Service Features and Call Processing Plan

By ROBERT J. ANDREWS, JOHN J. DRISCOLL, JOHN A. HERNDON, PHILIP C. RICHARDS, and L. RALPH ROBERTS

(Manuscript received April 4, 1969)

In No. 2 ESS modern telephone features are provided by concise call processing programs, which make extensive use of subroutines. This paper discusses the features to be available with the introduction of the system. It describes the call processing program plan used to achieve efficient use of program storage and gives details of supervision processing, digit handling, peripheral equipment control and translation. This paper also describes how call processing programs control the progress of an intraoffice call, and how they are used in testing of trunk and service circuits.

I. INTRODUCTION

The No. 2 Electronic Switching System performs the functions of a local telephone central office under the control of a stored program acting through data processing, input-output, and two-wire switching equipment. Virtually all the actions of the system are determined by the sequences of instructions coded and stored in memory.

The No. 2 ESS stored program may be divided into two parts: the call processing programs, which provide telephone service and operational features, and the maintenance and administrative programs, which maintain an operational system in the presence of troubles and diagnose the faulty units. Thus, the purpose of the stored program is identical to the purpose of the central office itself, including the implicit function of assuring dependable service.

This paper deals with the service features of No. 2 ESS, the call processing programs which provide these features, and with testing lines, trunks, and service circuits which use the call processing programs. The maintenance and administrative programs, the circuits and the equipment for No. 2 ESS are described in other papers.[1–4]

II. SERVICES AND OPERATIONAL FEATURES

The usefulness and value of No. 2 ESS to a telephone operating company will be determined by its service and operational features. The service features provide for the needs of telephone customers and the continual evolution of services throughout the life of the system. The operational features permit interfacing with transmission and signaling facilities, station equipment, and other switching systems. Additional operational features provide an efficient interface with operating company personnel by including facilities for day-to-day maintenance and administration,[1] for taking traffic and performance measurements and for recording data for computation of service charges.

2.1 *Service Environment and Performance*

No. 2 ESS is designed to provide service and operational features consistent with modern customer service needs and the environment in which medium size central offices may be expected to operate.[5]

The No. 2 ESS design anticipates a wide diversity of customer usage statistics from installation to installation. For example, 80 percent of Bell System central offices with between 1,000 and 10,000 lines have average busy-season busy-hour characteristics within the following ranges:

(*i*) Originating plus incoming calls per line	0.7–2.1
(*ii*) Originating plus incoming hundred-call-seconds per line	1.3–2.9
(*iii*) Originating traffic completing within the office	32%–92%

No. 2 ESS installations will be individually arranged or engineered using the modular attributes of the peripheral equipment design to provide a quality of service consistent with current operating company standards.[3] Table I shows the expected switching service performance objectives for No. 2 ESS.

The actual performance of an individual installation may occasionally fail to meet these standards when traffic volumes or patterns exceed the operating company forecasts which were used to engineer the equipment quantities, or if the equipment incurs physical damage.

2.2 *Telephone Services*

In general, No. 2 ESS provides the full complement of telephone services necessary in any modern telephone switching system. Although their implementation within No. 2 ESS is unique in many

TABLE I—SWITCHING SERVICE PERFORMANCE OBJECTIVES FOR No. 2 ESS

Dial tone speed (busy season—busy hour)	Probability $\leqq$ 0.015 of delay $\geqq$ 3 seconds
Line to line connections	Blocking probability $\leqq$ 0.04
Incoming trunk to line connections	Blocking probability $\leqq$ 0.02
Call processing irregularities	Probability $\leqq 1 \times 10^{-4}$

respects, the customer operating procedures and resultant system responses have been made uniform and consistent with other switching systems. These services can be considered in two categories, standard and custom.

2.2.1 *Standard Telephone Service*

Standard service consists of all dialable calls from and to customer lines using dial pulse signaling and 20 Hz ringing. In No. 2 ESS, service has been implemented to be consistent with existing switching systems. The system has flexibility to adapt to evolving needs and provide future services economically.

The term "all dialable calls" refers to the ability to accept and derive appropriate routing for all telephone numbers used in current practice. These range from the single digit zero (operator), through three digit service codes such as 411 (directory assistance) or 911 (emergency), up to ten digit numbers for direct distance dialing. Included in the design are capabilities which anticipate modifications of the existing number plan, for example, single digits (one or zero) as prefixes and arrangements for international direct distance dialing.

In addition to determining the appropriate route or specific line for each valid number, No. 2 ESS provides standard signals, tones, and recorded announcements. These are used to ring station equipment in completing calls and for notifying originating customers or equipment of each call's status. The audible ringing, busy and reorder tones are returned to the originator as notification that the directed destination is being rung, is busy, or that traffic or equipment conditions preclude reaching the destination. These tones are distinct sequences of precision tones for potential automatic recognition by station equipment.

Additional standard services, such as coin station and PBX calls, generally require additional switching functions and special electrical interfaces.

Coin service requires additional control and signaling functions to test for coin presence and to collect or return the deposit. No. 2 ESS is arranged to provide two versions of coin service. One allows dial tone before an initial deposit and free completion of operator or emergency calls. The other requires an initial deposit before any call can be originated.

In serving manual PBX systems, No. 2 ESS uses basic loop supervisory signaling on the group of lines to the PBX and also hunts for an idle member of the group when the directory number of the PBX is called. For dial PBXs and similar customer systems or equipment, a "ground start" supervisory signaling mechanism is utilized instead of a "loop" method in order to electrically exchange "busy-idle" and "start dial" state information with PBX equipment.

2.2.2 *Custom Services*

Custom services as supplements to the traditional standard services will also be available in No. 2 ESS to allow customers the option of further improving the efficiency and utility of their service.

Touch-Tone® calling requires the system to accept originations from such stations in addition to originations from conventional dial stations. When a *Touch-Tone*® telephone originates a call, the No. 2 ESS connects signal receiving equipment capable of recognizing frequency pairs.

Speed calling service allows a customer to specify a frequently used number by dialing only one or two digits instead of the full seven or more. For each customer who subscribes to speed calling, the system retains a list of abbreviated codes and the unique number assigned to each. The system also accepts dialed instructions from customers to add to or modify their own lists.

Customers who subscribe to three-way calling can add a third party to an existing connection by alerting the switching system with a supervisory flash (momentary on-hook), and then dialing the number for the added party.

A customer with call waiting service will, when he is engaged in a call, be notified with a tone if an additional call is directed to his number. He can then use a supervisory flash to alert the system to hold his original connection and connect him to the new call. Successive flashes will result in alternating connections between the two calls.

Call forwarding service allows a customer to direct the switching system to forward all calls for him to another number until he deactivates the service. Care has been taken in planning this service to reduce potential annoyance to other customers. The switching system automatically places a call to the forwarded number at the time the service is activated to encourage a customer to both verify the intended destination and to exercise the courtesy of notifying the other customer of the impending forwarded calls. In addition, as each subsequent call is forwarded the original number is rung briefly as a reminder that the service is active.

2.3 *Operational Features*

The significant operational features provided by No. 2 ESS are listed in Table II and are organized in three categories:

(*i*) Customer—for interchange of information with customers via their station or PBX equipment.
(*ii*) Intersystem—for operation with other switching systems.
(*iii*) Administrative—to interface with operating company personnel or with specialized systems where a function has been automated.

The manner in which the program uses customer class of service information, dialed digits, and the operational characteristics of equipment to implement these operational features is described in the following sections.

III. SMALL OFFICE PROGRAM APPROACH

No. 2 ESS is expected to be used primarily in those Bell System central offices where the telephone company is more concerned about cost than about maximum traffic capacity. As a result, substantial savings are obtained by emphasizing call storage and program storage economy rather than call handling capacity. Also, the use of a simple processing hierarchy leads to additional storage economy at some expense in system efficiency.

3.1 *Use of Subroutines to Reduce Program Storage*

The subroutine is widely used in No. 2 ESS to keep program size small, even though some extra real time is used in preparing data and transferring to the subroutine. Since additional program economy can be obtained by nesting subroutines (subroutines calling other subroutines), the No. 2 ESS processor is designed to readily per-

TABLE II—OPERATIONAL FEATURES

Customer

Supervision: loop and ground start plus sleeve lead function.
Pulsing: dial pulse and *Touch-Tone®* calling with precision dial tone.
Ringing: 20 Hz from ac-dc or superimposed equipment.
Prepay and dial tone first coin service.
Reverting call sequences.
Hotel-motel message register signals.
Line polarity reversals for PBX toll diversion.
Busy verification and ringback for operator or local test desk.
Party identification and foreign potential tests.
Permanent signal and partial dial recognition.
Announcements (6 channels).

Intersystem

Supervision: loop and E&M
Outpulsing: dial pulse and multifrequency (13 digits) with delay dialing, wink start or stop-go controls.
Inpulsing: dial pulse and multifrequency (13 digits) with immediate dial or wink start control.
Automatic number identification information to an adjacent switching center.
Alternate routing and foreign area translation.
Operator switchboard or traffic service position interfaces with inband coin control.
Directory assistance (information) interface.
Test desk interface.
Automatic intercept system interface.

Administrative

Maintenance personnel interface: teletypewriter channel and local control panel for: (*i*) equipment status, (*ii*) detail trouble diagnosis and clearance and (*iii*) making lines, trunks, and service circuits busy for maintenance.
Automatic fault recognition and diagnosis of major components with automatic system reconfiguration to sustain service.
Measurements of service and performance characteristics: internally assembled and recorded via teletypewriter channels.
Interface for external service observations.
Changes to customer service information via teletypewriter channel.
Automatic message accounting data: magnetic tape recording or station identity transmittal to a central automatic message accounting center.
Line load control: self monitoring.
Emergency system recovery facility to restore call processing capability in the presence of errors.
Audits of call store records to assure consistency.
Automatic line insulation testing: remote control via teletypewriter channel.
Trunk testing: manual dial access to individual trunks, automatically sequenced outgoing tests and terminations for incoming tests.
Line testing: dial access.
Office alarm system interface: attended or unattended.
Automatic overload controls.

mit using several levels of nesting. Thus, the lowest level subroutine does a very specific task in a very straightforward manner. Higher level subroutines have larger tasks to perform but these tasks are accomplished primarily by moving data appropriately and calling lower level subroutines for further action. This structure leads to a basic call handling program that consists mainly of calls to subroutines.

3.2 *Economical Use of Call Storage*

Call storage economy is achieved through the use of small call records and by assuring that only truly variable information is stored. Small call records require efficient packing of information. Several different items are packed in the same word, even though the packing and unpacking takes extra time. Translation data concerning participants in a call is not kept in the call record and must be obtained from program store whenever it is needed. In general, any data that can be reconstructed from other information is not retained in call store.

Programs examine the call records and take action as a result of any new information plus the information contained in the record. Although it is the program which takes action, it is convenient to refer to the record as performing the function, and this convention is followed here.

3.3 *Simple Processing Hierarchy*

Further program and call storage savings are achieved through the use of a simple processing hierarchy consisting of a main program level, a timed interrupt program level, and a demand interrupt program level. Although the demand interrupt takes precedence over the timed interrupt, there are virtually no priorities within a given level. Instead, all tasks are handled as they are encountered and each program does as much work as possible on one job before going on to the next. The demand interrupt occurs only from an indication of machine failure or from a manual request. The timed interrupt is used for only those tasks where a small delay may cause an error or a failure. All other tasks are handled on the main program level and in a random sequence without regard for the urgency of the task. A more frequent interrupt would be required were it not for a wired logic unit in the processor which does the scanning for customer dial pulses in place of the stored program. The function of this unit is explained in more detail in Section 5.2. This simple hierarchy saves program storage and call storage because as a task occurs it is performed regardless of priority instead of being passed to a task distributor program which would have to record details of each task for later execution. The only effect of this random sequencing is a slight, but unnoticeable delay for a few calls.

IV. BASIC CALL PROGRAM SYSTEM

4.1 *Main Program Loop*

The overall program control plan is shown in Fig. 1. The call processing functions are controlled by the main program loop. One hundred milliseconds is the normal time required to complete one pass through this loop. To handle calls as quickly as possible, a new main program loop is usually initiated following the particular timed interrupt which delivers most new supervisory reports to the supervision distributor.

The 100 millisecond loop time represents a compromise between the speed with which the system can react to new inputs for individ-

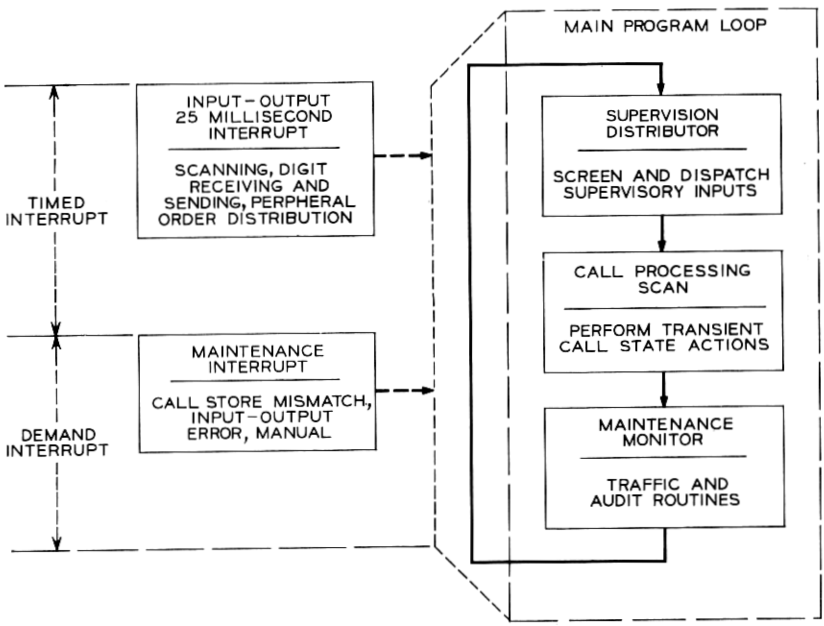


Fig. 1 — Program control plan.

ual calls and the efficiency with which it can process all calls. With a 100 millisecond loop interval, the amount of time wasted accessing calls that have no new information is less than 5 per cent while 100 milliseconds is the maximum delay before a new input is processed.

The supervision distributor is the first function in the loop. It distributes each new supervisory report to either the call record with which the report is associated or to a new call record if a supervisory report represents a new call.

The call processing scan accesses the transient call record associated with each nontalking call in the system. Any new information associated with a call is processed and an appropriate output response is generated.

The maintenance monitor is entered after all of the transient call records have been accessed. This program performs essential maintenance and administrative tasks including recording various traffic data, processing teletypewriter messages, and necessary audits to insure that the system is operating normally. Other maintenance tasks which may be deferred are handled as time allows.

4.2 *Input-Output 25 Millisecond Interrupt*

The timed interrupt used by call processing for input and output functions occurs every 25 milliseconds. The interrupt stops the main program, stores its program location and all machine registers in variable storage, and initiates those tasks required at that time. Although all of the tasks performed by the interrupt need to be performed more often than could be done by the main program loop, very few need to be done every 25 milliseconds. The various tasks are distributed among several interrupts at their required frequency as shown in Table III. This distribution of tasks attempts to equalize the amount of time required by each 25 millisecond interrupt. Notice that a few of the tasks must be executed every 25 milliseconds and these tasks determine the minimum time between interrupts.

4.2.1 *Detection of Supervision*

The detection of supervision falls into several categories that require different frequencies of processing. The most frequent is associated with line service requests. A wired logic facility allows the system to detect one line service request at a time. Because one request must be processed before the next one may be detected, the line scanner check occurs every 25 milliseconds to avoid a momentary limitation on service requests. Program scans of trunk circuit and

TABLE III—FREQUENCY OF TASKS PERFORMED DURING TIMED INTERRUPTS

Task*	See section	Input-output 25 millisecond interrupts (in milliseconds)																		
		0	25	50	75	100	125	150	175	200	225	250	275	300	325	350	375	400	425	450
Line scanner check	4.2.1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Line hit list rescan	5.1		X		X		X		X		X		X		X		X		X	
Fast trunk scan	4.2.1		X		X		X		X		X		X		X		X		X	
Normal trunk scan	4.2.1		X				X				X				X				X	
Trunk hit list rescan	5.1				X				X				X				X			
Tone digit check	4.2.2		X		X		X		X		X		X		X		X		X	
Dial pulse digit check	4.2.2				X					X					X					X
Sending check	4.2.2		X		X		X		X		X		X		X		X		X	
Multifrequency digit turn off	5.2.2					X						X						X		
Peripheral order distribution	4.2.3	X		X		X		X		X		X		X		X		X		X
Teletypewriter input-output	4.2.4	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Automatic message accounting output	4.2.4	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Maintenance functions	4.2.4	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

* Task is performed during interrupts marked X.

service circuit supervisory indications occur at either a 50 or 100 millisecond rate.

The 50 millisecond rate of the fast trunk scan is used for those trunks that can generate supervisory signals that last less than 100 milliseconds, such as an operator trunk which can generate an on-hook wink of between 70 and 130 milliseconds. In order to insure detection of this wink, these circuits must be scanned at least once every 70 milliseconds; so to be compatible with the system interrupt structure, they are scanned every 50 milliseconds.

Also scanned during the fast trunk scan are trunks from step-by-step switching systems which can present a seizure to the system followed almost immediately by customer dialing. The system must scan these trunks often enough to insure detecting the seizure before the first dial pulse comes from the customer. All other supervisory indications are scanned using the normal trunk scan at a 100 millisecond rate which represents a compromise between system processing efficiency and system response time to a given change of supervision.

4.2.2 *Digit Receiving and Sending*

Digit receiving and sending involves the receipt of dial pulse, tone, and multifrequency digit information, and the sending of dial pulse and multifrequency digits. Each one of these functions has its own set of timing constraints which must be adhered to. The timing sequences involved in the digit processing facility represent a combination of all these timing constraints.

Since the minimum time between tone or multifrequency digits is about 80 milliseconds, the program does a tone digit check every 50 milliseconds. At the same time it must be capable of receiving dial pulse digits, since at the time a party requests service it may not be known whether he will send dial pulse or tone digits. Dial pulse reception requires the discrimination between a disconnect from the customer and a dial pulse. Further, the system must discriminate between the off-hook periods separating two dial pulses and the off-hook period following the end of a dial pulse digit. Neither the on-hook nor the off-hook portion of a dial pulse should exceed 125 milliseconds.

A disconnect signal or the off-hook time between two digits should always exceed 250 milliseconds. Therefore, the system does a dial pulse digit check every 125 milliseconds. It concludes that the subscriber has disconnected if he is on-hook for two successive dial pulse digit checks without having been off-hook in between. It concludes that the subscriber has finished dialing a digit if he is off-hook for two successive dial pulse digit checks without being on-hook in between.

For sending operations, multifrequency digits require a 150 millisecond cycle and dial pulse digits require a 100 millisecond cycle. The greatest common multiple, 50 milliseconds, is thus used as the interval for performing a sending check for any sending operation which may be required.

4.2.3 *Peripheral Order Distribution*

Distribution of orders to peripheral circuits during the timed interrupt is necessary to assure efficient use of peripheral control circuitry. Because sequences of orders are often involved in setting up or changing paths through the network, the total time required for sequences must not exceed the maximum time permitted for the temporary interruptions of connections. Therefore, peripheral order distribution occurs every 50 milliseconds and, where appropriate, circuits are interrogated for responses from previous orders at the same time.

This 50 millisecond rate is chosen to coincide with the operate time of the network control, which requires slightly less than 40 milliseconds to complete its most complicated action. Since it can take up to 8 milliseconds to complete peripheral order distribution during any one period, the 50 millisecond rate of sending these orders assures that the network controller used at the end of one series of distributions is available for use at the beginning of the next series.

4.2.4 *Other Interrupt Functions*

Other functions performed during the timed interrupt include teletypewriter input-output, automatic message accounting output, and several maintenance functions. The teletypewriter input-output work is handled once every 25 milliseconds during the interrupt to assure that no characters typed by the craftsman are missed and that the output typing rate is constant. Automatic message accounting output data are distributed to an incremental tape recorder every 25 milliseconds to assure an adequate capacity with a maximum call rate system.

Maintenance functions handled during the timed interrupt are those routine tests which require extremely critical timing. Although routine maintenance is normally considered quite low priority, some tests which involve hardware response times require critical timing once the test is initiated. This timing is so critical that any time variation resulting from the amount of work done during the interrupt cannot be tolerated. Therefore, these maintenance functions are performed at the beginning of the interrupt if they are needed.

4.3 *Call Processing Control*

The primary call processing control in No. 2 ESS rests with the set of programs associated with the transient call records. A transient call record is associated with each active nontalking call in the system and consists of 8 call store words, 16 bits wide as shown in Fig. 2. The transient call record controls the progress of the call for the interval starting with dialing through ringing until an answer signal has been received and the connection is in the talking state. There are up to 256 of these transient call records in a No. 2 ESS, each independently handling its own telephone call. Since the transient call record handles calls in all possible states, it is not practical to have a fixed layout of data within the entire eight-word block. Therefore, Fig. 2 serves as an example of a typical layout.

4.3.1 *Progress Marks*

The first word of the transient call record contains the address of the program used by the processor to reinitiate the processing of a call on each main program loop. This program address is called a

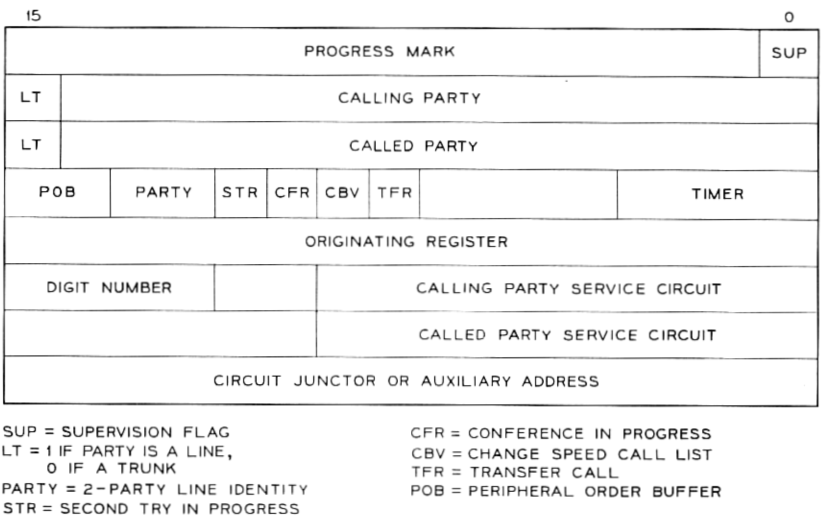


Fig. 2 — Transient call record.

progress mark and it must be provided whenever the transient call record is in use.

For example, if a party is dialing, the program addressed by the progress mark will inquire if any new digits are present or if the customer has delayed too long between digits. During the main program loop each transient call record is accessed and control is transferred to the program indicated by the progress mark, and that program then inquires if there is any new information associated with the call. If there is nothing new, the same progress mark is left in the transient call record and the next transient call record is accessed. When there is new information, the program acts upon that information and generates any output data required. If the completion of this action changes the state of the call, a new progress mark is written into the transient call record to reflect the new state.

The process of moving from one transient call record to the next is handled by a special advance instruction in the No. 2 ESS processor. This instruction relies upon the fact that all transient call records are adjacent to each other and all of them begin at a location whose low three bits are all zeros. In addition, a particular machine address register is used by the transient call record programs for reading and writing data in the transient call record, so that it always points at some address within the eight words of the transient call record.

When the advance command is executed, the next transient call record is determined by first zeroing the low three bits of the address register and then adding eight to the resulting address. The register then points to the first word of the next transient call record. The processor reads the progress mark and transfers control to the indicated program. Since the advance command performs this entire function, it is not necessary to actually return to the main control program between the processing of two transient call records. The program is not concerned with the absolute location of the transient call record, so that data is accessed simply by modifying the low three bits of the address register.

4.3.2 *The Transient Call Record Layout*

The second and third words of the transient call record almost always are used to hold the calling and called party's identification. The high bit of this word determines whether the party is a line or a trunk and the other 15 bits specify a line's terminal number or a trunk circuit's scan point number. The No. 2 ESS switching network is limited to fifteen 2,048 terminal networks or 30,720 terminals, and, therefore, 15 bits can be used to define any line terminal. It has been determined that twelve 1,024 point scanners are adequate to handle all trunks and service circuits in a No. 2 ESS office, and fourteen bits are used for this purpose. The use of the second and third words of the transient call record is limited to defining the calling and called parties so that tracing programs can find the transient call record associated with a particular line to determine information about the state of a call or to set up a no test connection.

The layout of the remaining five words of the transient call record are somewhat less standardized. The fourth word usually contains a set of flag bits which indicate various substates that do not affect the normal flow of call processing, but permit the same programs to handle several different situations. In addition, this word contains four bits which are used for various timing functions and two bits which are used for sequencing the delivery of orders to the periphery. The fifth word contains the address of the originating register, which receives and stores digits from the customer and also handles the sending of digits to another central office.

The low ten bits of the sixth word are used to store the identity of any service circuit which may be connected to the calling party identified in the second word. Likewise, the low ten bits of the seventh word contain the identity of a service circuit associated with the called party identified in the third word. The eighth word contains the identity of a circuit junctor used to connect the calling and called party. When an originating register or service circuit or circuit junctor is not required, these words may be used to store other useful information which aids in processing the call. For example, during dialing, when no circuit junctor is required and no service circuit is associated with the called party, class information for the calling party is stored in the seventh word and an auxiliary address is stored in the eighth word. This address is used in the processing of dialed digits.

V. PRIMARY PROCESSING FUNCTIONS

The programs used to process transient call records are called progress mark routines and they form the backbone of the call program structure of No. 2 ESS. All supervisory inputs are delivered to the transient call record, where they are interpreted by the progress mark routines. All new digits are examined and interpreted by the progress mark routines. These routines call various subroutines for purposes of translating information, selecting equipment, providing timing, and sending orders to peripheral equipment to complete connections for customers. The following sections describe most of these functions in detail, showing the relation among various circuit functions, interrupt program functions, and progress mark routines.

Figure 3 shows the basic information flow of the call processing system and indicates the relation between circuitry and program. It shows how information comes into the system from peripheral circuits through both wired logic in the processor and programs executed during the timed interrupt. The information thus gleaned is passed on to the progress mark routines through several different paths. The progress mark routines which are executed at the base level examine this information using various subroutines, and pass orders to the interrupt program for distribution to peripheral circuits. When this distribution has been completed, the interrupt program notifies the progress mark routines so that a new state of the call may be recognized.

5.1 *Processing of Supervision*

5.1.1 *Line, Trunk, and Service Circuit Scanning*

Two types of supervisory scanning are provided in No. 2 ESS. Line scanning detects requests for service from customer lines. Trunk and service circuit scanning detects changes in supervisory states associated with calls in progress and detects requests for service from incoming trunks. Trunk and service circuit scan points provide more changes of supervision than do the more numerous line scan points. These scan point changes may be of shorter duration and may require more rapid system response than line scan points. These characteristics lead to two entirely different methods for scanning.

No. 2 ESS line scanning takes advantage of the fact that a line's request for service is always an on-hook to off-hook change. Once a request has been served, the line ferrod is disconnected from the line, causing the scan point to go on-hook and remain on-hook for the duration of the call. When the ferrod is restored to the line at the end of a call, the customer is on hook so that the scan point continues to indicate on-hook. As a result, only those scan points which represent unserved customer requests for service are off-hook at any time. Once properly initialized, an input-output circuit in the No. 2 ESS processor automatically cycles through all of the line scan points row by row and looks for any lines that are off hook. When it finds an off-hook,

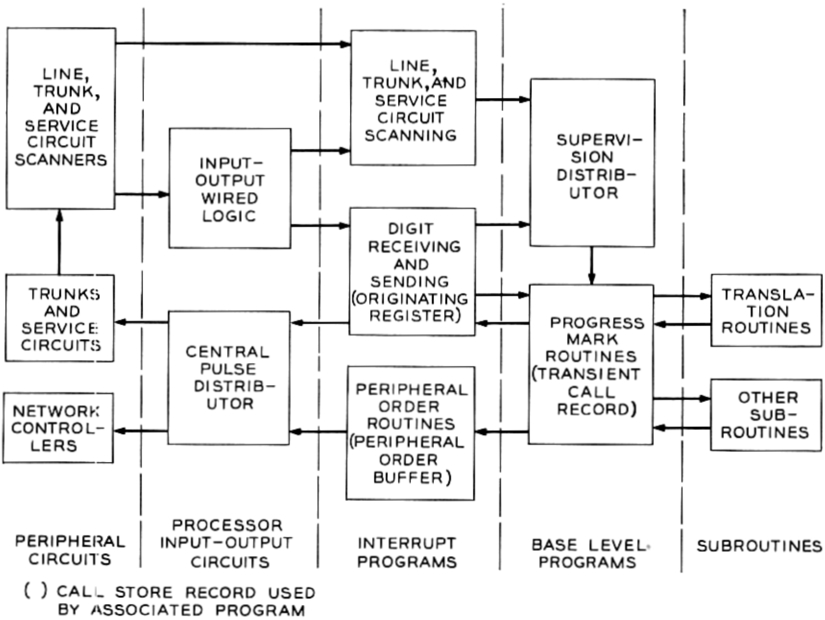


Fig. 3 — No. 2 ESS call processing information flow.

the circuit stops cycling and indicates the scanner row containing the line service request. Later the circuit is interrogated by program, the row identity recorded in a hit list for later use, and the circuit is reinitialized to start cycling from that point.

Since the significant supervisory changes from trunks and service circuits can be either off-hook to on-hook or on-hook to off-hook, it is necessary to scan these points by comparing the present state of the point with the last-look bit in call store which represents the previous state. This scanning is done in the input-output (timed) interrupt as described in Section 4.2.1.

For many reasons, supervisory changes lasting only a few milliseconds can occur and do not truly reflect a valid change in the state. On some occasions it is possible for the trunk scanning program or the line scanning circuit to detect these momentary hits. To reduce the number of hits passed on to the processing program, the address of the scanner row in which a change is detected, is stored in a hit list. 50 milliseconds later this same row is again scanned by program and if a change still exists, then a valid supervisory change is assumed to have occurred.

Table III shows the relative times of the line and trunk hit list rescans. The probability of a hit plus a valid change occurring in the same row at the time of the rescan is very low. The occasional hit that does get reported will be eventually ignored by the processing programs but it does require some additional processing time. Once a valid supervisory change has been detected, the scan point number is placed in one of three lists for eventual processing by the supervision distributor. There is a separate list for reporting line origination requests (origination hopper), trunk and service circuit off-hook reports (off-hook hopper), and trunk and service circuit on-hook reports (on-hook timing list). The latter two lists are also used by the supervision distributor for further timing of the supervisory changes.

The supervisory scanning programs include several defensive programming features which detect failures in the scanner circuits which may not be detected by other means. On the average, only about two scan point changes should occur in a 1024 point scanner during any single 100-millisecond program scan. Therefore, if many more than this occur, there is adequate reason to suspect a failure in the scanner and a more detailed diagnostic program is called to determine whether this occurrence is actually a failure or just a momentary peak in the number of scan point changes.

5.1.2 *Supervision Distributor*

The supervision distributor program which functions in the main program loop has two primary tasks to perform. First, supervisory changes detected in the interrupt program are timed and compared with other supervisory reports to further eliminate momentary hits and to also discriminate between flash signals and valid on-hooks. This timing function is only associated with trunk and service circuit scan point changes since no on-hook changes can be reported for line scan points. The two lists associated with this timing function are the off-hook hopper and the on-hook timing list.

All valid on-hooks must last for at least 250 milliseconds before they are reported. Therefore, the on-hook timing list is checked every main program loop for all scan point changes that have been in the list for longer than the 250 milliseconds. Any off-hook scan point change that follows an on-hook change by less than 250 milliseconds is a trunk flash. The flash is detected by matching all of the scan point numbers in the off-hook hopper against all of the scan point numbers in the on-hook timing list before the off-hooks are further processed. When a match is detected, a flash report is processed for that scan point rather than an off-hook.

The second task of the supervision distributor is to determine the state of the circuit which generated the report. There are three possible states that can occur: (*i*) no call in progress, (*ii*) transient or nontalking state, and (*iii*) stable, or talking state. The state of each circuit is recorded in a two-word call store record, called the terminal memory record associated with that circuit. If the call is in a transient state, the address of the transient call record associated with the call is stored in the terminal memory record as shown in Fig. 4.

Two bits are also stored in the terminal memory record to indicate its particular function in the transient call. To report a change, the supervision distributor accesses the transient call record indicated in the terminal memory record and checks the contents of the progress mark word. If the low bit of this word is a 1, the program changes that bit to a 0, writes it back into the progress mark word and

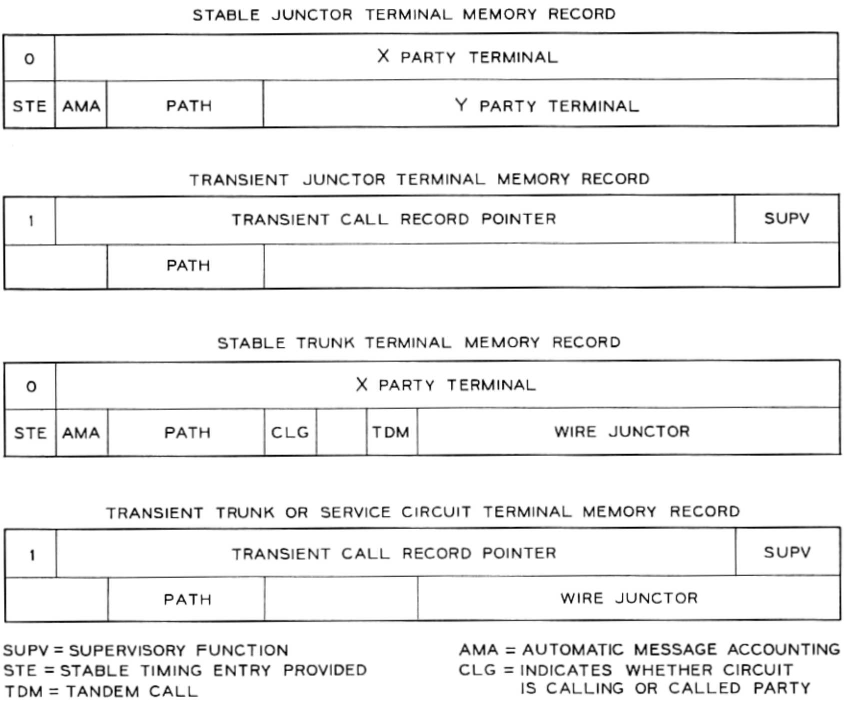


Fig. 4 — Terminal memory record formats.

also writes a five-bit code elsewhere in the transient call record to indicate the specific details of the supervisory change. If the low bit of the progress mark is a 0, supervision cannot be reported to the transient call record and the program moves the change into a queue for later distribution to the transient call record. This procedure assures that only one supervisory report will be delivered to the transient call record at a time, and permits the transient call record to block any supervisory reports simply by using a progress mark with its low bit 0 during those intervals when it would be impossible to properly interpret the supervisory report.

In other instances, if the terminal memory record indicates that the state of the circuit is idle or stable, the supervision distributor selects a transient call record and places the change report and the circuit identity in it. It also inserts a progress mark to address a program which will appropriately decode the supervisory change either as a request for service or as a disconnect signal in the case of a stable call. The terminal memory record is also updated to indicate its new transient state and the identity of the transient call record.

Line service requests are handled by the supervision distributor through the line origination hopper. Since lines have no terminal memory records, and since the only changes in this list represent requests for service, the supervision distributor attempts to select a transient call record for each service request in this hopper. Any unsuccessful attempts will leave the request in the hopper for further attempts. If a transient call record is selected, the line's terminal equipment number is stored in the second word and a progress mark is placed in the first word. The program addressed by the progress mark will later attempt to connect the line to a digit receiver.

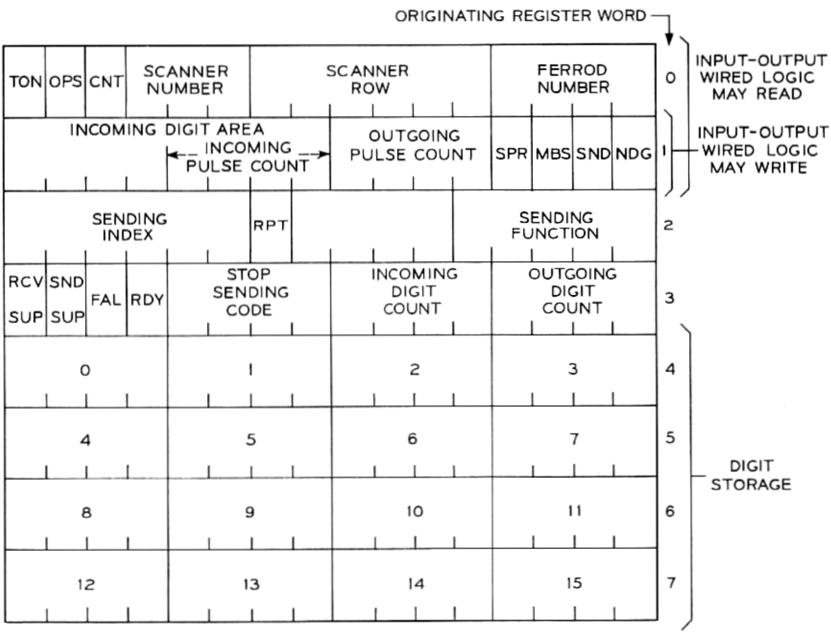
Flashes from trunks are detected by the supervision distributor, but flashes from customers are detected by progress mark routines which time the on-hook period after the change is reported to the transient call record. This method of handling customer flashes is used because only a few customer classes of service permit flashes and there are only a few states where a flash is treated differently from an on-hook followed by an off-hook. Only in those instances is timing necessary to determine if the customer is flashing.

5.2 *Processing of Digits*

All digit receiving and sending functions in No. 2 ESS use an originating register and a set of associated programs under direction of the progress mark routines (see Fig. 3). The originating register layout as shown in Fig. 5 consists of eight 16-bit call store words. The lower four words of the register contain space for storing up to 16 digits, each digit being represented by a four-bit binary coded decimal code. The upper four words of the originating register contain control data for the interfaces among the wired logic digit handling facility, the timed interrupt digit handling programs, and the main program level progress mark routines. The system has a capacity of up to 128 originating registers, but only one is associated with a particular call.

5.2.1 *Receiving*

There are three types of digits that must be received—dial pulse digits from lines or trunks, tone digits from lines, and multifrequency



CNT = CONTROLLER BIT
FAL = FAILURE BIT
MBS = MAKE-BREAK STATUS
NDG = NEW DIGIT FLAG
OPS = OUTPULSING SPEED
RCV = RECEIVER SUPERVISION BIT
RDY = READY FLAG
RPT = REPORT FLAG
SND = SENDING FLAG
SPR = SIGNAL PRESENT
SUP = SUPERVISION FLAG
TON = TONE PRESENCE BIT

Fig. 5 — Originating register.

digits from trunks. For dial pulse digits, the originating register is used to count the pulses detected by a scan point. In the case of tone or multifrequency digits, the receiving circuit indicates through a set of scan points which tones are present and through a signal present scan point that a proper combination of signals has been present for a minimum time. The receiver circuit may be arranged to permit both tone and dial pulse digits to be recognized, thus permitting parties on a party line to have both rotary dial and *Touch-Tone*® telephones.

A wired logic facility in the No. 2 ESS processor scans the originating registers at a 10-millisecond rate. Each time an originating register is accessed by the wired logic, the first word is read and the indicated row of 16 scan points is examined. In the case of dial pulse reception, the status of a single scan point is compared with the make-break status bit, and if a change from off-hook to on-hook has occurred, the bit is updated and the low bit of the second word, the new digit flag, is set to 0. If a change from on-hook to off-hook has occurred, the make-break status bit is updated, the new digit flag is again set to 0 and the incoming pulse count is incremented by 1.

In order to detect a completed digit, or a disconnect, the interrupt processing program scans through all the originating registers every 125 milliseconds and sets the new digit flag to a 1. If this flag remains 1 for another 125 milliseconds, and if the make-break status bit indicates an off-hook, the program assumes that a new digit is completed. If the make-break status bit indicates on-hook, then the program concludes that the line has disconnected.

When a digit is completed, the program removes it from the incoming digit area, stores it in an appropriate digit slot of the originating register and then zeros the incoming digit area. As long as the incoming digit area is all zeros and the digit receiver scan point used to supervise the call is off hook, the wired logic continues to zero the new digit flag, thus preventing the program from examining the incoming digit area when it is empty.

Tone and multifrequency digits are recognized by scanning and comparing the signal present scan point with the signal present bit. When a new signal is present, the states of six scan points representing the multifrequency digit or eight scan points representing the tone digit are recorded in the incoming digit area and the new digit flag is set to a 1. Since multifrequency and tone digits may come as often as once every 80 milliseconds, the program examines the originating registers once every 50 milliseconds. If a new digit is present, it is removed from the incoming digit area, converted to a binary coded decimal representation and stored in an appropriate digit slot in the last four words of the originating register.

The position for storing a completed digit is determined by the four-bit incoming digit count in the fourth word of the originating register. The program that moves a new digit in place determines its location from the incoming digit count which is then incremented to provide a pointer for the next digit. The program also sets the transient call record ready flag in the fourth word of the originating register. Approximately every half second the transient call record controlling a dialing call checks the originating register for the ready flag. If the flag is not set a time check is made to avoid tying up common equipment. If the flag indicates that a digit is present, the progress mark routine checks the incoming digit count and if enough digits have been received, the progress mark routine will then perform any appropriate action.

For example, after three digits have been received, the progress mark routine will attempt to determine the routing of the call and if it is a local call, after seven digits have been received, it will attempt to set up the appropriate ringing connection. Whether or not the transient call record actually interprets the digits, it resets the ready flag in the fourth word so that it will not have to check the originating register again until another digit is received, thus minimizing the amount of time spent checking until there is new information available. In the case of receiving multifrequency digits, the transient call record ready flag is not set when each digit is received; rather, it is set only when the start digit is received by the wired logic indicating that all of the digits are present. This arrangement is provided because it is not practical to translate the digits as they are received since the total number of digits can be variable.

5.2.2 *Sending*

There are only two types of digits which may be sent, dial pulse and multifrequency. To insure a high degree of precision, the timing functions needed to send these digits are under the control of the processor. Timing for output dial pulses is provided by a wired logic facility which distributes start and stop pulses over a timing bus to the dial pulse sending circuits.

Sending is initiated from a transient call record by a progress mark routine after a sender circuit has been connected to a trunk. At that point a sending index is inserted into the third word of the originating register along with the sending function. The progress mark routine places the identity of the first digit to be sent in the outgoing digit count area and the last digit to be sent in the stop sending code area.

The sending index points to a call store list that contains the scan point number and peripheral decoder* address of the sender circuits. The scan point number is needed by the interrupt program to determine if a start sending signal has been received. In the case of dial pulse outpulsing, when the start sending signal is received, the interrupt program moves the digit indicated by the outgoing digit count into the outgoing pulse count area of the originating register and then increments the outgoing digit count by 1. The program also sends a peripheral decoder order to operate relays in the sender to start sending dial pulses. Every 100 milliseconds the wired logic decrements the outgoing pulse count by 1 until that count reaches 1. At that point it sets the sending flag so that the interrupt program may inform the sending circuit to stop sending pulses.

After turning off the dial pulses the interrupt program loads a constant into the outgoing pulse count to time the interval between digits so that the sending flag will be raised when it is time to send the next digit. The sequencing of functions required to time digit and interdigit intervals is controlled by the sending function identified in the third word of the originating register. Since both the timing of the dial pulses and the timing of the originating register scans are under control of the processor, it is possible to guarantee that the starting and stopping of dial pulse digits occurs during the off-hook portion of the dial pulse cycle so that no short or long pulses occur.

In the case of multifrequency sending, the entire digit is sent to the sending circuit by the interrupt program and the wired logic facility is only used to time the length of the digit. All multifrequency digits except the key pulse digit are sent in synchronism for all originating registers and are turned on for 75 milliseconds and then off for 75 milliseconds. (The key pulse digit, the first digit sent, must remain on for 100 milliseconds.) While sending, the interrupt program accesses the originating register every 150 milliseconds, reads the next digit and sends peripheral decoder orders to the sender to turn on the pair of tones representing the digit. The number of the sender is then placed in a special list. 75 milliseconds later this special list is examined by the interrupt program and peripheral decoder orders are sent to all senders in the list to turn off any multifrequency tones they are send-

* A peripheral decoder is a shift register circuit used to receive serial messages from the processor and operate relays in trunk and service circuits. (See Ref. 3 for details.)

ing. The timing for multifrequency digit turn off is shown in Table III.

In order to send the proper number of digits, the program, before it sends a new digit, compares the stop sending code and the outgoing digit count to see if it has reached the last digit. If not it checks to see if another digit has been received as indicated by the incoming digit count. This procedure is used to permit overlap outpulsing. When the last digit has been sent the ready flag is set to indicate that fact to the transient call record.

For program simplification, the stop sending code is used throughout the processing of dialing by the progress mark routines to indicate the total number of digits expected. This arrangement permits the interpretation of digits to be handled independently from any prefixing digits that may have been received, since the position of the digits to be interpreted is determined relative to the stop sending code rather than from an absolute position within the originating register.

5.3 *Peripheral Order Programs*

Network connections are set up and torn down by means of peripheral order buffer programs which execute during the timed interrupt. Using information recorded in 16 word buffers, these programs send orders to network controllers, peripheral decoders, and scanners in order to perform actions needed by base level call and maintenance programs. Because a network controller requires 40 milliseconds to execute an order from the central processor, peripheral order buffers are examined every 50 milliseconds by the peripheral order buffer execution program. This execution frequency guarantees that all network controllers are idle at the beginning of the peripheral order buffer execution program and makes efficient use of them.

The sixteen call store words provided for a peripheral order buffer allow capacity for storing information (scan points, peripheral decoder address, paths) for up to three circuits. The most typical peripheral order buffer action disconnects a terminal from one circuit and connects it to another. The majority of these actions can be handled by one peripheral order buffer. Actions involving more than three terminals require the use of more than one peripheral order buffer. Other actions provided by the peripheral order buffer execution programs include:

(*i*) Simple disconnects: Trunk, service circuit, or junctor circuit cut through contacts are opened and the line ferrods of any lines involved in the call are restored by closing the line cut-off ferreeds.

(*ii*) Simple connects: A network connection is set up between two terminals and circuits associated with one or both terminals are placed in an initial state.

An action taken by the peripheral order buffer execution program is accomplished by executing a sequence of generalized tasks which include sending a network order, sending a peripheral decoder order, scanning a ferrod, or delaying before another task is attempted. Use of the orders may be illustrated by the specific orders for the action which connects dial tone to a line. (See Ref. 3 for details of circuit operation). The orders are:

(*i*) Perform a false cross or ground check and then connect the digit receiver terminal to one terminal of the specified wire junctor.

(*ii*) Perform a false cross or ground check and connect the other terminal of the wire junctor to the line, opening the line ferrod cut off contacts (delay 50 milliseconds).

(*iii*) Perform a power cross test on the line (delay 50 milliseconds).

(*iv*) Operate the cut through contacts in the digit receiver by means of a peripheral decoder order to connect the digit receiver to the line.

(*v*) Delay 50 milliseconds (1 peripheral order buffer cycle) to allow relay operating transients to decay.

(*vi*) Scan the digit receiver line supervisory ferrod to check by means of an off-hook transition that there is network continuity.

(*vii*) Place the digit receiver relays in the dial tone state.

The peripheral order buffer execution programs actually execute other tasks than those illustrated, but the main body of the peripheral order buffer actions are composed of the tasks illustrated.

Two methods are used by the program to execute the 300 different peripheral order buffer actions currently required. Most actions do not occur often so a method has been devised to allow a small program to handle these actions at the expense of some system real time. These slower actions are described by data words stored sequentially in memory which are decoded one at a time by the program to determine the sequential tasks needed to complete each peripheral order buffer action. The sequence needed for a particular action is defined by storing an index to the sequence table in the peripheral order buffer when it is selected and initialized. The sequence is terminated by an end code.

In addition to sequence tables, trunk state tables are used by the peripheral order buffer programs to determine which trunk relays must be operated to place the circuit in a desired state. These tables also indicate how ferrods are assigned for a trunk, and describe to the peripheral order buffer execution program which ferrods to scan for continuity checks, and other scan tasks. When the peripheral order buffer is initialized, a trunk state table pointer is stored in the peripheral order buffer for each trunk. The state table is then indexed from the starting location defined by the pointer to determine how the circuit relays must be operated to reach a particular trunk state. Use of the circuit state table allows only one task sequence to be used to describe a given peripheral order buffer action for all trunk types. This technique also yields good flexibility for the incorporation of new trunk types or unforeseen peripheral order buffer actions.

A second method is used for performing frequently needed peripheral order buffer actions. This method uses direct program code rather than sequence tables and is used primarily for network actions needed for setting up intraoffice, incoming, and outgoing calls. Also included in this category are all simple disconnects and actions which may be completed exclusively by means of peripheral decoder orders. By saving the time required to address and decode the contents of words in the sequence table, the real time consumed by these frequent actions may be minimized. Combination of the two methods of executing peripheral order buffer actions permits compromises between system real time and program size.

A second body of routines called the peripheral order buffer loading program is used to handle peripheral actions. This program is a collection of subroutines which interfaces with base level call and maintenance programs to load the information into a peripheral order buffer which is needed by that buffer's execution program. When called, the loading program selects an idle peripheral order buffer and, because the transient call record rarely contains information which is directly usable, performs translations upon trunk scan points and other information recorded in the transient call record to obtain peripheral assignment information. The terminal memory records used by a transient call record are also addressed to retrieve path information which must also be interpreted before it can be used by the peripheral order buffer execution program.

When a base level transient call record program requires a peripheral order buffer action, the type of action as well as the lines, trunks, and service circuits to be used in the action must be specified to the loading routine. A calling sequence is used which consists of the instruction which transfers control to the loading program entry point, one or more data words, and a failure and a success return. The data words of the calling sequence are decoded by the loading program to determine what action is requested.

Several important features are provided by the peripheral order buffer execution programs which have not previously been mentioned. These include blocking unwanted supervisory reports caused by network and circuit actions, minimizing open intervals on the transmission paths of established calls and retrying failing network orders.

Supervisory signals are generated by setting up and tearing down network paths which are important for checking network continuity but which serve no other useful purpose. An example of such a signal is the off-hook which results from connecting a line to a circuit junctor or a trunk. The line is known to have previously been off hook and an off-hook report adds no new information. If the scanning programs are allowed to report the off-hook, however, approximately 1 millisecond of real time is required to discard it. For this reason the peripheral order buffer execution program changes the state of the last look bit kept for the ferrod in call store when a continuity check reveals that it has changed state. The scanning program, which executes after the peripheral order buffer execution program, detects no change of state.

Open intervals in the transmission path are caused by peripheral order buffer disconnect-connect actions after a connection has been established. When such an action involves a line, an advance network controller reservation technique is used to limit the open interval to 150 milliseconds. Since the connect action consists of setting up two half paths in the network, a half path is first connected for the circuit to which a line is being connected. The network controller needed to connect the line is then reserved by marking its status bit busy in memory. The cut through relay of the circuit from which the line is being disconnected is opened and the execution is terminated until the next peripheral order buffer cycle.

During the next peripheral order buffer execution cycle (50 milliseconds later) the reserved network is used to connect the line to the new circuit and 50 milliseconds later the cut through relay of this second circuit is closed by a peripheral decoder order. Failure to find an idle network controller for use in connecting the line causes the peripheral order buffer execution to delay opening the line's transmission path until an idle controller becomes available.

The peripheral order buffer execution program works in conjunction with a "working mode program" which executes just prior to the peripheral order buffer execution program. By scanning the F, S, and T scan points of the network controllers, the working mode program is able to detect if any network controllers failed network orders during the previous peripheral order buffer execution cycle. When a failure indication is found, it is the function of the working mode program to record the failure and cause all orders sent during the previous peripheral order buffer execution cycle to be retried. A second failure causes the failing peripheral order buffer to be located by means of a special peripheral order buffer execution cycle in which orders sent during the previous cycle are formed and examined to determine which order used the failing controller. Location of the failing order and the peripheral order buffer from which it was sent allows the order to be examined for correctness and triggers maintenance actions which attempt to find a system configuration which can communicate the failing order to the network controllers. A series of up to three retries can occur and success at any stage allows the peripheral order buffer to continue its sequential tasks.

5.4 *Translation*

Translation data occupies a large percentage of program storage in all No. 2 ESS offices. With less than one module (16,384 words) required in the smallest offices, storage for translation information must be capable of growing to more than eight modules (131,072 words) for larger offices. Efficient use of translation storage is necessary to achieve low system cost.

The simplest and fastest translator would be one which uses the number to be translated (terminal equipment number, directory number, trunk scan point number, and so on) to directly index storage to obtain the desired data. This approach is wasteful of storage in almost all cases. In a small office (for example, 4000 lines) only $\frac{1}{5}$ to $\frac{1}{4}$ of a maximum sized translator would be filled and rarely would it be completely filled.

To achieve translation storage efficiency and ease of growth in No. 2 ESS, the major translators have been divided into blocks of translation words, called subtranslators. A portion of the number being translated is used to address an entry in a master table index and retrieve the starting address of the desired subtranslator. The remaining portion of the number is used to index to the desired entry in the subtranslator. The word located by this procedure may be either the output data or a pointer to the expanded data for the more unusual cases.

In No. 2 ESS even parity is used on data words and odd parity is used for program instructions. Encoding parity in this way allows the processor to detect and prevent erroneous use of translation words as instructions or instructions as translation data. When a parity error is detected, a duplicate processor is switched on line in an attempt to continue service. Should the error occur in translation data, it may cause two or more processor switches and require system emergency action to recover its processing capability.* A primary cause of this type of error is improper accessing of storage resulting from the use of incorrect translator indexes.

Protection against such errors is built into most of the translation subroutines. To implement this protection, additional information describing each subtranslator is included in the master table index. These data describe both the number of words per subtranslator entry and the maximum index value for the subtranslator. Each use of the translator can then be checked to insure that the index used to access the subtranslator remains within its boundaries. Errors detected by this check cause a teletypewriter printout and cause error indications to be returned to the calling program.

Another potential source of errors in translation data is service order changes which must be introduced into the system by the telephone company. Errors introduced into certain areas of the translation information such as the master table index can be catastrophic; for this reason, no facility for altering such storage is provided via teletypewriter. These critical items may be changed when necessary through use of the office data assembler program to recompile translations.[6]

Several strategies are invoked in an attempt to guarantee the integrity of service order information typed into the machine for lines and trunks. One example of these strategies forces service orders for line changes to include both the equipment number and directory number of the line. The redundancy in the input data allows simple but powerful checks to be made before using the change data. Insuring that service order information is independent of program store addresses and internal storage formats also helps to minimize errors in input information.

5.4.1 *Originating Translation*

The line origination translation provides a conversion from the line equipment number to the line class, service data and billing number. Each line in the No. 2 ESS office has a terminal equipment number which is used to refer to that line during the processing of its calls. The class data derived from the line origination translation includes

* System emergency actions are program actions which attempt to eliminate the call or calls causing system errors.

a major class indicating the type of service such as individual, two-party, PBX, and so on; a screening class which is used to determine routing and charging for calls dialed by this line; and various class bits which indicate the type of dialing and the priority of service. This translation is used when a digit receiver is selected, when a billing entry is prepared for automatic message accounting, when various special service requests occur, and when special control functions are required.

The first step in the translation shown in Fig. 6 involves the use of the high six bits of the equipment number to select the terminal equipment number translator associated with the network on which the line appears. This translator has a one-word entry for each terminal equipment number, and the low nine bits of the number indicates this entry. In a majority of cases, this translator contains an abbreviated class indication for the line and the line's billing number (directory number). The five-bit abbreviated class represents the most common combinations of service features in the office. It is expanded in another translator to indicate all of the class of service information including the major class, the screening class and the various class bits.

A sizable number of lines, however, cannot be represented by an abbreviated class because they require more information than a single word can contain. In these cases, the single word associated with the line equipment number contains a pointer used to locate two or four words of data which further expand the class information for that line. These individual class of service expansions are laid out in the same format as the abbreviated class expansions to assure that the resultant output is independent of the translation method. The presence of the pointer is indicated by an all zero abbreviated class.

Two party lines are an example of a situation where additional data words are required. Each party has a separate class and billing number which must be listed as a separate entry. Each entry is then treated as though it came from the terminal equipment number translator. Another example of the variation from the normal pattern is the PBX line where the class of service applies to the entire group of PBX lines. In this case, the pointer indicates to which PBX group (PBX number) the line belongs and its member number within the group. As a result, only one specification of class information is needed for the entire PBX group. The grouping is also used by the machine to mark a PBX line busy as soon as it originates to prevent it from being selected for a call to the PBX.

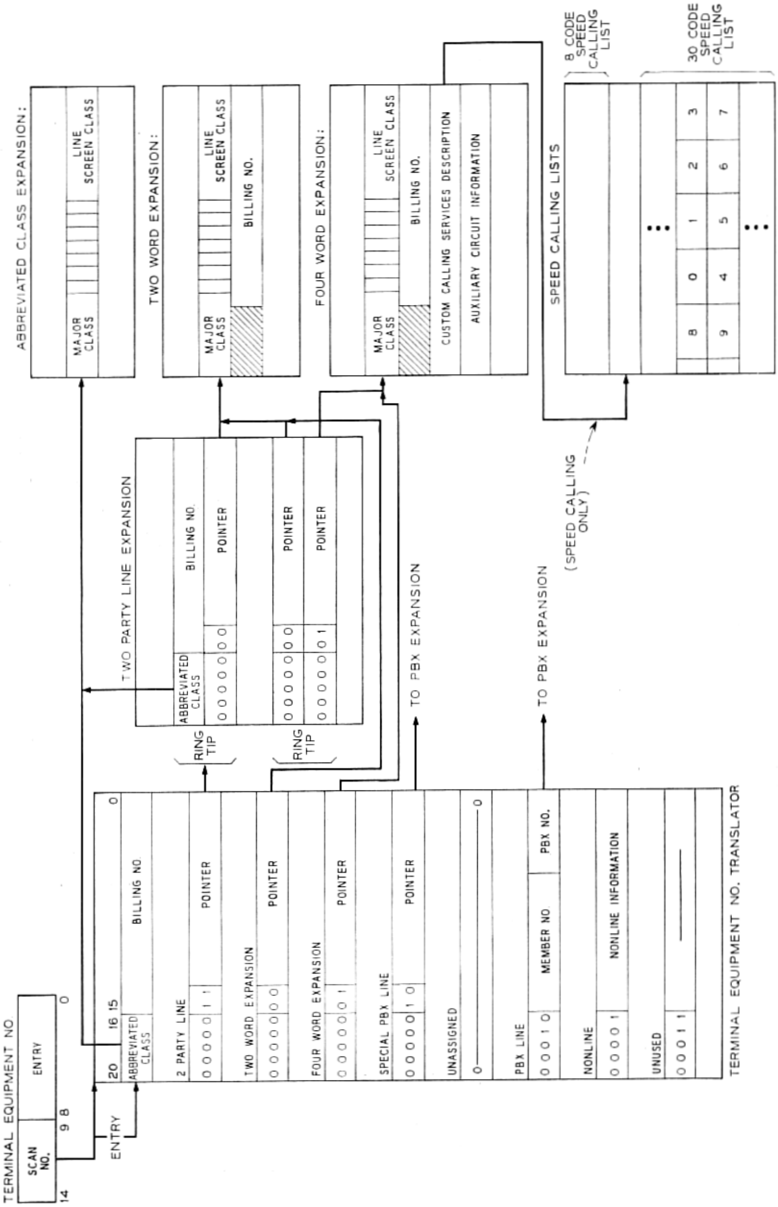


Fig. 6 — Originating translation.

5.4.2 *Three-Digit Translation*

The function of three-digit translation is to convert the dialed digits and the originating customer's class of service (obtained from the originating translation) to applicable routing and charging data. The translation must decide if a call is permissible, and how to complete and charge it. The translation is complicated by the varying charges on coin, toll, and message register operations, and the discrimination required for extended area service and wide area telephone service (WATS).

The three-digit translation could be implemented using a program store table for each three-digit code with an entry in every table for each class of service. Such an arrangement, however, would require several hundred thousand program words which is clearly uneconomical. The objective of this translator is to provide the same amount of information using 32 screening tables each with 32 entries, amounting to a total of 1,024 program store words. Several stages of compression of the input data are necessary to permit the use of such small screening facilities.

The three digits dialed by a customer are converted into a ten-bit binary number. As indicated by Fig. 7 the high nine bits select one of 400 words in the code point translator and the low bit selects one of two entries within the word. The resulting eight-bit number defines an entry in the code group expansion table. Many three-digit codes dialed by the customer can be treated identically and, therefore, they use the same expansion.

There are several types of code group expansion entries. The first is used when it is desired to determine the amount of traffic for a specific office code. This preroute peg count entry contains the location of the counter and a pointer to another code group expansion entry which is used for routing the call. The second is a conflict entry which provides two pointers to other code group expansions, one is used if the three digits are determined to be an area code; the other is used if the three digits are determined to be an office code. This facility will eventually be required when the same three digits are used for both office and area codes.

A third type of entry in the code group expansion points to a foreign area translator which provides for further translation of the fourth, fifth, and sixth digits after a customer dials an area code of a nearby number plan area. This foreign area translator facility permits direct routing to specific office codes in the foreign numbering plan area. As

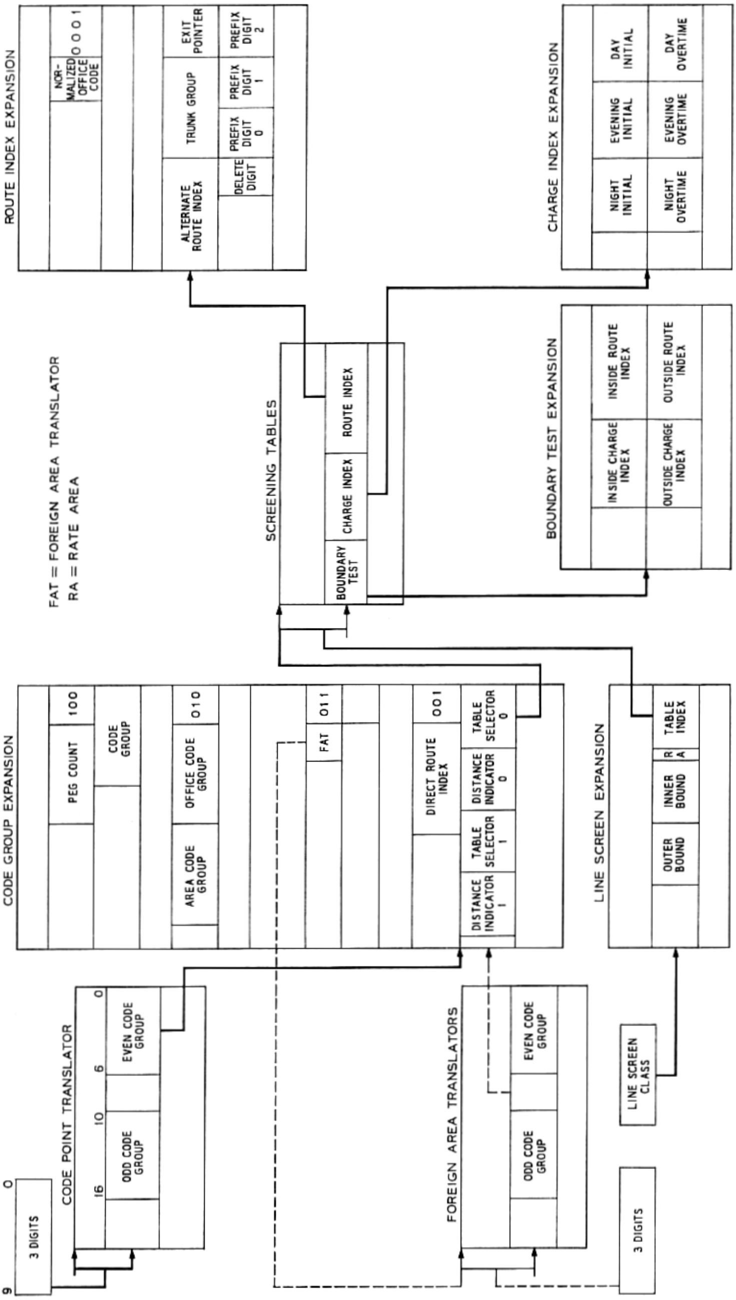


Fig. 7 — Three digit translation tables.

indicated, the foreign area translators have exactly the same format and output as the code point translators and are indexed with the fourth, fifth, and sixth digits dialed by the customer.

The final and most prevalent type of entry is one which indicates the screening table that should be used for this call. If the charge treatment for different three-digit codes is the same but they are routed over different trunk groups, a different direct route index is provided in each of the code group expansion entries, but they point to the same screening table. If the screening process results in a route index equal to zero, then the direct route index is used.

Since it is fairly common for a central office to serve two different rate areas, two screening table selectors have been provided in the code group expansion. The selector to be used in a particular call is determined by the rate area bit in the line screening class expansion for the originating line. Notice that two direct route indexes are not required because regardless of the rate area involved, the direct route to the office having a particular three-digit code should be the same. Also provided for each rate area in this code group expansion is a distance indicator.

The line screen expansion has an entry for each line screen class, and each entry represents a different call treatment. The line screen class is obtained from the originating translation and it is independent of the calling lines major class. Only one type of entry is used in the line screen expansion. It contains an index to the screening table which was previously selected by the code group expansion. It also contains the rate area bit indicating the rate area of the line along with an inner and outer boundary specifier.

The screening table is selected by the code group expansion and the entry is selected by the table index from the line screening expansion. There is only one type of entry in these tables, and each contains a route index and charge index which are to be used for routing the call and providing charging information. If the route index found is zero then the direct route index specified by the code group expansion is used. Also contained in this entry is a pointer to the boundary test expansion to be used for this call.

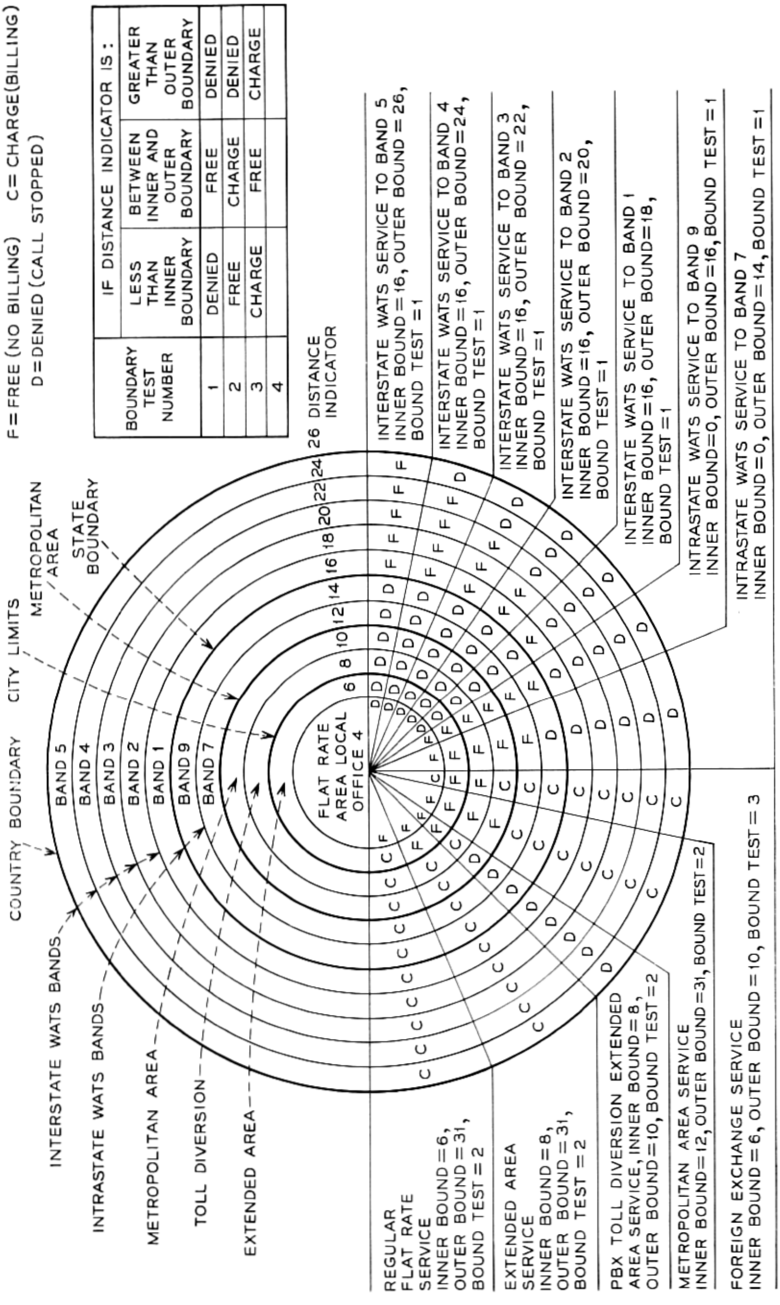
The route index expansion provides one type of entry which specifies the trunk group that should be used for completing the call, and it provides an alternate route index if the trunk group specified is all busy. In addition it indicates through an exit pointer how many digits are to be expected along with any modification necessary in the dialed information, such as prefixing and deleting digits. A second type of entry is provided for intraoffice calls. It contains a normalized office code for later use by the directory number translator, and the three digits of the actual office code.

The charge index expansion is used to determine the appropriate rate to be used in charging for the call. Entries in this table include those for coin and message register calls. Figure 7 shows a typical entry.

A boundary test expansion has been included in this three-digit translator to provide additional screening capability without using a large number of additional program store words. This feature takes advantage of the fact that special charge treatments such as extended area service, and WATS do not require different routing patterns but only different charging techniques and that the details of the charging are determined by the automatic message accounting center during the processing of automatic message accounting tapes and not at the time the call is placed. It also takes advantage of the fact that there is a good correlation between the treatment of various three-digit codes and their distance from the central office, and that a particular customer class of service need have only two boundaries where the method of charging for a call changes.

The boundary test results are obtained by making an arithmetic comparison between the distance indicator obtained from the code group expansion and the inner and outer boundaries obtained from the line screen expansion (Fig. 7). If the distance indicator falls between the inner and outer boundaries, the route and charge index as specified in the screening table are used. Otherwise, the route and charge index as provided by the indicated entry in the boundary test expansion are used. The entry in the boundary test expansion is determined by the boundary test pointer in the screening table. Within that entry, the inside route and charge indexes are used if the distance indicator is less than the inner boundary while the outside route and charge indexes are used if the distance indicator is greater than the outer boundary. If either the route or charge index resulting from the boundary test expansion is zero, then the route or charge index specified in the screening table is used.

An example of the compression of information gained from use of the boundary test is shown in Fig. 8. The upper portion of the circles defines an assignment of distance indicators to 11 code groups. The lower portion of the figure defines 12 line screening classes and assigns inner and outer bounds and boundary test indexes to each of them.



BOUNDARY TEST NUMBER	IF DISTANCE INDICATOR IS: LESS THAN INNER BOUNDARY	BETWEEN INNER AND OUTER BOUNDARY	GREATER THAN OUTER BOUNDARY
1	DENIED	FREE	DENIED
2	FREE	CHARGE	DENIED
3	CHARGE	FREE	CHARGE
4			

Fig. 8 — Boundary test diagram.

If no other screening classes were provided for the central office illustrated in the example, then only one screening table would be required and only three entries within that table would be needed. Without the boundary test facility, this example would require 11 screening tables with 12 entries each or a total of 132 words.

5.4.3 *Directory Number Translation*

Translation of directory numbers is the process of associating either a terminal number and terminating class or an error treatment indicator with each directory number which can be received from a line or trunk. For most lines in an office a single 21-bit word is sufficient, but lines subscribing to the call waiting, call forwarding, or the series completion feature require more than one word to contain all terminating line data. For them, a two-word expansion is needed and a pointer is used in the terminating translator to locate the expansion. Formats for entries in the terminating translator are shown in Fig. 9.

To afford reasonable breakage and ease of directory number assignment, the directory number translator has been subdivided into 100 word blocks called terminating translators. A terminating translator is indexed with the binary equivalent of the tens and units digits of the directory number to obtain the desired entry. For each office

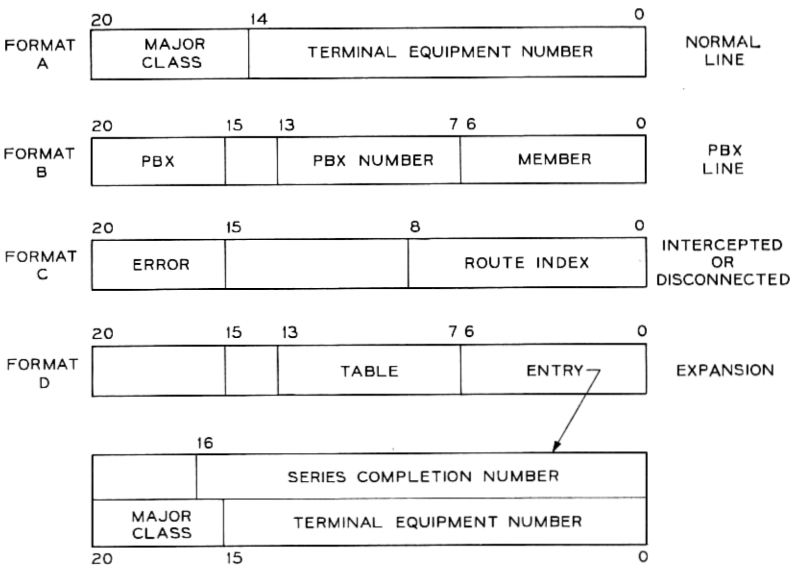


Fig. 9—Terminating translator entry formats.

code used, 100 of these tables may be provided. Access to the terminating translators is gained through an index called a normalized office code which is obtained from the three digit translation for intraoffice calls or from trunk translation information for incoming calls. For each normalized office code there is an entry in the master table index which contains the address of a number group translator. The number group translator contains 100 entries, each of which is a pointer to a terminating translator. Indexing the number group translator with the binary equivalent of the thousands and hundreds digits of the directory number locates the entry which contains the location of the corresponding terminating translator. The entire directory number translator is depicted in Fig. 10.

5.4.4 *Trunk and Service Circuit Translations*

Trunk and service circuit translation subroutines are used to obtain peripheral circuit assignment information. Outputs from the translator consist of the terminal memory record address, terminal equipment number, trunk group and member number, peripheral decoder enable address, and scan point numbers.

Although both scan point numbers and trunk group numbers are used as inputs to the trunk and service circuit translations, only the use of scan points is described here.

The manner in which storage is accessed in translating a universal trunk frame scan point number is indicated in Fig. 11. The depicted subtranslator contains two words for each of the 128 circuits on a universal trunk frame bay and two additional words used to derive the terminal memory record address and peripheral decoder enable for these circuits. The uniformity of universal circuit scan point and peripheral decoder assignments allows simplicity of the translator and compression of the information stored.

There is no pattern to the manner in which the scan points and peripheral decoders are assigned to miscellaneous trunks and service circuits. Two additional words describing these assignments must, therefore, be stored for each miscellaneous circuit to allow these extra degrees of freedom. A subtranslator accessed for micellaneous scan points is shown in Fig. 12.

VI. DETAILS OF AN INTRAOFFICE CALL

A better understanding of how the call processing functions are controlled to provide telephone service may be obtained from a detailed description of an intraoffice call.

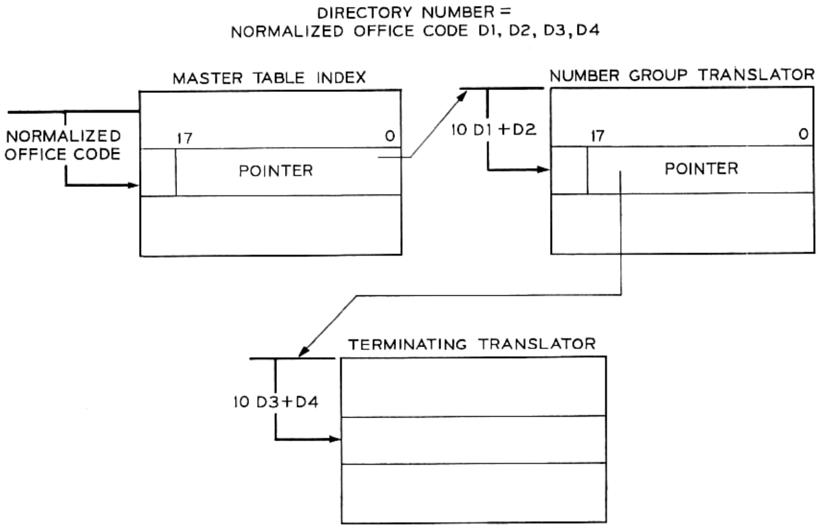


Fig. 10 — Directory number translator.

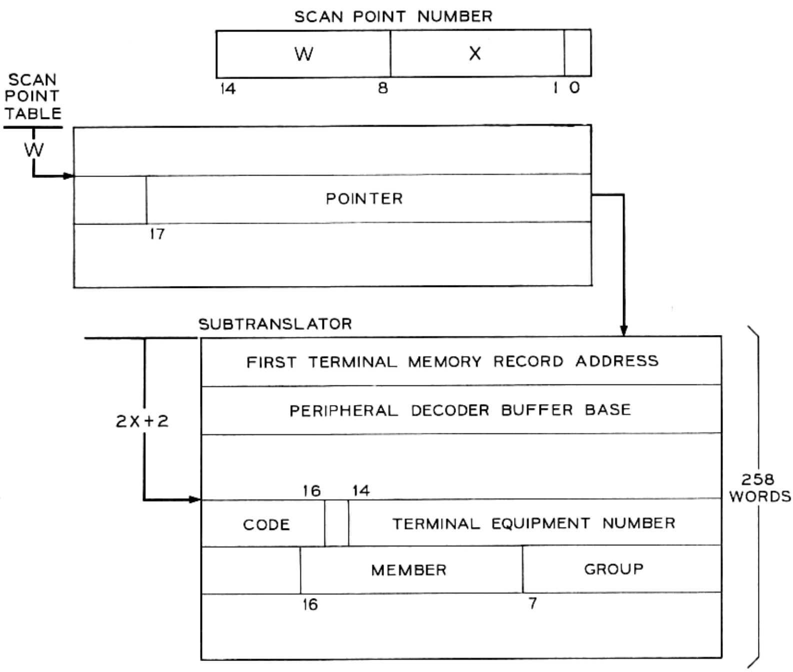


Fig. 11 — Universal trunk and junctor translator.

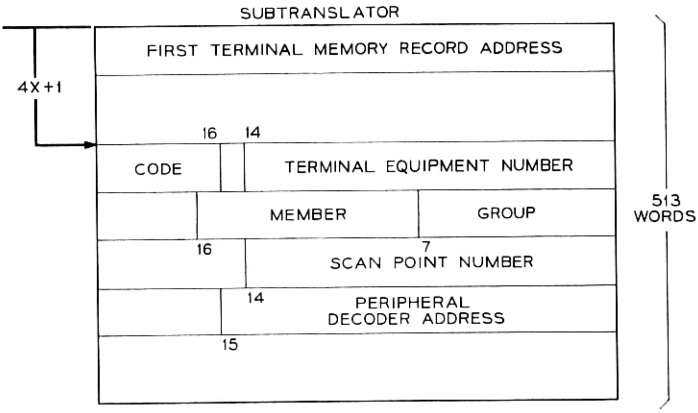


Fig. 12 — Miscellaneous scan point subtranslator for 128 circuits.

When a subscriber goes off hook the ferrod associated with his line is saturated. This fact is detected by the line scanner, and the interrupt program performs hit timing before placing the line terminal equipment number in the originating hopper (see Section 5.1.1). Later the supervision distributor detects the nonzero entry during its processing of the hopper, and searches for an idle transient call record. When one is found, the terminal number of the line is written into the second word of the transient call record, and the line origination progress mark is recorded in the first word. The line terminal number is then cleared from the origination hopper. If an idle transient call record is not available, processing of the origination hopper is terminated and the line number is allowed to remain in the hopper to await an idle transient call record.

At the conclusion of the supervision distributor program's execution, the call processing scan is begun. During this scan the transient call record containing the originating line is encountered and control is passed by means of the progress mark to the line origination program. This program's first action is to obtain a translation of the line number to determine the originating treatment for the line. An idle originating register must be found to receive digit information from the customer, and its address is recorded in the fifth word of the transient call record (Fig. 2). The line originating class data is then examined to determine whether a dial pulse or combined receiver is needed to receive digits from the line. The circuit group number of the desired receiver is supplied to a subroutine which selects a digit receiver of the type required by the customer.

The selection subroutine is responsible for locating an idle digit receiver and reserving a network path for use in connecting the line to the receiver. When an idle receiver is found, it is made busy in call store and the circuit group number is used by the service circuit translation programs to obtain its terminal equipment number. The network map contained in call store is then examined, and an idle path from the calling line to the receiver is selected. The links corresponding to the idle path are made busy in the network map and information describing the path is placed into word 1 of the receiver terminal memory record (see Fig. 4). At this time the terminal memory record is also placed into the transient state, and the address of the transient call record is written into the first word of the terminal memory record. A ten-bit circuit number is placed into the sixth word of the transient call record to identify the selected receiver. In the event that the selection program is unable to locate a path to the selected receiver, the receiver status bit is made idle and a second receiver is selected. A path selection is then attempted for the second receiver. If no path is found on the second try the failure is communicated to the line origination program.

All equipment identities needed to connect dial tone to the calling line are now stored in the transient call record. To achieve the connection a subroutine is called which selects a peripheral order buffer and loads it with all circuit and path information required by the peripheral order buffer execution program. The loading program translates the digit receiver ten-bit number contained in the sixth word of the transient call record to obtain the peripheral assignments for the circuit. No further actions can be taken for the call until the connection is made and so the peripheral order buffer loading program stores a new progress mark in the transient call record to await completion of the connection.

This progress mark causes a program to check during each base level transient call record scan if dial tone has been given. It also provides for queueing in the event that all peripheral order buffers were found busy by the peripheral order buffer loading program.

The final action of the dial tone peripheral order buffer is to record the scan points of the digit receiver in the originating register so that digit collection may be initiated by the wired logic digit receiving circuits. This action must be done after dial tone is returned in order to insure that relay operating transients are not mistaken for dial pulses. Because as much as 100 milliseconds may elapse before the base level program becomes aware of the successful connection, the scan point numbers must be recorded in the originating register during the interrupt so that dial pulses will not be missed. The successful connection is reported to the transient call record by the peripheral order buffer loading program, and the progress mark is changed to address the digit reception program.

The digit reception program is written in a general way so that it may be used to collect digits for all types of calls and still allow good flexibility in the interpretation of the digits. Once each 500 milliseconds this program examines the ready bit (Fig. 5) contained in the originating register addressed by the fifth word of the transient call record. A zero in this bit indicates that no new digits have been received and tests are then made to determine if partial dial time-out has occurred. If the ready bit is set to 1, a new digit has been received. A comparison is then made between the incoming digit count contained in the fourth word of the originating register and a digit number stored in the fifth word of the transient call record. If the incoming digit count is found to be less than the digit number, the ready bit is zeroed and no further action is taken. If the incoming digit count is equal to or greater than the digit number, control is transferred to the address contained in the eighth word of the transient call record. A user of the digit reception program may then initialize the digit number to a number of digits which need to be collected before translation is necessary and designate the location where control is expected when that number of digits have been received. An additional feature of this program allows the user to specify by means of a bit in the transient call record whether the digit reception program should dispose of dialing time-out indications in a normal manner or whether the user program should be informed of a time out.

During the course of a call, the first incoming digit is examined as soon as it has been dialed for the purpose of providing the 1 and 0 prefixing features. Of course an initial 0 can also signal an operator call. For lines subscribing to custom calling features, two digits must also be examined so that requests for the service may be detected. The digit reception program normally must wait for three digits, however, before the type of call being placed can be discovered. A subroutine is used to translate the first three dialed digits, and the returns from this subroutine inform the digit reception program of the type of call being placed. An intraoffice call indication from the translation subroutine causes the digit reception program to await the completion of seven digits before the called number is fully determined. For this case the translation subroutine supplies a charge index and an indicator called a normalized office code which designates that one of a maximum of six office codes has been dialed. Both the charge index and normalized office code are recorded in the seventh word of the transient call record.

Reception of the seventh digit causes the digit reception program to call the directory number translation program to decode the full number (see Section 5.4.3). The normalized office code is supplied to the directory number translation program and is used to index the proper number group translator. The translation returns a terminal equipment number and a called party terminating major class to the digit reception program. The terminal equipment number is used to locate the scanner last look bit to determine if the called party is busy. This bit has a zero value when the line is idle and assuming this to be the case, the last look bit is set 1 to busy the line.

A talking path is necessary for the final line-to-line connection, and a path hunt subroutine is used to search the call store network map for this path. In order to guarantee a low probability of blocking on all connections, the identity of the A link used for the line side of the digit receiver connection is passed to the path hunt program. In searching for a talking path the subroutine attempts to reuse this A link. It should be observed that if this tactic is not employed, at most three A links are available for use in the talking connection. This condition, of course, lowers the probability that a path can be successfully found.

The path hunt subroutine selects a circuit junctor for use in connecting the two lines because the lines must be supervised at the junctor during the talking interval. When a path has been found, the terminal memory record of the circuit junctor is made transient and path information is recorded in the second word of the terminal memory record. The identity of the A link reserved for the called line side of the talking connection is returned so that it can be shared with the path used for ringing the called party.

Selection of a ringer includes an attempt to give immediate ringing to the called party. By consulting information recorded in the call store by the ringing and tone frame maintenance program, the ringer group is selected which will ring the station soonest. When a path is found for use in ringing the called line, it is recorded in the ringer terminal memory record and this record is placed in the transient state.

Calls which require timing during the talking interval (message register, local coin) must be identified in a call store list called a stable call timing list during that interval. Before proceeding to ring the called line an idle two-word entry called a stable timing entry is located and reserved for the call. Failure to locate the stable timing entry when it is required causes the call to be intercepted.

Automatic message accounting calls are recorded by means of a nine-track magnetic tape machine using a triple entry format. If the call requires automatic message accounting billing the initial entry is formed and placed in a call store buffer for later transmittal to the tape machine.

Ringing is supplied to the called party by execution of a peripheral order buffer loading program which selects a peripheral order buffer and loads information in it necessary for connecting the ringer. Successful execution of this routine causes another peripheral order buffer to be loaded to disconnect the digit receiver and connect the calling line to the circuit junctor selected for the final talking path. Audible ringing is then returned from the junctor circuit to the calling party. The base level call program awaits completion of each of these peripheral order buffer actions by executing a call to the peripheral order buffer loading program during each transient call record scan as described previously. Completion of the second peripheral order buffer action causes the originating register, the digit receiver, its associated terminal memory record and path memory used for the dialing connection to be idled in memory. The ringing progress mark is then stored in the first word of the transient call record to await further customer action.

In order to conserve system real time, the ringing program executes only one instruction during each transient call record scan while awaiting a response to ringing. This instruction simply skips past the ringing transient call record and causes the following transient call record to be processed. As described in Section 5.1.2, the supervisory entry point into the ringing program is taken when the supervision distributor reports supervision to the transient call record. When this program branch is reached, bits stored in the transient call record by the supervision distributor are decoded to determine the source of the change in supervision (calling or called party) and the new supervisory state.

Indication of an off-hook condition of the line supervisory ferrod in the ringer circuit causes the program to call the peripheral order buffer loading program to select and load a peripheral order buffer with information necessary to disconnect the called party from the ringer and connect him to the circuit junctor. The peripheral order buffer execution program also changes the state of the junctor circuit relays so that audible ringing is removed from the calling party and the two parties are connected in a talking state.

Successful execution of the peripheral order buffer actions allows the ringer circuit, and the ringing path to be idled in memory. A new progress mark is then stored in the first word of the transient call record causing a one-second period to be timed as a check for valid answer. During the one-second period on-hook reports from the called party are ignored so that on-hooks generated by switch hook bounce will not be erroneously interpreted. A disconnect received from the calling line, of course, terminates the call.

After the called party has remained off-hook for one second, charge guard timing is begun by storing a new progress mark in the first word of the transient call record. This timing period extends for two seconds and is provided for insuring that the connection is truly established before a charge is made for the call. An on-hook report received from the called party during the "charge guard" interval initiates timing as if the called party had terminated on an established call.

Upon completion of the two-second charge guard timing interval, charging is begun. For automatic message accounting calls, an answer entry is formed and placed in the automatic message accounting buffer. Coin or message register charge timing is initiated by addressing the previously selected stable timing entry and changing its format so that the stable timing entry can start timing of the initial charge interval. The transient call record is then cleared and the circuit junctor terminal memory record is used to record the identities of the calling and called lines. Bit 15 of the first word of the terminal memory record is also set equal to 0 to indicate that the call is stable. No further action is required for the call until one of the lines takes further action or further charging action becomes necessary because of time out of the stable timing entry.

To terminate the call one of the parties must hang up. The on-hook condition causes the junctor circuit ferrod used to supervise the on-hook line to indicate the on-hook transition when it is next interrogated by the trunk and junctor scanning program. After 50 milliseconds of hit protection timing, this program reports the on-hook scan point number to an on-hook timing list which is routinely examined by the supervision distributor. The on-hook report is allowed to remain in the on-hook timing list for 250 milliseconds before further action is taken, in order to guarantee the on-hook condition.

During a subsequent execution of the supervision distributor, examination of the on-hook timing list indicates that time out has occurred. The terminal memory record corresponding to the circuit junctor is then accessed via a translation of the scan point number and discovered to be stable. This terminal memory record state causes the supervision distributor to select an idle transient call record for handling tear down of the call. The calling and called line numbers are recorded in the second and third words of the transient call record and a disconnect progress mark is written into the first word. Additional information is placed in the transient call record to allow determination of whether the calling or called party terminated. The circuit junctor terminal memory record is then made transient by the supervision distributor.

During the call processing scan the disconnect program is accessed via the progress mark and the on-hook party is determined. Should the calling line be the first to disconnect, he is disconnected from the circuit junctor by means of a peripheral order buffer action and the called party is retained in the transient call record awaiting his on-hook signal. Receipt of the calling party disconnect also causes the following actions:

(*i*) An automatic message accounting disconnect entry is recorded in the automatic message accounting buffer if required.
(*ii*) Coin or message register timing is terminated and the associated stable timing entry is cleared if employed.
(*iii*) A coin collect is performed on the calling line if applicable.
(*iv*) The calling line last look (status) bit is set to 0.

The ensuing hang up by the called party results in another peripheral order buffer action to disconnect him from the circuit junctor. The circuit junctor, its associated terminal memory record, the path, and the line last look bit are then idled in memory and the transient call record is cleared. When the called party hangs up prior to calling party disconnect, actions similar to the above are taken, but an 11 second period is timed prior to release of the connection. A calling party disconnect during the time out period causes the call to be terminated without further timing.

VII. TESTING

Call processing programs, automatic progression tests and manual progression tests are all used to detect trunk and service circuit abnormalities. The diagnostic programs take advantage of the many functions that are common to those required by the normal call processing routines. In this way large program and call store economies are realized. For example a transient call record is used to control the test sequence. This permits the call store area used by the diagnostic program to be released when the test is completed. It also allows the test programs to use call processing routines for selecting circuits, sending peripheral orders, and outpulsing.

7.1 *Trunk and Service Circuit Testing*

Trunk and service circuit maintenance facilities in the No. 2 ESS have three main objectives: (*i*) to automatically detect faulty circuits and remove them from service as soon as possible after a trouble condition arises, (*ii*) to provide a teletypewriter message indicating the action taken and a trouble number to pinpoint the trouble, and (*iii*) to provide facilities to aid in manual testing and repair.

The main phases of trunk and service circuit maintenance (detection, diagnostic, and repair) are shown in Fig. 13. Trouble detection may result from automatic or manual detection tests or troubles may occur during call processing.

In the normal processing of calls, checks are made at key points for abnormalities that may exist. Timing is performed at places where supervision, such as a start pulsing signal, is expected from another office. Continuity and foreign potential checks are made whenever a new network path is set up. Checks are built into every scanner and peripheral decoder order. Whenever one of these checks detects a trouble condition that could have been caused by a trunk or service circuit, diagnostic programs are called in to test the suspected circuits.

Diagnostic tests for trunks and service circuits are designed to insure that the circuit can carry out its normal functions. The circuit being tested is connected through the network to other service or test circuits and a sequence of "test steps" is performed which produces pass or fail results. Data obtained from the test is used to produce a trouble number which consists of the "test step" and the results of the test, such as ferrod responses. This approach permits one number to serve both as the dictionary number for the trouble locating manual and as raw data.

Outgoing trunks are tested by placing a test call to a "test line" in the far end office. Tones and supervisory signals are transmitted by the test line as a part of the test sequence and to indicate the results of tests made on the trunk at the far end. A tone detector is connected

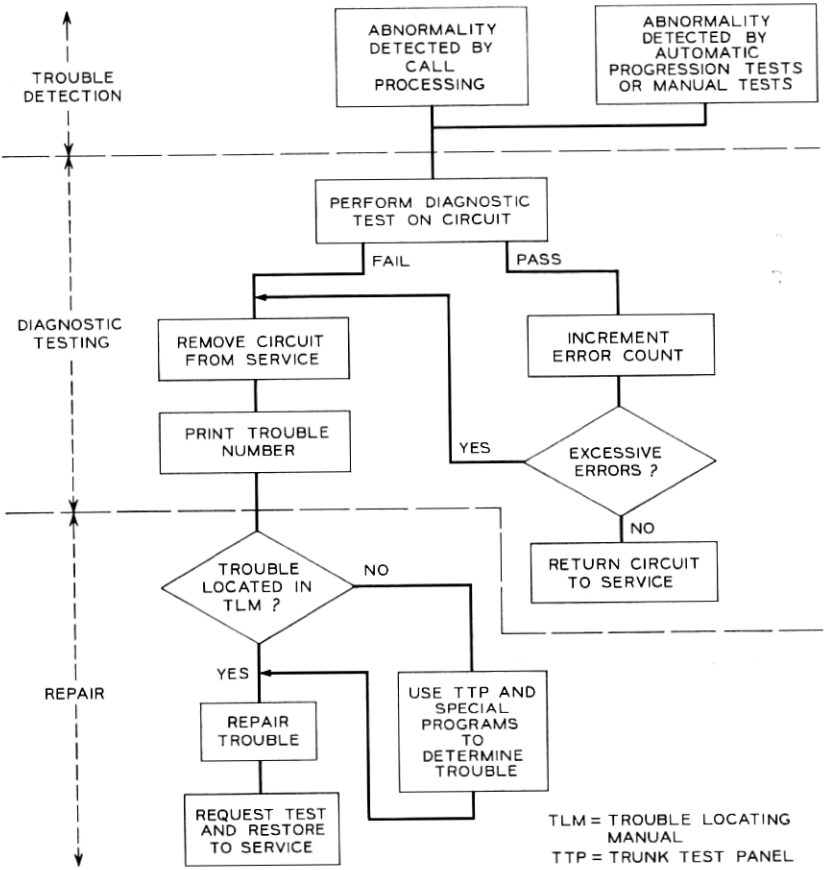


Fig. 13 — Trunk and service circuit maintenance plan.

to the near end of the trunk to detect these tones and busy or reorder conditions. The test program monitors the tone detector and the trunk supervisory ferrods for these responses and compares the results to "all tests pass" data that are stored in memory.

The philosophy of testing service circuits differs somewhat from that of trunk testing. Since trunk tests involve other central offices, testing techniques are dictated by the test facilities available in those offices. However, service circuits are contained wholly within one office and complete freedom exists in the design of such tests.

Two basic approaches are used in service circuit testing. One method is to connect the circuit to a specially designed test circuit through the switching network. Marginal signals are generated by the test circuit to insure that the service circuit can function properly under any condition that may be encountered in normal service. Both circuits are sequenced through a series of states, and their ferrods are scanned in each test state. The results are then matched against "all tests pass" data stored in memory. This approach is used to test customer dial pulse receiver circuits where the test circuit simulates marginal customer dial pulses that might be transmitted over "worst case" customer lines.

The other method used in testing service circuits is to use the complement of the circuit being tested to generate the input signals. Multifrequency transmitters and receivers are tested in this manner by selecting a transmitter (receiver) at random to test a particular receiver (transmitter). A test circuit is connected into the network path between the two circuits to introduce transmission loss and distortion. Over a period of time all combinations of transmitters and receivers are tested as pairs and nonworking or marginal circuits are detected.

If the tests described above were carried out in response to errors detected by call processing programs, then a test failure results in the circuit being removed from service; an "all tests pass" is treated in the manner described in Section 7.2. If the tests were carried out in response to automatic progression testing or manual testing then a failure on the first pass through the tests is treated as a trouble detection and the test is repeated. The results of the second try are treated in the same manner as a failure from a call processing test request.

Safeguards are included to prevent the automatic removal of too many circuits in a group. Two successive failures in an automatic progression test causes a test of the test circuit. The number of circuits that can be automatically removed from service is limited to a small percentage of that group.

A manual test of a trunk or service circuit is initiated via the maintenance teletypewriter or the trunk test panel. Teletypewriter requests may specify a particular circuit, group, or all circuits to be tested. A diagnostic test is always performed on a circuit when the craftsman attempts to restore it to service.

7.2 *Error Analysis*

When a trouble condition is encountered by call processing programs and the subsequent test of the circuit yields an "all tests pass," an error is recorded for that circuit. Errors may be caused by overload conditions in the far end office, dirty relay contacts, transient noise, or by a marginal condition that is not detected by the diagnostic program. The general scheme of error analysis is to detect circuits that are more "error prone" than other circuits of the same type. Error counts are kept on a limited number of circuits at a time. The error count is compared with an "excessive error count" which depends upon the total errors accumulated by all the other circuits in the same "error analysis group." An excessive number of errors results in the circuit being removed from service and a teletypewriter message is printed to inform the craftsman. If a circuit does not accumulate enough errors to be removed from service, it is replaced in the error analysis list by another circuit that is suspected of high error rate.

7.3 *Growth Testing*

Errors in the installation of new circuits or in the translation data associated with that circuit not only result in the circuit not working but may also affect other circuits. This seriously degrades service and may cause symptoms of troubles in other parts of the system such as scanners or the central pulse distributor.

Special programs have been provided which check the translation data associated with a circuit and perform an "installation test" of the circuit. The regular diagnostic program is used for testing service circuits; trunks are tested with the outside loop specially terminated to permit testing before assignment to a regular trunk group and to isolate circuit troubles from those that might be caused by outside facilities.

7.4 *Line Testing*

In No. 2 ESS, line testing facilities are concentrated at the local test cabinet and the local test desk. The cabinet is located at the central office while the desk may be at some remote centralized location. Continuity, leakage, foreign potential, and ferrod tests may be made from either the desk or cabinet. Permanent signal testing will be, as far as possible, under the control of the desk. The trunk test panel may also be used to test lines when necessary, although it is intended to be used primarily for trunk and service circuit testing.

Automatic line insulation tests, which check for excessive leakage and foreign potential, are conducted periodically on all idle lines in the office. Test failures are reported to a teletypewriter at the local test desk. This test may also be requested by the craftsman via the local maintenance teletypewriter or the local test desk teletypewriters.

A line test facility that can be used by an installer from the customer's premises or by a craftsman at the local test desk is the station ringer test circuit. It is used to test for station ringer ground and to test *Touch-Tone®* telephone dials. Tones from the test circuit indicate proper operation or failure.

VIII. CONCLUSION

No. 2 ESS provides the switching functions needed by a modern central office by means of an electronic data processor under the control of a set of call processing programs. This paper enumerates the switching function provided and discusses the organization and structure of the call processing programs used to minimize the program size. It describes in detail the primary processing functions and gives an example of an intraoffice call to illustrate the use of these functions to provide telephone service and to provide circuit testing capability.

The large variety of telephone services offered by No. 2 ESS is provided by a program of modest size. The system is expected to be economically competitive in the small and medium size central office field, and the program structure has made a major contribution to this competitive position.

REFERENCES

1. Beuscher, H. J., Fessler, G. E., Huffman, D. W., Kennedy, P. J., and Nussbaum, E., "Administration and Maintenance Plan," B.S.T.J., this issue, pp. 2765–2815.
2. Browne, T. E., Quinn, T. M., Toy, W. N., and Yates, J. E., "Control Unit System," B.S.T.J., this issue, pp. 2619–2668.
3. Digrindakis, J., Freimanis, L., Hofmann, H. R., and Taylor, R. G., "Peripheral System," B.S.T.J., this issue, pp. 2669–2712.
4. Lonnquist, C. W., Manganello, J. C., Skinner, R. S., Skubiak, M. T., and Wadsworth, D. J., "Apparatus and Equipment," B.S.T.J., this issue, pp. 2817–2863.
5. Spencer, A. E., and Vigilante, F. S., "System Organization and Objectives," B.S.T.J., this issue, pp. 2607–2618.
6. Barton, M. E., Haller, N. M., and Ricker, G. W., "Service Programs," B.S.T.J., this issue, pp. 2865–2894.