724	2.5
Unassigned area			10,038	9.4
Total			107,999	100

runs were made varying the traffic intensity of the presented load.
During these runs, the quarter-hour traffic printouts provided by the
ADF system were used for capacity measurements and for average
usage of each call store facility. Peak values for the call store usage
counts were obtained by reading the traffic counters inside the ADF
system about once per minute.

The peak and average usage of each facility as a function of traffic
load was analyzed. As a result, the present procedures for engineering
call store appear to be quite valid, with some minor changes. Making
use of these changes in the traffic engineering procedures, Fig. 15 il-
lustrates an approximate relationship between the number of call store
words required per input line and the average input load per line. The
number of call store words include the requirements for input trans-
mission (input processing registers, assembly-dissassembly blocks, and
message processing blocks), output transmission (output processing
registers, assembly-disassembly blocks, and message processing blocks),
and holding traffic on the message queue (message queue registers).
This illustrates the design choice between call store facilities for input
and output transmission and facilities for delaying traffic. As the input
load increases, a breaking point is reached where the load on each
output line is one erlang. At this point there is a choice of adding
more output lines and associated call store output transmission facili-
ties or adding only message queue registers to hold the traffic for fu-
ture delivery. The latter is less expensive if one can put up with the
increased service delays.

Some knowledge of the future traffic requirements is needed to make
recommendations concerning future traffic engineering and call store

assignment of the ADF system. Call store must be put to the most effective use to reduce transmission plant cost while meeting the traffic needs. The maximum number of call stores were provided with the New York system. It is, therefore, assumed that the network load will increase as a result of the addition of stations to the present plant in preference to the addition of large quantities of transmission facilities. In the process, the load on individual lines should be balanced.

Because future traffic requirements are unknown, Table III proposes an allocation of call store in which a traffic load of about 3.5 times the present Long Lines load can be handled. If the Long Lines traffic forecast indicates that capacity levels are needed beyond that which can be achieved with the call store allocation of Table II, additional transmission plant cost will be incurred. By converting all the 100 words per minute lines to 150 words per minute, the amount of data
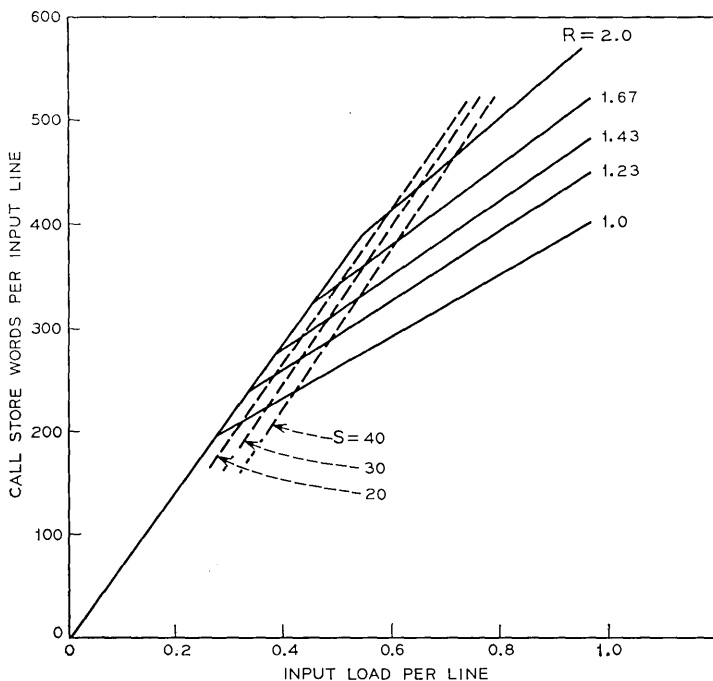
Fig. 15—Approximate call store requirements per input line as a function of load per input line for the major traffic dependent registers. S = Service delay in minutes. R = Ratio of output to input lines. Output message length: 1200 characters. "Busy hour": 1 hour followed by a low traffic period for emptying message queues. Multiple address factor: 3.0.

TABLE III—PROPOSED CALL STORE ALLOCATION FOR TRAFFIC DEPEND-
ENT FACILITIES

Facilities	Words per Register	Number Assigned	Total Words	Percent
Assembly-disassembly blocks	32	711	22,752	21.1
Input processing registers	16	182	2,912	2.7
Output processing registers	19	415	7,885	7.3
Message processing blocks	32	1,333	42,656	39.5
Message queue registers	3	10,006	30,018	27.8
Other items			1,776	1.6
			107,999	100

passing through the computer would approach the system's real-time capacity, and a traffic load of about five times the present Long Lines load could be handled. The system capacity limits are summarized in Fig. 16.

IV. FIELD EXPERIENCE

By May of 1970, the No. 1 ESS ADF switching system at 811 Tenth Avenue, New York, had been in continuous operation for 15 months. During that time, hardware and software changes were introduced to
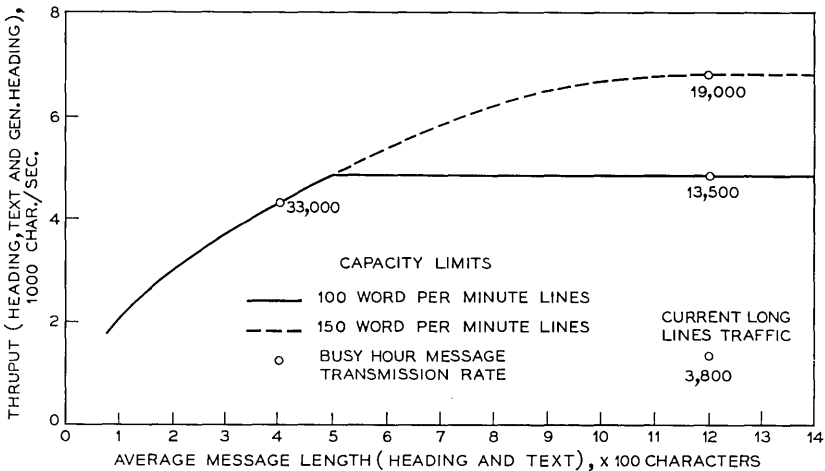
Fig. 16—No. 1 ESS ADF capacity limits: —— 100 wpm lines; – – – 150 wpm lines; ○ busy hour message transmission rate.

correct troubles encountered, new service features were added, and a final generic program was installed without interrupting service.

Although certain operational difficulties were encountered during the early months, the system has performed satisfactorily. The network is now being used to handle all Long Lines administrative messages, commercial service orders, traffic service orders, service results, payroll, plant circuit orders and expense analysis reports. As shown in Fig. 17, over 20,000 messages (originated plus terminated) were handled daily in the first month of operation. As new projects were added to the network, the daily traffic grew to 35,000 messages averaging 1,200 characters each. Additional traffic will be added during the next year to increase the load to 120,000 messages daily.

4.1 *Service Experience*

4.1.1 *Grade of Service*

The quality of service of a store-and-forward system may be measured by the: (*i*) delivery time of messages, (*ii*) number of lost or misdirected messages, and (*iii*) number of station troubles per day. Because of the nature of store-and-forward service, no one criterion tells the whole story.

The pick up and delivery time of messages depends on how heavily a multistation line is loaded and, therefore, is a function of engineering rather than switching center service. On lightly loaded lines, the
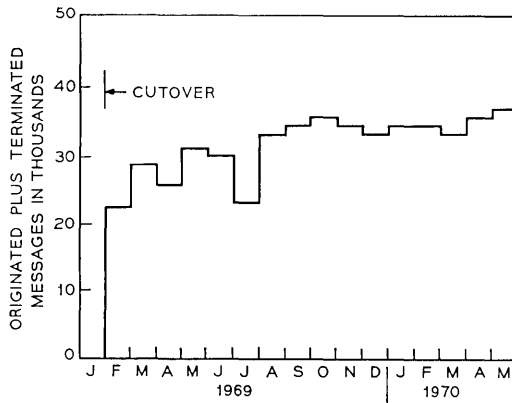
Fig. 17—Administrative-data network average daily traffic, with an average message length of 1200 characters.

pick up is immediate and the delivery to any station on the administrative-data network is less than one minute. On heavily loaded lines, pick up delays may be as long as 20 minutes and delivery may be delayed varying amounts depending on the length of the queue for the terminating station. Customer reaction to service has been excellent, but some additional lines have been added in some locations to reduce overload on key message center lines.

During the first week of operation, some messages were lost during a duplex disk failure. Since that time over 10,000,000 messsages have been handled without loss or misdirection of messages accepted by the system. About one message in 5,000 is undeliverable and the originator is asked to send the message again.

The number of station and transmission troubles is about one-fifth the monthly Bell System average for conventional teletypewriters in the field. This improved performance can be attributed to the teletypewriter electronic controller, new data sets, and continuous monitoring of the network by the No. 1 ESS ADF switching center.

4.1.2 *Message Retrieval*

The number of retrieval requests have been averaging about 250 per day or about 0.7 percent of daily traffic. These retrievals are needed by the user because of station problems, loss by local attendant, delivery to an additional terminator, or desire for a second copy. The breakdown for each of the above categories is not known in detail, but most retrievals result from station problems. The amount of retrieval traffic is below the level predicted and the retrieval service given has proven satisfactory to the user.

4.1.3 *Network Management*

The network management center described in Ref. 3 has proven to be very effective in managing a network as large as the Administrative-Data Network. On many occasions when traffic bottlenecks occurred because of transmission or terminal outages, network management personnel were able to pinpoint these bottlenecks and reroute the traffic until the line or station troubles were cleared. In addition to the specific network management features, the memory capacity of the system also is helpful in handling large backlogs of traffic. For example, the system provides for an interconnection with off-line commercial computers which process payroll data and other projects. Because these commercial computers are occasionally out of order, messages

destined for the data processing center cannot be delivered and traffic
in excess of 1,000 messages has been stored and queued for delivery in
No. 1 ESS ADF for a day or more.

4.1.4 *Network Maintenance*

The normal maintenance procedure for data transmission and sta-
tion plant is to test and repair facilities in the light of customer
trouble tickets. To identify trouble before a customer reports it, No. 1
ESS ADF has been designed to locate trouble within seconds and re-
port the trouble automatically to the maintenance center. This is pos-
sible because the switching system is programmed to poll every sta-
tion not originating or terminating traffic every two seconds. By
analyzing the results of polling and other "handshaking" signals, it is
possible to determine station problems such as power failure, teletype-
writer out of paper, jammed paper, or general station failure. Open,
shorted, or noisy lines are also detected and reported to the mainte-
nance center.

These automatic trouble reports now supplement customer trouble
reports and, in many cases, malfunctions are being fixed before the
customer is aware of a problem. The average time to repair admin-
istrative-data network data line and station troubles is below the Bell
System average for similar equipment.

4.1.5 *Addition of New Features*

As a result of early operational experience, several new features
seemed desirable to improve system performance. For example, im-
provements were made to identify the specific error in a heading
format of a rejected message so that the originator could more easily
make corrections. Provisions were also made for automatic control of
a tape punch or other auxiliary equipment at a station by using ad-
dress mnemonics. Features also were added to improve the procedure
for introducing translation changes for new lines and stations.

The new or improved features were introduced temporarily with-
out service interruption and were all included in the final generic
program.

4.2 *Operational Problems*

4.2.1 *System Down Time*

The No. 1 ESS ADF switching center and the network it controls
are designed for continuous operation without loss of messages or de-

lays in message delivery. As described in Refs. 3 and 4, the programs were designed so that auditing programs could monitor and correct any call store words which were mutilated by software or hardware malfunctions. In addition, special programs called "emergency action programs" were provided to take care of excessively mutilated data which could not be handled by audits, or failures of system operation which could not be cleared by simple maintenance. These programs restart certain registers and memory locations and make it possible for the system to recover and continue processing.

Such emergency action programs are divided into four successive phases, each automatically called in if the situation is not corrected by the preceding one. The first three are relatively short and do not interfere with data processing. Phase 4 lasts about 40 seconds and does interrupt processing. However, this does not significantly affect service since originations are queued for input and output and the delay is not noticeable to the customer. Messages already accepted by the system are not mutilated, but messages being originated while a phase 4 program is in progress are aborted and the originator is automatically advised to send the message again.

These defensive tactics proved to be very effective in reducing system down time during the early months after cutover as they were able to maintain satisfactory service even with hardware and software troubles. Figure 18 shows the system down time since cutover. The relatively high down time shown during February and March resulted from program and hardware troubles not detected until after
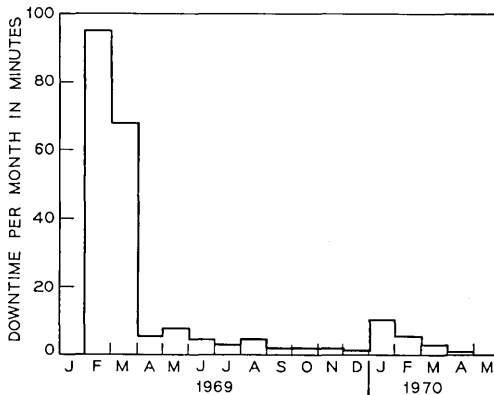
Fig. 18—System downtime in minutes.

a continuous live load had been placed on the system. These troubles were rapidly corrected. Total down time per month then dropped dramatically and has varied from 1 to 5 minutes per month. During January 1970, when the final generic program was introduced, several phase 4 emergency programs were required before the system would run satisfactorily on the new generic program. These emergencies accumulated 10 minutes of down time during the midnight hours.

4.2.2 *Types of System Problems*

Analysis of trouble reports show that software accounted for many of the early troubles. Program troubles were encountered in the tape retrieval system when it was under heavy retrieval load. Unfortunately, the load test facilities used before cutover were not capable of testing a heavy message retrieval load. Several other problems not sensitive to load, but to improper system operation by the user became evident. For example, one user tried to send a 905-multiple address message. The design limit is 379 addresses, with a check to reject messages asking for a greater number. However, the program was in error and did not detect the illegal request, so the machine switched automatically to emergency action programs.

Some maintenance programs were too sensitive. For example, thresholds for allowable error rates were set too low and, in some cases, the maintenance strategy of system reaction to malfunction had to be revised.

After the initial software troubles were corrected, the remaining troubles generally resulted from either hardware failures or improper procedures by maintenance personnel. Circuit pack failures were few and in line with previous data, as reported in Refs. 5 and 6.

Two analog units in the system, the tape transport and the disk file, required the most attention. The tape transport is sensitive to dirt, tape wear, and transport adjustments. Since 12 tape units are available, the problems were not critical to system operation, but they did require considerable maintenance. New alignment procedures, cleaner rooms, and close inspection of magnetic tape quality has reduced the troubles.

The disk file had failures in head diodes. An improved diode is now available, which is gradually being put into service. The disk file also had several "head crashes;" that is, the head touched the rotating disk, scratching off the magnetic coating. Such head-touching problems can most likely be attributed to dirt. New routines have been set

up for cleaning the sealed disk units periodically and replacing the 0.2 micron dirt filter more frequently. The long term solution appears to be the development of a closed self-purging system so that air is recirculated and dirt is not allowed to accumulate.

V. CONCLUSION

The development of No. 1 ESS ADF has resulted in the design of several new types of hardware and the creation of some new testing techniques. The introduction of this system puts into Bell System operation for the first time disk files, acoustic delay lines, magnetic tape retrieval systems, and an electronic autonomous scanner-distributor for data lines. Newly introduced are: batch program debugging on an ESS machine, dual position program test console with matchers that can be set electronically, program controlled computer load testing, real-time printout of monitor dump facilities, and a complete set of improved utility programs to facilitate program compiling, loading, debugging, and insertion of program changes.

The operation of the new system at 811 Tenth Avenue, New York, since cutover, has been very satisfactory. Although the system is not being loaded to handle maximum traffic now, a live load test was run to demonstrate very large traffic handling capability and the ability to operate in real-time overload without mutilating, aborting, or losing messages. The measured traffic handling capacity of this system is a function of message length. For example, the system can handle 19,000 messages per hour for 1,200 character messages or 33,000 messages per hour for 400 character messages.

The system has proven to be reliable and during the last 15 months has handled 10,000,000 messages without loss. The monthly system down time is averaging about two minutes per month with excellent prospects for further reduction as more experience is gained with the system.

The field experience gained since cutover confirms the effectiveness of the service features, the adequacy of system traffic capacity and the reliability of system operation.

VI. ACKNOWLEDGMENTS

ing reported on in this article. In particular, we would like to acknowl-
edge the contribution of R. T. Herbst who initially conceived the
new debugging techniques and encouraged the development of the com-
puter-controlled load box. In addition, we acknowledge the contribu-
tions of C. A. Mast and G. W. Smith, Jr., for the development of the
computer-controlled load box, of M. C. Lange for the development of
the five-level load box, and of M. O. Fichter for development of unique
program administration and debugging techniques.

REFERENCES

1. Ewin, J. C., and Giloth, P. K., "No. 1 ESS ADF: System Organization and
    Objectives," B.S.T.J., this issue, pp. 2733–2752.
2. Haugk, G., Tsiang, S. H., and Zimmerman, L., "System Testing of the No. 1
    Electronic Switching Systems," B.S.T.J., 43, No. 5, Part 2 (September 1964),
    pp. 2575–2592.
3. Kienzle, H. G., Nicodemus, K. L., Smith, M. T., Jr., Weber, E. W., and
    Zydney, H. M., "No. 1 ESS ADF: Message Processing Program Organiza-
    tion," B.S.T.J., this issue, pp. 2753–2829.
4. Aitcheson, E. J., and Cook, R. F., "No. 1 ESS ADF: Maintenance Plan,"
    B.S.T.J., this issue, pp. 2831–2856.
5. Haugk, G., "Early No. 1 ESS Field Experience I- 2-Wire Systems for Com-
    mercial Applications," IEEE Trans. Comm. Technology, 15, No. 6 (De-
    cember 1967), pp. 744–750.
6. Tsiang, S. H., Haugk, G., and Seckler, H. N., "Maintenance of a Large Elec-
    tronic Switching System," IEEE Trans. Comm. Technology, 17, No. 1
    (February 1969), pp. 1–9.

Index to No. 1 ESS ADF Articles

Contributors to This Issue

EDWARD J. AITCHESON, B.S.E.E., 1962, Virginia Polytechnic Institute; M.S., 1964, Rutgers University; Bell Telephone Laboratories, 1962—. Mr. Aitcheson has been a member of the No. 1 ESS ADF Maintenance Planning and Development Department since 1967. At the present time, he is involved in maintenance planning for No. 4 ESS. Member, Tau Beta Pi, Eta Kappa Nu, Sigma Xi, Phi Kappa Phi, Kappa Theta Epsilon.

CYRUS F. AULT, B.E.E.E., 1950, University of Southern California; M.S., 1955, Stevens Institute of Technology; Bell Telephone Laboratories, 1955—. Mr. Ault has worked on ESS memory and circuit design since 1955. Member, IEEE, Eta Kappa Nu, Tau Beta Pi.

WILLIAM A. BAKER, B.E.E., 1953, Manhattan College; M.E.E., 1960, New York University; Bell Telephone Laboratories, 1953—. Mr. Baker was involved initially in the design of magnetic power apparatus. After a leave of absence to serve in the Civil Engineer Corps, U. S. Navy, he became involved in power supply design and magnetic memory development. He is presently Supervisor of the Memory and Digital Network Group. Member, Pennsylvania Society of Professional Engineers, IEEE, Eta Kappa Nu; Associate Member, Naval Institute.

W. R. BALDINGER, B.S.C.E., 1933, University of Illinois; Western Electric Company, 1935-1949; Bell Telephone Laboratories, 1949—. Mr. Baldinger initially worked on the development of equipment for local switching systems. Later he supervised a group responsible for the development of equipment for Automatic Message Accounting, Common Control and *Touch-Tone®* calling features. Since 1962, Mr. Baldinger has been Supervisor of a group working on the design of electronic equipment for TSPS No. 1.

DUANE R. BARNEY, B.S.E.E., 1936, Newark College of Engineering; M.S.E.E., 1938, Massachusetts Institute of Technology; Western Electric Company, 1940–1949; Bell Telephone Laboratories, 1949—. Mr. Barney's first work concerned system design of weapon direction systems for Navy use. He later supervised a group responsible for testing and operation of a military electronic switching system and the ADF system. He presently supervises a group working on test model design for No. 4 ESS.

ADAM C. CARNEY, B.S.E.E., 1962, Polytechnic Institute of Brooklyn; M.E.E., 1964, New York University; Bell Telephone Laboratories, 1962—. In addition to the work described in this issue, Mr. Carney has been engaged in the development of four-phase data sets and the exploratory design of a voice-excited vocoder. He is presently concerned with the physical design of data communications systems. Member, Eta Kappa Nu, Sigma Xi, Tau Beta Pi.

G. T. CLARK, B.S.M.E., 1952, Bradley University; Western Electric Company, 1956–1961; Bell Telephone Laboratories, 1961—. Mr. Clark was first engaged in the physical design of Step-by-Step Common Control equipment and later worked on the design of 758C PBX equipment. He has coordinated the physical design of TSPS No. 1 equipment, including detail design of network and position subsystem switching equipment. He is currently working on physical design of new features for TSPS No. 1.

W. K. COMELLA, B.S.E.E., 1963, University of Colorado; M.S.E.E., 1965, Columbia University; Bell Telephone Laboratories, 1963—. Mr. Comella worked on design of circuits for PBX-Automatic Identified Outward Dialing and then maintenance programming for TSPS. He is Supervisor of the group responsible for network and trunk maintenance programs for the Automatic Intercept System. Member, IEEE, Eta Kappa Nu, Tau Beta Pi.

RICHARD F. COOK, B.S.E.E., 1953, Northeastern University; Bell Telephone Laboratories, 1953—. Mr. Cook's work has included circuit and system design of military switching and communications systems. More recently he has been engaged in the maintenance development and planning for the No. 1 ESS ADF and the No. 4 ESS. Member, Eta Kappa Nu, Tau Beta Pi.

JOSEPH E. CORBIN, B.S.E.E., 1930, Pennsylvania State University; Bell Telephone Laboratories, 1930—. Mr. Corbin first worked on radio receiver and radio distribution systems for apartment houses and hotels. He later was engaged in work on aircraft receivers and the development of the receiver for the Coastal Harbor system. During World War II he worked on communication equipment and fire-control radar. He then was involved in work on commercial radar for Great Lakes ore boats and supervised groups involved in the Navy Intercept study, the APS-64 bombing system, and the Doppler spectrum analyser for the DEW Line. He later supervised groups that developed circuits and logic for NIKE-ZEUS and the Time-Division Switch for UNICOM. He was responsible for the design of Input/Output for No. 1 ESS ADF. Currently he is responsible for the Time Division Grid for No. 4 ESS Toll. Member, Eta Kappa Nu, Sigma Tau.

G. A. CULP, A.A.S. (Electrical Engineering), 1962, DeVry Technical Institute; Bell Telephone Laboratories, 1962—. Mr. Culp has worked in the No. 5 Crossbar and No. 4 Toll Crossbar Physical Design departments. With the Toll Crossbar Department, he has been active in the development of the SPC No. 1A Piggyback Twistor Store and the No. 4 Toll Electronic Translator System. He is currently engaged in the development of Common Channel Interoffice Signaling.

CHESTER M. DAY, JR., B.A., 1958, Bowdoin College; B.S.E.E. and M.S.E.E., 1958, Massachusetts Institute of Technology; Bell Telephone Laboratories, co-op 1956-1958, 1958—. Mr. Day's early work was in the research area on remote maintenance for No. 5 Crossbar Offices, switching logic using cylindrical thin films, and a broadband switching system for the *Picturephone®* visual telephone system and data. He subsequently supervised the group which developed the central office portion of AIOD and did the exploratory development for the small AIOD system. Since 1966, he has worked on peripheral unit maintenance programming for TSPS No. 1. Member, Sigma Xi, Eta Kappa Nu, IEEE, ACM.

G. RAY DURNEY, B.S.E.E., 1962, University of Utah; M.S.E.E., 1964, Rutgers University; Bell Telephone Laboratories, 1962-1970. During his career at Bell Laboratories, Mr. Durney was engaged in the development of circuitry associated with the Stored Program Control Processor and Peripheral Circuits.

JAMES C. EWIN, B.S. (M.E.), 1950, University of Maryland; O. S. Peters Co., Washington, D. C., 1950–53; U. S. Army, Ballistics Research Laboratory, Aberdeen Proving Ground, Md., 1953–55; Bell Telephone Laboratories, 1955—. Mr. Ewin worked for ten years in the Switching Systems Engineering Division on line concentrators, Wide Area Data Service and No. 2 ESS. He is currently Head of the Data Systems Planning Department with responsibility for identifying potential data services based on customer needs, for analyzing maintenance operations, for providing economic and service models to evaluate data services and maintenance strategies and for providing requirements on test equipment and new data systems of which No. 1 ESS ADF was a particular case.

S. M. FITCH, B.S.E.E., 1960, and M.S.E.E., 1962, Purdue University; Bell Telephone Laboratories, 1963—. Mr. Fitch has worked on the development of data stations for message-switching applications, most recently as a supervisor in the Data Communications Systems Laboratory. He is currently attending New York University under sponsorship of the Doctoral Support Plan. Member, IEEE, Eta Kappa Nu.

PAUL K. GILOTH, B.A., 1942, Beloit College; B.S.E.E., 1947, Northwestern University; Illinois Bell Telephone Company, 1947–1950; Bell Telephone Laboratories, 1951—. Mr. Giloth worked initially on analog computer simulators for military applications. Following this he supervised development of a transistorized bombing and navigation system and the guidance computers for the NIKE-ZEUS ABM system. In 1961 he was appointed Head of the UNICOM Test Model Department and was responsible for digital terminal equipment and the store and forward message portion of the UNICOM system. In 1963 he became Head of the Data Switching Systems Department and was responsible for development of the No. 1 ESS ADF Data Switching System. As Head of the No. 1 ESS AUTOVON Department, he is now responsible for AUTOVON development and support programming and testing for No. 4 ESS. Member, IEEE, Sigma Xi.

JOHN A. HACKETT, B.S.E.E., 1959, University of Maine; M.S.E.E., 1961, New York University; Bell Telephone Laboratories, summer, 1958, 1959—. Mr. Hackett was first involved in improvements and additions to Step-by-Step circuits. Later he worked on improved main-

tenance of Step-by-Step offices using call-through testing and designed new permanent signal circuits for Step-by-Step offices. He also was engaged in the initial design of TSPS peripheral circuits. Presently he is Supervisor, TSPS No. 1 Position Subsystem Group. Member, IEEE, Tau Beta Pi, Phi Kappa Phi.

KENNETH A. HELLER, B.S.E.E., 1953, Lehigh University; Bell Telephone Laboratories, 1953—. Mr. Heller worked initially in the Military Electronics Area. He was later involved in work on the Navy Intercept Project, the SAGE air defense system, the TITAN ICBM system, and the UNICOM global military communication system. He supervised groups responsible for maintenance programming in electronic switching systems including UNICOM and No. 1 ESS ADF. Since 1966, he has worked on TSPS No. 1 and was Site Supervisor at the Morristown installation from the start of system testing throughout cutover. Since April 1969, he has been responsible for development of TSPS No. 1 growth procedures and recent change programs. Member, Pi Mu Epsilon, Eta Kappa Nu.

R. J. JAEGER, JR., B.A. (Math), 1951, Hofstra University; Bell Telephone Laboratories, 1951—. Mr. Jaeger started his Bell System career with the Long Lines Department in New York City in 1941. After serving as a Naval Aviator in World War II, he returned to Long Lines and in 1951 came to Bell Laboratories to do design work in the No. 4 Toll Crossbar System. He has worked on Toll Switching Systems, Telegraph Switching Systems, the Time Assignment Speech Interpolation (TASI) System used on overseas cables, the Panel and Step-by-Step Local Switching Systems, and in recent years Traffic Service Position System No. 1 (TSPS). He is Head of the Planning and Operational Programming Department (TSPS). Senior Member, IEEE; Member, Kappa Mu Epsilon, Sigma Kappa Alpha.

AMOS E. JOEL, JR., B.S., 1940, and M.S., 1942, Massachusetts Institute of Technology; Bell Telephone Laboratories, 1940—. Mr. Joel worked initially in the fields of relay engineering, crossbar system testing and in the fundamental development studies of telephone switching systems. During World War II, he was engaged in the design of circuits for early relay digital computers and for cryptographic and cryptanalysis machines. Subsequently, he was concerned with the preparation

of texts for and in the teaching of switching design, the design of auto-
matic message accounting computer circuits, and the making of funda-
mental engineering studies of new switching systems. He was Head of a
department responsible for the development planning of the Bell Sys-
tem's first electronic telephone switching systems. Mr. Joel served as
Director of the Common Systems Switching Laboratory from 1961
to 1967. He is presently a Switching Consultant. Fellow, IEEE; Mem-
ber-at-large, IEEE Communication Technology Group; Chairman of
the Board of Directors, IEEE International Communications Con-
ference; Member, IEEE Communication Switching Committee, Sigma
Xi, Association for Computing Machinery, American Association for
the Advancement of Science.

HERBERT W. KETTLER, B.S.E. (Physics), 1964, and M.S.E.E., 1965,
University of Michigan; Bell Telephone Laboratories, 1966—. Mr.
Kettler has worked on the TSPS peripheral recognition program and
emergency action program design and implementation, support com-
puter systems for switching laboratories and TSPS field support.
Member, IEEE, ACM, Phi Kappa Phi, Tau Beta Pi.

A. W. KETTLEY, B.S.E.E., 1952, University of Vermont; Bell Tele-
phone Laboratories, 1952—. Mr. Kettley was initially assigned to the
Step-by-Step development group where he took part in current engi-
neering for Step-by-Step, the development of Touch-Tone® calling for
Step-by-Step, and the design of common control for Step-by-Step. He
later supervised an exploratory development group investigating the
modernization of Step-by-Step. During the TSPS development, he
supervised several groups involved in software design and is currently
in charge of the TSPS Operator Actions Programming Group.

HARRY G. KIENZLE, B.S.E.E., 1953, Dexel Institute of Technology;
M.S. (Mathematics), 1961, Stevens Institute of Technology; Bell Tele-
phone Laboratories, 1953—. Mr. Kienzle first participated in the design
of bombing and navigational computers. Upon return from a two-year
tour of duty with the U. S. Army Missile Research and Development
Division, he was concerned with the development of the guidance com-
puter for the NIKE-ZEUS ABM system. He later supervised a group
responsible for system design of the UNICOM communication system

and then a group responsible for the operational programs for the No. 1 ESS ADF. He presently heads a department concerned with No. 4 ESS maintenance. Member, IEEE, Eta Kappa Nu, Tau Beta Pi, Phi Kappa Phi.

GEORGE W. KINDER, E.E., 1961, University of Cincinnati; M.E.E., 1963, New York University; C&P Telephone Company of West Virginia, 1958–1960; Bell Telephone Laboratories, 1961—. Mr. Kinder supervises a group engaged in the design of Peripheral System Units for various Bell System applications. Member, IEEE, Tau Beta Pi, Eta Kappa Nu.

H. R. LEHMAN, B.S.E.E., 1962, New York University; M.S.E.E., 1964, New York University; IBM Watson Research Center, 1965–1966; Bell Telephone Laboratories, 1962–1965, 1966—. Mr. Lehman has worked in circuit and logic design in the development of electronic data switching systems. He is currently engaged in the design of high-speed switching systems. Member, Tau Beta Pi, Eta Kappa Nu.

FRANK H. MYERS, B.S.M.E., 1958, and M.S.E.E., 1960, Ohio University; Bell Telephone Laboratories, 1960—. Initially Mr. Myers was involved in the development of the switching network and crosspoint for No. 1 ESS. He then supervised the development of the P.B.T. store. Currently he supervises a group that is developing computerized facilities for development and production testing. Senior Member, IEEE.

KEITH L. NICODEMUS, B.S.E.E., 1952, University of Iowa; M.S., 1959, Stevens Institute of Technology; Bell Telephone Laboratories, 1952—. Mr. Nicodemus has worked on system development aspects of the acquisition radar for NIKE-ZEUS and on the logic design of the Time Division Switch for UNICOM. On the No. 1 ESS ADF project, he has worked on the Buffer Control design specifications, system coordination, design of audit programs and the debugging of system program and equipment. Since February 1970, he has been working with the logic design of equipment for the No. 4 ESS project. Member, IEEE, Tau Beta Pi.

G. PARKER, B.E.E., 1953, Brooklyn Polytechnic Institute; Graduate of Communications Development Training Program, Bell Telephone Laboratories, 1956; M.S.E.E., 1961, Columbia University; Bell Telephone Laboratories, 1953—. Mr. Parker has worked on teletypewriter station and central office control circuits and data sets. He is currently engaged in development of *Data-Phone*® data sets for low- and medium-speed applications. He is supervisor of the Frequency-Shift Modem Group in the Voiceband Data Department. Member, Tau Beta Pi, Eta Kappa Nu, Sigma Xi, Alpha Phi Omega.

E. J. PASTERNAK, A.B., 1957, Harvard University; M.S.E.E., 1962, and E. E. Professional, 1966, Columbia University; Bell Telephone Laboratories, 1962—. Mr. Pasternak initially worked in Exploratory Development where he performed studies on data processing structures. Much of this work was incorporated into the hardware design of the SPC processor. He later supervised the group that developed the Recent Change and Audit programs for TSPS No. 1. He presently heads the SPC Development Department, with responsibility for both the hardware and software design of the SPC common systems processor.

J. L. POTTER, B.S., 1964, University of Iowa; M.S., 1966, Stevens Institute of Technology; Bell Telephone Laboratories, 1964—. Mr. Potter was initially involved in programming of the UNICOM ESS. After completing the program design training program, he worked on maintenance programming for the No. 1 ESS ADF. Currently he is engaged in fault recognition maintenance design of the No. 4 ESS time division switch. Member, Phi Beta Kappa, Phi Eta Sigma.

EDWARD M. PRELL, B.S.E.E., 1962, University of Kentucky; M.S.E.E., 1964, Columbia University; M.M.S., 1969, Stevens Institute of Technology; Bell Telephone Laboratories, 1959—. Mr. Prell has been concerned primarily with the design of the SPC 1A processor and the problems of providing automatic maintenance facilities for electronic switching systems. He presently supervises a group responsible for the maintenance programming for the processor and the systems test laboratory. Member, Eta Kappa Nu, Tau Beta Pi.

GEORGE RIDDELL, B.E.E., 1947, and M.E.E., 1954, City College of New York; Western Electric Company, 1942–1951; Bell Telephone Laboratories, 1951—. At Bell Labs., Mr. Riddell has been engaged in the development of Step-by-Step, Panel, No. 1 Crossbar and the TSPS No. 1 systems. At present he is Supervisor of the SPC-1A Processor and Peripheral Circuits Development Group in the TSPS No. 1 Laboratory.

WILLIAM B. ROHN, B.S.E.E., 1951, Polytechnic Institute of Brooklyn; Bell Telephone Laboratories, 1954—. Mr. Rohn performed operations analysis studies for the SAGE air defense system and missile projects. He worked on the development of a traffic simulator for the UNICOM project and the development and testing of call processing programs for the 4-wire No. 1 ESS system for the government. During the initial development of TSPS, he supervised a group that developed mainten-ance programs for the SPC equipments. He presently supervises a group responsible for SPC programming and planning for extended trunking arrangements.

D. A. SCHMITT, B.S.E.E., 1965, St. Louis University; M.S. (Math), 1968, Stevens Institute of Technology; Southwestern Bell, 1962–1965; Bell Telephone Laboratories, 1965–1969. At Bell Labs, Mr. Schmitt worked on the maintenance control, maintenance restart, and utility programs for TSPS No. 1. Member, Eta Kappa Nu, National Society of Professional Engineers, Pi Mu Epsilon, Alpha Sigma Nu.

M. F. SIKORSKY, B.S.E.E., 1961, Newark College of Engineering; M.E.E., 1963, New York University; Bell Telephone Laboratories, 1961—. Mr. Sikorsky was initially associated with exploratory studies of stored program common control for Step-by-Step local office equip-ment where he did both hardware and software design. In TSPS No. 1, he has worked in the call processing program design area and at pres-ent is Supervisor of the Planning and Requirements Group. Member, IEEE, Eta Kappa Nu, Tau Beta Pi.

M. T. SMITH, JR., B.S.E.E., 1961, Clarkson College of Technology; M.S.E.E., 1963, New York University; Bell Telephone Laboratories, 1961—. Mr. Smith has worked in the development of operational soft-ware for both the UNICOM and the No. 1 ESS ADF systems. He is currently involved in maintenance software development for the No. 4 ESS Toll system. Member, IEEE, Eta Kappa Nu, Tau Beta Pi.

R. G. SPENCER, B.S.E.E., 1966, Oregon State University; M.S.E.E., 1968, Northwestern University; Bell Telephone Laboratories, 1966—. Mr. Spencer worked on Magnetic Disk Systems for No. 1 ESS ADF. As a member of the Processor Design Department, he is now involved in the design of advanced mass storage systems.

MRS. FRANCES B. STREBENDT, B.S. (Mathematics Education), 1965, Eastern Illinois University; Bell Telephone Laboratories, 1965—. Mrs. Strebendt worked on the design and development of message retrieval and journal file retrieval software for No. 1 ESS ADF. Presently she is involved in the development of an automated process in which simulation tests and results are used in generating diagnostics for No. 4 ESS.

R. M. TAYLOR, Bell Telephone Company of Pennsylvania, 1946–1961; Bell Telephone Laboratories, 1961—. Mr. Taylor was first engaged in circuit development for the No. 12 Service Observing Desk and the Mechanized Service Observing System. Since joining the TSPS No. 1 Development, he has been responsible for the design and application of tests for system evaluation.

RAY C. TOWNLEY, M.E., 1944, and M.S.E.E., 1951, Stevens Institute of Technology; Bell Telephone Laboratories, 1946—. Mr. Townley was first engaged in studying dial speed limitations of local signaling circuits. He subsequently performed circuit and logic design for military systems and participated in flight tests of the first flyable transistorized digital computer in 1958. Later he established timing specifications and designed units of the NIKE-ZEUS and UNICOM systems. He coordinated development efforts of the DSD for No. 1 ESS ADF. At present he is working on the Automatic Intercept System No. 1A. Member, IEEE.

ERICH W. WEBER, B.S.E.E., 1958, Case Institute of Technology; M.S.E.E., 1960, University of Arizona; Bell Telephone Laboratories, 1960—. Mr. Weber has worked on the design of operational programs for two message switched systems: UNICOM and No. 1 ESS ADF. He presently supervises a group responsible for the design of No. 1A ESS administrative programs. Member, Tau Beta Pi, Eta Kappa Nu.

JOHN R. WILLIAMS, B.E. (E.E.), 1960, Vanderbilt University; M.S. (E.E.), 1961, University of Illinois; Teaching and Research Assistant, University of Illinois, 1960–1961; U. S. Navy, 1961–1964; Bell Telephone Laboratories, 1964—. Mr. Williams was involved in system design, system test hardware development, operational software design, and system verification for the No. 1 ESS ADF through 1969. Currently he is engaged in the design of autonomous scanning and signal distributing hardware for the No. 4 ESS. Member, Tau Beta Pi.

H. M. ZYDNEY, B.A., 1954, Columbia College; B.S., 1955, and M.S.E.E., 1959, Columbia University; Bell Telephone Laboratories, 1959—. At Bell Labs., Mr. Zydney has contributed to the development of data transmission equipment for TWX and DATREX services. He supervised a group which designed the operational programs for station control in the No. 1 ESS ADF. He now heads the Data Network Control Department responsible for switching planning and circuit design for digital data network services. Member, Tau Beta Pi, Eta Kappa Nu.

CONTENTS

(Continued from front cover)

Bell System