THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 62, No. 6, July-August 1983
Printed in U.S.A.

Modernization of the Suburban ESS:

Hosting the No. 10A Remote Switching System

By D. W. BROWN,* J. J. DRISCOLL,* F. M. LAX,* M. W. SAAD,* and J. G. WHITEMYER*

(Manuscript received August 31, 1982)

The No. 10A Remote Switching System (RSS) has brought a new dimension to telephone switching in the rural area. The capability to host a 10A RSS was first made available on the metropolitan switches, the No. 1 and No. 1A ESS. The capability has now been extended to the suburban switch, the No. 2B ESS. This article describes the additions and modifications made to the No. 2B ESS to allow it to host the 10A RSS. The discussion also covers the administrative and maintenance features provided and their associated host software implementation.

I. INTRODUCTION

1.1 Definition and basic description

A local Electronic Switching System (ESS) provides the call control for a 10A Remote Switching System (RSS). The 10A RSS acts as a slave executing orders sent to it from the host ESS and reports events, such as line originations, to the host. A major advantage of this type of distributed control is that the complex tasks of call processing can use existing host software and share host equipment and trunking facilities. This sharing of host ESS software makes it possible to easily

* Bell Laboratories.

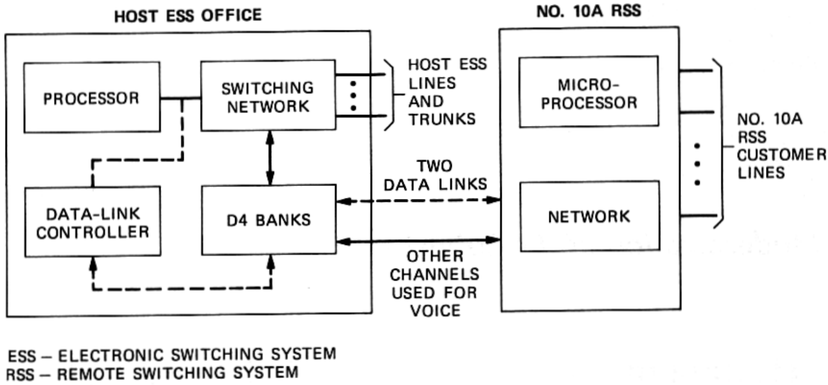


Fig. 1—RSS configuration using digital carrier.

provide RSS lines with the sophisticated features that are offered to host ESS lines.

The major components required for 10A RSS operation include a host ESS, one or more 10A RSS frames, a data-link controller, interconnecting voice channels, and data links. Figure 1 shows this configuration for an application using digital connectivity between the host ESS and a 10A RSS (remote terminal). At this time, the host function has been developed for the Western Electric No. 1 ESS, No. 1A ESS, and No. 2B ESS machines. The data links are used for communication between the host ESS and the 10A RSS and provide the means by which orders from the host are transmitted to the 10A RSS and acknowledgments are returned to the host. Voice channels are used to provide the RSS lines with access to the host network and are selected dynamically. The data link employed for the No. 2B ESS-RSS communication function is a new design utilizing an intelligent Serial Peripheral Unit Controller Data Link (SPUC/DL).* The 10A RSS data-link communication makes use of portions of the X.25 protocol.

The 10A RSS basic frame can serve up to 1024 lines. A companion frame may be added to allow up to 2048 lines to be served by a single RSS entity. The design is such that the basic element of growth can be as small as eight lines. Voice and control communications between the remote and the host can be made over either digital or analog carrier facilities and the range may be as great as 280 miles, depending primarily on the type of transmission facility.

In the event of total carrier system or data-link outage, the 10A

* Acronyms and abbreviations used in this paper are defined at the back of this Journal.

RSS is arranged to automatically transfer to a stand-alone mode of operation, which provides basic telephone service between stations connected to that RSS unit. In the stand-alone mode, special provisions can be made to handle emergency types of traffic such as "911". Details regarding stand-alone operation may be found in an earlier BSTJ article, "Remote Terminal Firmware," by D. A. Anderson et al. in the April 1982 issue on the 10A RSS.

All major units of the 10A RSS are duplicated, and a continuous dialogue is exchanged between the host and remote site concerning the overall health of the system. The basic system philosophy is that the remote unit is a complete slave to the host and merely reports events to the host; the RSS then receives a stream of explicit orders from the host concerning that event. All maintenance procedures must be controlled from the host. Initiation of diagnostics and other functions may be further remoted to a Switching Control Center (SCC).

II. CALL PROCESSING

2.1 Data communications

2.1.1 Hardware overview

Figure 2 shows the 10A RSS—No. 2B ESS host interface and the

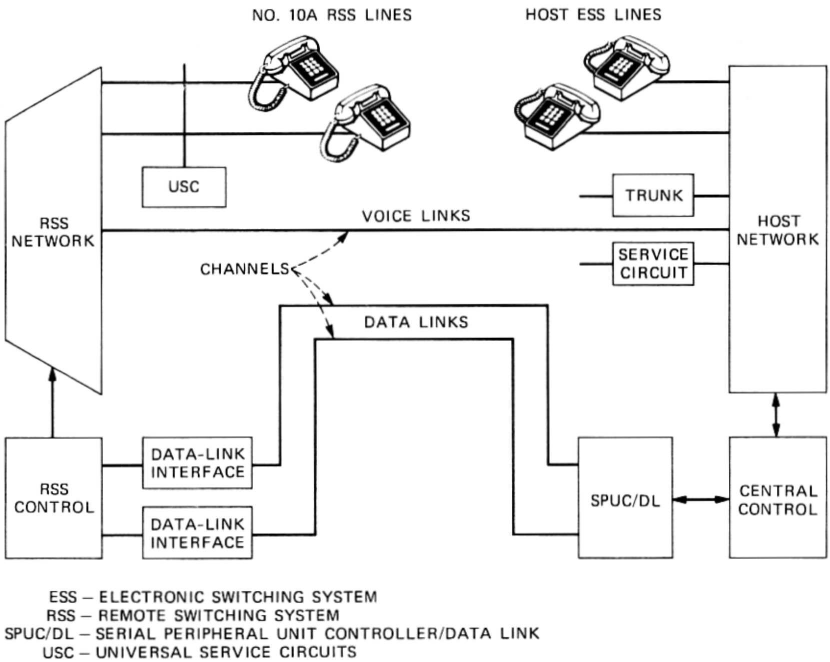


Fig. 2—10A RSS—No. 2B ESS host interface.

hardware components involved in the transmission of data between the remote terminal and the host. Communication between the two machines takes place over a pair of low-speed data links that share the same transmission facilities as the voice channels that interconnect the remote terminal with the host. Each data link is placed on a separate transmission facility for reliability. Where carrier facilities are used, the links are assigned to a dedicated voice channel on the carrier system, with each link being assigned to a separate carrier terminal.

Both RSS links are 2400 bps synchronous links. The on-line link is active and carries the entire data traffic between the host and 10A RSS, while the off-line link is maintained in a standby state as a spare. The on-line, off-line status of the links is determined by the host office and is based on error information accumulated by the software responsible for driving the links. At the remote terminal the link is interfaced to the RSS microprocessor through a Data-Link Interface (DLI) circuit. The DLI provides a small amount of data buffering and performs a number of control functions essential to implementing the synchronous link protocol.

At the host, the link interfaces with a functionally similar line interface unit that is part of the SPUC/DL. The function of the SPUC/DL is to provide the control for physically transmitting and receiving data on the links and to provide data buffering for the host office. The data being transmitted and received by the SPUC/DL are buffered on a per-link basis within the terminal. Sufficient data buffering is provided to allow the host to efficiently exchange large blocks of data with the terminal on a schedule that is efficient to the host.

2.1.2 Software overview

The routines that control the data transmission between the remote terminal and host are located in the remote terminal, the SPUC/DL, and the host office. There are two basic functions to be performed: data must be transferred reliably over the link and an interprocess communication system must be provided to allow software processes in the host to communicate with processes in the remote terminal. These two functions are provided by the data-link protocol software and a set of message-routing routines. The protocol provides virtually error-free transmission of data over the link by executing a set of error-detection and error-correction procedures. The message routines allow a process in one machine to direct a message to a process in the other. These two systems are largely independent and bear a hierarchical relationship to one another in the sense that the message routing routines rely on the link protocol routines to accurately transmit data from one end of the link to the other.

2.1.3 Data-link protocol

The protocol routines are executed in the remote terminal and the SPUC/DL. The 10A RSS application uses the link-level portion of the X.25 protocol to control the link. It is a bit-oriented protocol designed for synchronous link operation. To provide for error detection and correction, the data to be transmitted are segmented into numbered blocks termed frames. As frames are transmitted they are sequentially numbered and a cyclic check code is computed over the data in the frame. The frame numbering scheme makes it possible to identify frames for retransmission and to detect missing frames in the received data. As frames are processed at the receiving end of the link, the cyclic check code is recomputed and compared to the one transmitted with the frame. A mismatch indicates that a transmission error has occurred. A positive acknowledgment is returned to the data transmitter for all frames received correctly, and a retransmission request is returned if a frame is received in error.

The protocol software at the SPUC/DL has the additional function of providing link status reports to the host machine. Error conditions such as high transmission error rate, frame acknowledgment time-outs, and loss of data carrier are monitored by the protocol and reported to the host. The transmission error rate is determined from the number of retransmission commands received and sent by the SPUC/DL protocol program. From this data the host data-link state control can take action to remove a link from service if it becomes inoperative or if its throughput is restricted because of excessive data errors.

2.1.4 Message-routing routines

The routines that are responsible for routing data between individual processes in the two machines are executed by the remote terminal and the host processor. These programs assume that data received from the link protocol programs are error free and that any additional error control procedures for detecting transmission errors are unnecessary. These programs are designed to transmit data between buffers associated with client programs in the two machines. A program having data to transmit will load the data into its associated buffer. The buffer will be activated for the message-routing routines and the data will then be transferred to a buffer associated with the destination program in the other machine. The message format is shown in Fig. 3.

2.1.5 Host data transmission

The buffering technique adopted for the transmission of peripheral orders to the remote terminal was designed to be compatible with the structure of the existing host software. The order buffering and mes-

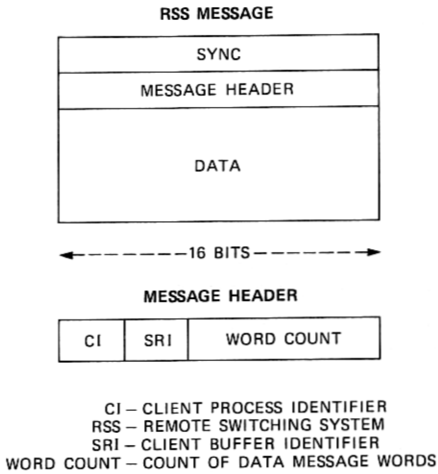


Fig. 3—RSS message structure.

sage transmission must be tailored to the host software structure if the existing call-processing and maintenance routines are to be preserved. A few remarks on the nature of the call-processing programs are necessary to understand the interface requirements.

The host call-processing programs are divided into call segments that process an input from a subscriber or a peripheral circuit to completion in one real-time segment. All the peripheral orders required to process the input are generated by the separate set of input/output (I/O) programs. Peripheral operations in the host or remote terminal take tens or hundreds of milliseconds to execute, which precludes their direct execution within the call segment. During execution, a typical program segment may generate several remote terminal peripheral messages that are destined for different RSSs. In most cases it will also generate a number of orders to be executed in the host periphery in conjunction with the remote terminal actions. The execution routines must be able to coordinate the transmission of the remote terminal orders to the different RSSs. Frequently, it is necessary to execute the remote terminal-host peripheral actions in a predefined sequence where either the host or the remote terminal action must occur first.

The call-processing and maintenance programs are coupled to the I/O programs through a set of I/O buffers that are loaded with the peripheral orders to be executed. All host peripheral orders are buffered in Peripheral Order Buffers (POBs), where they are executed by the POB execution programs designed to handle the timing requirements presented by the host periphery.

A set of Remote Order Buffers (ROBs) in the host machine buffer the peripheral orders being transmitted to the remote terminals. They are loaded and administered by the call-processing programs in the same manner as the peripheral order buffers. The orders to be sent to the remote terminal are loaded in the ROB via a set of order macros that provide a high-level interface with call processing. When order loading is complete, the ROB is activated and the I/O routines transmit the orders to the remote terminal. An individual ROB may be used to send orders to any RSS. The host can have any number of ROBs pending to send orders to an RSS; however, each remote terminal has a fixed set of eight ROB records that are used to store orders received from a ROB at the host. The ROB records are buffers associated with the peripheral order execution program in the remote terminal that executes the orders transmitted from a ROB.

2.1.6 ROB execution protocol

A rudimentary protocol has been established to coordinate the activities of the call-processing routines at the host with the execution of ROB orders at the remote terminal. Several orders will be grouped together into a single message at the host to be transmitted to the remote terminal. The orders in the message are executed at the remote terminal and upon completion an acknowledgment is returned to the host. The acknowledgment will indicate whether all the orders in the message were successfully executed and must be received by the host before any further actions will be permitted on this call. If the remote terminal encounters a failure in executing an order, the acknowledgment message will specify the failed order to the host and the remote terminal will suspend execution of all remaining orders in the ROB record.

It is essential to receive a positive confirmation on the status of the orders for several reasons. If an order failure occurs, the host fault-recovery routines can be scheduled to clear the call from the system. In addition, the acknowledgment allows the host to correctly sequence any other peripheral actions with the remote terminal orders. When the acknowledgment is received, the host can activate an associated POB or return to a call-processing program to implement the next action on the call. The restriction that additional RSS orders will not be transmitted until the acknowledgment is received also prevents the host from transmitting multiple sets of orders for the same call that would be executed in an arbitrary sequence at the remote terminal.

2.2 RSS network associated data

The processing of RSS calls requires the allocation and management of resources that are physically located at the 10A RSS or shared between the host ESS and the 10A RSS. These resources include the channels that interconnect the host and 10A RSS, receiver off-hook

(ROH) tone circuits located at the 10A RSS, and the 10A RSS network crosspoints. Also, the status of RSS lines is maintained at the host. Several factors were considered in deciding whether to place these functions in the 10A RSS or in the host ESS. These factors are:

1. The effect on service because of the additional time delay if the 10A RSS has to be interrogated to determine line status and to hunt voice channels and network paths.

2. The additional software development required if the host ESS programs have to take a real-time break to interrogate the 10A RSS to obtain a line's status. Host software is structured around data that can be accessed without taking a real-time break.

3. The duplication of development effort that is required to provide the same functions in several host ESS systems.

Factor 3 indicates that the overall development effort would be reduced by placing the line, channel, and 10A RSS network path administration in the 10A RSS. However, the service criteria and the effect on the existing host software were judged to be more important. Therefore, these functions are allocated to the host ESS.

The overall development effort is reduced by making the program and data structure designs independent of the host ESS. Thus, they are highly portable between ESS machines. This section describes the data structures required to administer the RSS resources.

2.2.1 Data structures

Data structures are required in the host ESS memory (call store) to record the status of the RSS facilities and to provide for their administration. To simplify the engineering of the office, these data structures are provided in sizes that correspond to the three basic network sizes of the 10A RSS. Each of these structures is described below.

2.2.1.1 Network block. One network block is provided for every 10A RSS. Its size is fixed regardless of RSS equipage. Among the subblocks contained in the network block are the network map, the channel status blocks, and the remote miscellaneous scan-point status map.

Scan points are provided at the 10A RSS for uses such as alarms, make-busy keys, and stop-hunt keys. The remote unit periodically scans these scan points and, via the data link, reports any changes to the host ESS, which updates the bits in its map to indicate the present state of the scan point.

2.2.1.2 Path memory remote record. A Path Memory Remote (PMR) record is provided for each possible line and channel network appearance. Since the 10A RSS network can be equipped in three different sizes, considerable ESS memory is saved by also providing PMRs in block sizes corresponding to the network size. PMRs contain information about the state of the terminal and a pointer that is used to point to another memory block (call register or path memory for junctor) involved in the call.

2.2.1.3 Path memory for junctor record. A Path Memory for Junctor (PMJ) record is a block of call store that is associated with a junctor in the 10A RSS network. It is used to store path and terminal information when the junctor is in a network path. A PMJ also contains a state and pointer, which are used to link to another PMJ or to a call register. A PMJ is provided for each equipped junctor. Blocks of PMJs are allocated based on the RSS network size.

2.3 Call-processing strategy

The RSS host call-processing software provides an ESS central office with the capability to supply ESS features to lines served by the remote switching system. Since most of the call-processing functions for RSS lines are performed by the host ESS office, a full family of ESS features can be provided to the remote subscribers. The RSS call-processing software resident in the host ESS provides the means of controlling a remotely located switching system by taking advantage of existing equipment and control capability in the ESS. Firmware in the remote terminal supplements the host call-processing software appropriately. All call-processing control resides in the host ESS and any required actions at the RSS are requested via data-link messages to the RSS. This permits the host to exercise total call control.

2.3.1 Originating call

A line originating in the RSS is first recognized during line scanning performed by the RSS microprocessor. The RSS line-scanning program in the remote terminal recognizes the line off-hook, performs hit timing, and sends an origination request data-link message to the host ESS. If service is allowed, the host marks the RSS line busy and hunts an idle voice channel between the RSS and ESS. It also hunts a path through the RSS network from the originating line to the selected voice channel, and selects a customer digit receiver in the host along with a host network path from the voice channel to the receiver. A Remote Order Buffer (ROB) is then executed to send appropriate data-link messages to the remote terminal to set up the RSS network path and set the line supervision mode to repeat supervision of the originating line over the channel in the dialing (fast repeat) mode. To minimize the dial-tone delay interval, when the above ROB is initiated the host executes orders to its periphery (via a POB mechanism) to set up the host network path between the voice channel and receiver to provide dial tone. That is, the ROB and POB are executed in parallel.

Processing of the call from this time on proceeds basically the same way as an origination by a host line. Digits are collected and analyzed by the same host software used to process host calls. At the completion of dialing and digit collection, a data-link message is sent from the host to the remote terminal to set the line supervision mode to repeat the supervision of the originating line over the channel in the talking (slow repeat) mode. This mode conserves remote terminal microprocessor real-time capacity.

The RSS originating call, from this point on, is routed and completed normally (excluding terminations to RSS lines) just as non-RSS line origination processing. This originating call configuration is depicted in Fig. 4. Upon answer by the called party or completion of outpulsing, the talking connection is established from the voice channel through the host network. RSS answer timing, billing, traffic, and other administrative functions are all applied and performed by the host just as for non-RSS calls. If the call terminates to an RSS line, special terminating RSS functions are performed, as discussed in the following sections. When either the calling or called parties disconnect, disconnect functions are performed, as discussed in Section 2.3.5.

2.3.2 Terminating call

An RSS terminating call is recognized when the host ESS performs the called number [terminating Directory Number (DN)] translation on digits collected from an originating host line or trunk. RSS lines

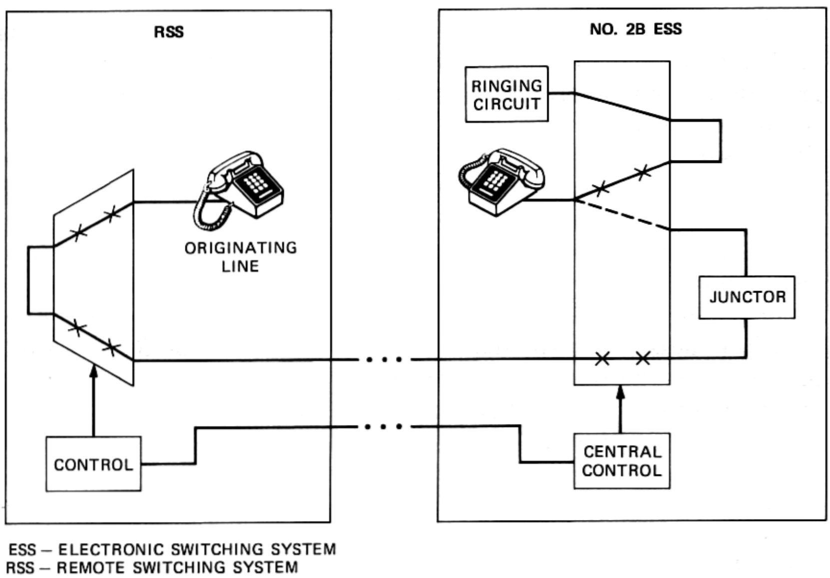


Fig. 4—RSS originating call.

are distinguished from host lines by special RSS indicators in the terminating line translation output. After the translation is completed, special actions, as with the RSS originating call, are required to set up ringing. The host hunts an idle voice channel to the RSS, hunts a path in the RSS network between the voice channel and the terminating line, and seizes an idle host ringer and audible service circuit with associated host network paths to the voice channel and originating line or trunk, respectively. In addition, a talking path through the ESS network (between the voice channel and the originating line or trunk) is reserved.

A ROB is activated to send data-link messages to the remote terminal to connect the terminating line to the voice channel and apply ringing to the line. Upon receipt of the data-link orders, the remote terminal selects an idle universal service circuit along with a metallic bus and time slot to provide the type of ringing specified in the data-link message. Supervision of the line is transferred across the voice channel to the host in the fast repeat mode.

Upon successful execution of the ROB data-link orders in the remote terminal, the host executes a POB to set up paths in the host network from the voice channel and the originating line or trunk to its associated service circuit. Power cross and low-line resistance tests are done on the voice channel from the host ringing circuit. The host ringing circuit is then left in a state to monitor ring trip sent by the remote terminal over the voice channel to the host. Actual ringing is applied to the line by the Universal Service Circuit (USC) at the remote terminal; the ESS host ringing circuit does not apply ringing voltage to the voice channel, but is only used to monitor for ring trip. This call configuration, as depicted in Fig. 5, maximizes use of the existing terminating call sequences in the host.

When the called party answers, the remote terminal automatically releases and idles the ringing facilities (USC, metallic access bus, and time slot), relays the ring trip report (off-hook signal) of the line across the voice channel, and sets the supervisory mode to slow repeat.

The host ESS detects answer over the voice channel at the ringing service circuit, tears down the ringing and audible circuit connections in the host, and sets up the talking path that was previously reserved between the voice channel and the originating line or trunk. If the originating line in the RSS terminating call description is actually another voice channel to the same RSS as the terminating line, the call is considered an intra-RSS call and special actions are invoked as described in Section 2.3.4.

Disconnect actions are identical to those for the RSS originating call except for the disconnect timing associated with the terminating versus the originating party.

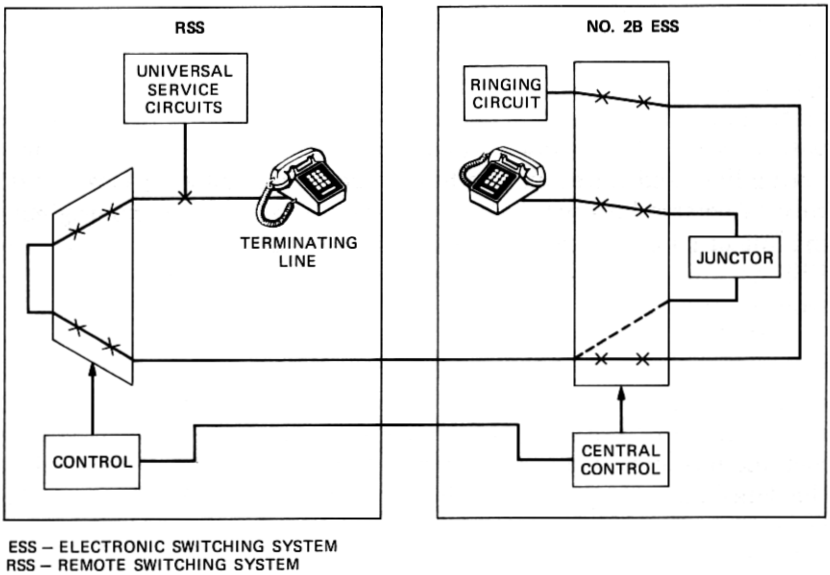


Fig. 5—RSS terminating call.

2.3.3 *RSS reverting calls*

RSS reverting calls involving a call between two parties on a party line are handled in a manner similar to host reverting calls. However, ringing is provided in a way similar to how it is applied on RSS terminating calls with the exception that two time slots are needed in the RSS remote terminal so that ringing can be applied to both customers. This RSS ringing option, which can be either ac/dc or superimposed, is completely independent of the host ESS office ringing option, or any other RSS served by the same host. The RSS universal service circuit has the capability to provide either ringing option under firmware control.

2.3.4 *Intra-RSS call*

An intra-RSS call, where both the originating and terminating parties are served by the same RSS, is handled initially as a combination of an RSS originating call and an RSS terminating call. In the No. 1 ESS and No. 1A ESS applications, after answer, the host initially establishes a talking path within its network between the two voice channels. Immediately following the establishment of this talking connection, certain RSS actions are invoked to reswitch-down the call so that the talking connection resides entirely within the RSS network. This releases the ESS network path and voice channels for use on other calls. In the No. 2B ESS application, when answer is received and both parties are served by the same RSS, the call is reswitched-down immediately without first establishing a talking connection through the host. This sequence will not result in a momentary open interval after the initial talking connection has been established.

The reswitch-down action is initiated when the host hunts an RSS network path within the RSS between the originating and terminating lines. A ROB is activated to send data-link orders to the remote terminal to disconnect both line-to-channel network paths and connect the two lines through the RSS network. The supervisory mode of the RSS lines is set to scan for either a disconnect or switchhook flash, depending on the features associated with each line. Since the intra-RSS connection is entirely within the RSS, a change in supervisory state of the line must be reported over the data link to the host. The sequence of intra-RSS call configurations including reswitch-down is illustrated in Fig. 6.

If a network path in the RSS is not available or if one of the lines is an RSS coin line, the intra-RSS call is not reswitched-down and is connected through the host ESS network. Intra-RSS calls involving coin lines are not reswitched-down in order to utilize host coin-disconnect routines and thus simplify disconnect actions.

The use of various custom calling services or other special services requires a reswitch-up of an intra-RSS call to establish a talking path between the two parties via the host network using two voice channels. This allows existing host software and equipment to be utilized to provide these customer services. The following operations require a reswitch-up operation on an intra-RSS call:

1. A flash by an RSS customer to add on a third party
2. A terminating call to one of the two parties of an intra-RSS call that has the call waiting/terminating feature
3. A busy verification test by an operator of one of the two parties of an intra-RSS call.

When the host determines that a reswitch-up function must be performed, for any of the reasons given above, the host seizes two idle voice channels to the RSS and hunts a path between them in the host network; the host also hunts a path between the two voice channels and both lines in the remote terminal. A POB is executed in the host to set up a talking connection between the two voice channels followed by a ROB to send data-link messages to the remote terminal to disconnect the intra-RSS talking connection, connect each line to a voice channel, and set the supervision state of each line to the talking mode (repeat supervision of the line over its respective voice channel). Once the intra-RSS call is reswitched-up to a talking connection via the host ESS network, the original service requested can be provided just as it would to a normal line-to-line connection of two host lines.

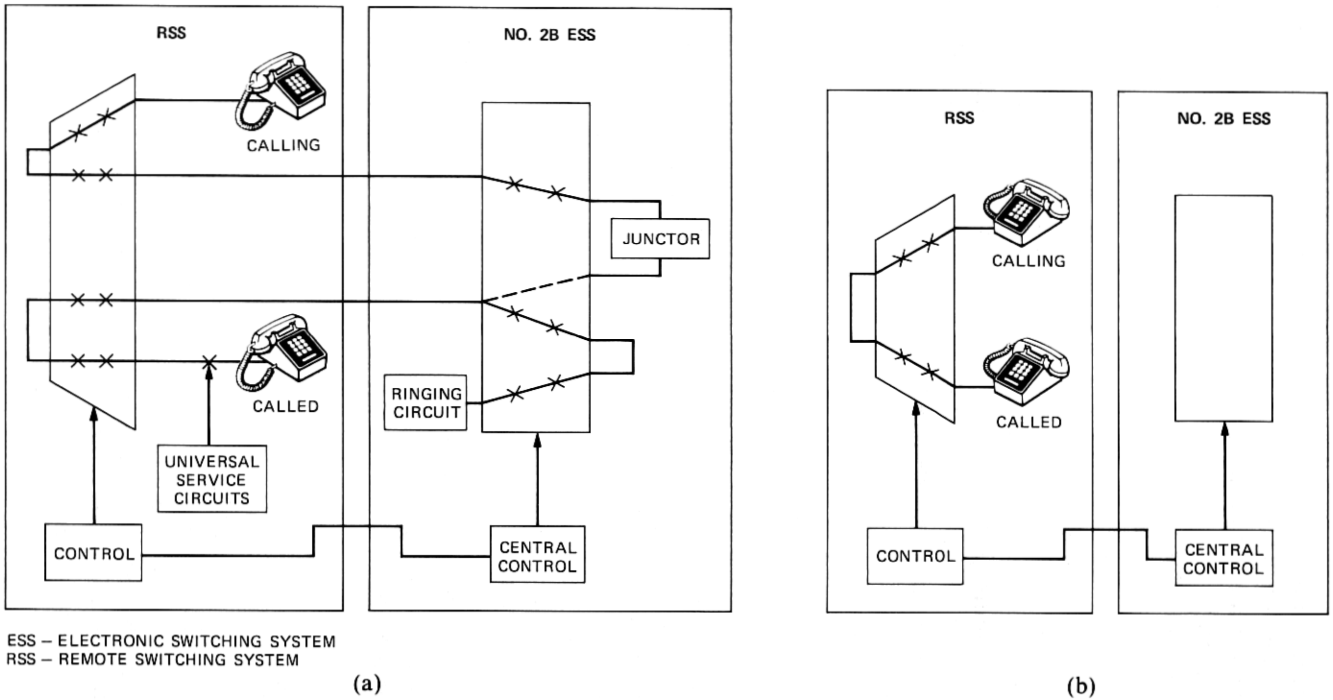


Fig. 6—Intra-RSS connection sequence. (a) Ringing connection. (b) Talking connection.

2.3.5 *Disconnect functions*

Disconnect actions for RSS calls are a function of the particular call configuration involved, i.e., intra-RSS calls or RSS calls through the host network. For call configurations involving both RSS and ESS paths, the ESS disconnect programs control the call-disconnect actions. The same disconnect sequence that is performed on ESS host lines is used on RSS channels that terminate on the host line network. This disconnect control strategy provides the ability to centrally recognize an RSS channel during normal host-disconnect processing. This recognition occurs when ESS programs perform a restore-verify action on the RSS channel. At this point in the host-disconnect processing, unique host RSS disconnect modules are invoked to disconnect the network path of the RSSs. The normal host-disconnect program and RSS-disconnect module then autonomously complete their respective disconnect actions. The only common resource between the two control programs is the RSS channel. The ESS host-disconnect program administers the host end of the channel on its network, and the RSS host-disconnect program administers the RSS end of the channel on its network.

In the case of an intra-RSS call, where the call configuration involves a connection totally within the RSS network and no channels or ESS network paths are involved, the RSS lines are supervised in the 10A RSS. Hits, flashes, and on-hooks are detected by the RSS; flashes and on-hooks are reported to the ESS host via data-link message. These messages are routed to unique RSS disconnect control modules in the host for proper processing. In the case of an on-hook, these programs perform the proper disconnect timing and then execute ROBs to disconnect the intra-RSS network paths and idle the lines involved. The intra-RSS call is reswitched-up on receipt of a flash message (as discussed in Section 2.3.4).

2.4 *Database integrity*

In an electronic switching system, the status of resources and telephone calls is recorded in temporary memory. This data can become mutilated because of program bugs, hardware errors, or program design errors, resulting in the loss of resources or system degradation. The problem is further complicated by RSS because parts of the new data structures in the host ESS are duplicated in the remote terminal. The new structures are:

1. PMRs for lines
2. PMRs for channels
3. Network map
4. Remote order buffers (ROBs)
5. Remote miscellaneous scan point map.

The actual data stored in the two copies of these structures are not identical in all cases since the host and remote terminal do not perform identical functions. For example, the state stored in a line PMR at the remote terminal represents the supervisory state of the line (origination, repeat supervision onto a channel, high and wet, etc.), whereas the state in the host PMR represents both the status of the line (idle, busy, maintenance) and the type of path memory configuration, as discussed previously. However, there is a mapping between the two sets of states in that the host-line state implies the possible supervisory states of the line. Deviations from this mapping should only be due to the time lag of the data link.

To ensure the integrity of the data structures, the first step is to prevent as many errors as possible. RSS software applies many of the techniques that have been successfully used in other ESS projects to avoid potential causes of errors. Some of these are good documentation, standardized program interfaces, structured design, structured programming, a high-level programmer's language, and a standardized data definition language. In addition, access to the new data structures introduced into the host is limited to the administrative programs that have the responsibility for that particular database.

The second step is to make the programs as error tolerant as possible since data errors will still occur. The main technique for this is defensive programming. The degree to which a data error is propagated through the system depends upon how the programs use the data. In order to have a minimal effect upon the system, programs should account for bad data. Some specific types of defensive coding techniques are:

1. Range checks on data to prevent overindexing tables
2. Accounting for all possible subroutine return code values
3. Use of symbolic definitions for data values
4. Accounting for all possible program inputs
5. Invalid data value checks.

Despite the preventive and defensive techniques that are employed, errors can still occur in the data. Programs are required to detect these errors and restore the facilities to the proper condition to avoid system degradation. These audit programs are responsible for detecting and correcting data errors and the initialization programs are responsible for restoring system facilities when the degradation is severe enough to cause major system degradation.

2.4.1 Audit programs

The integrity of the data structures is checked and corrected by a set of audit programs. Each audit program is individually tailored to a specific data structure or group of data structures and determines if the data items follow certain established rules. If the checks fail, the audit programs make appropriate corrections to all resources (both software and hardware) associated with the particular error and print error messages on the teletypewriter. The audit programs also aid in the initialization of the data structures.

The audit philosophy adopted for RSS is that each entity will maintain the integrity of its databases independently. If the host finds a discrepancy in its database, it corrects the problem by resetting the state of all host resources involved and instructs the remote terminal to put its facilities into a known (usually idle) state. If the remote terminal finds a discrepancy in its database, it sends a message to the host and the host audit programs initiate the actions given above.

Thus, there are three classes of audits associated with the RSS system:

1. Host audits that maintain the internal integrity of the data structures in the host
2. Remote terminal audits that maintain the internal integrity of the data structures in the remote terminal
3. Audits that guarantee that the host and remote terminal data structures are consistent.

Audit classes 1 and 3 are discussed below.

2.4.1.1 Host audits. Existing host audits are extended to include the new data structures and new data values introduced with RSS. The main audit modifications are for the RSS path memory, the RSS network map, channels, and ROBs.

The RSS path memory and network map are audited by making the following checks:

1. Point-to-point-back checks are performed between PMRs and PMJs.
2. Point-to-point-back from a PMR or PMJ to a call register are performed if linkage to a call register is indicated.
3. The junctor busy-idle bit in the network map is checked to ensure that it is idle if and only if the PMJ is idle.
4. Each of the other network map bits that is marked busy is checked to guarantee that it is in a valid path.

ROBs are audited by periodically rebuilding the idle link list and by timing ROBs associated with transient call records. If a ROB remains busy for an extensive length of time, the ROB and any associated call register are idled. All paths and circuits are also idled.

The corrective action taken by the host audits is to idle facilities (hardware and software) in the host and to send orders to the remote terminal to cause the facilities at that end to be idled.

2.4.1.2 Synchronization between host and remote terminal. The problem of maintaining the data structures in the host and remote terminal in synchronization is greatly simplified by taking advantage of the normal system operation. The remote terminal updates its data in response to orders from the host. Bits in the remote copy of the network map are marked busy or idle in response to orders to set up or tear down network paths. Thus, no network map audit is required between the host and remote terminal since the host does the hunting and idling of paths and the remote copy will tend towards the proper state even if it does temporarily get out of step.

Similarly, ROBs are controlled from the host end and normal operation will result in the remote copy being brought back into step with the host.

The remote terminal audits check that all equipped lines have supervision turned on. Any equipped lines that are unsupervised are reported to the host via a data-link message. If the host audits determine that the line state really calls for supervision to be on, the line and any associated resources are idled.

Periodically, the host sends a copy of its version of the remote scan point map to the remote terminal, which overwrites its map with the host's data. If the hardware state of the scan point differs from the map, the normal scan program will detect this as a change and report it to the host, resulting in both copies being brought back into step.

2.4.2 Remote terminal initialization

System initialization programs are responsible for correcting errors that prevent the system programs from cycling correctly. The initialization programs are usually executed as a consequence of errors being detected by the processor check circuits that monitor the sanity of the system operation. In the remote terminal, the primary checks for monitoring proper program operation are the system sanity timer, which monitors the main program cycle time; the write protect circuitry, which prevents illegal writes into program store; and certain peripheral error checks, which detect attempts to access unequipped areas of the periphery. If any of these errors are detected, it is indicative of an error in the system database. The method of recovery is to initialize a segment of the data and then return to the normal program cycle.

Since the initialization process inherently destroys a portion of the call-processing data, a corresponding set of calls will be lost, and it becomes a requirement for the initialization program to release the peripheral circuits associated with these calls. Any network links, channels, or service circuits employed on these calls must be idled by the initialization program before a return to normal system operation is begun. This is accomplished by releasing all the circuits that are marked idle in the initialized database.

Whenever an error is detected by a fault-detection circuit, a processor interrupt is generated that executes the fault-recovery programs. Various fault-recovery actions are taken, depending on the type of errors detected and their frequency.

The amount of data that is initialized on the first error is small. As successive errors are detected, the severity of the initialization is increased with the consequent loss of progressively greater numbers of calls. The ultimate action is to initialize the entire database and restore the system to an idle state. The goal of handling the fault recovery in stages, with increasingly more severe initializations, is to restore the system to a working mode with the loss of a minimal number of calls.

At either the remote terminal or the host, there are three fundamental levels of initialization: a minimal clear, a transient clear, and a stable clear. A minimal clear involves the initialization of the variable data associated with the active processes in the system and has the potential of disrupting, at most, several calls. A transient clear will initialize all the data in the system that is related to any call in progress. All calls in the process of being established or disconnected will be lost; however, calls in a stable talking state will be preserved. The final state of initialization is a stable clear where the entire database is reinitialized and all calls are lost.

For the RSS system, the initialization process is somewhat more involved than normal because the host and remote terminal databases are interrelated and an initialization level (phase) in one machine affects the database of the other system. Although the host machine is responsible for the control of the remote terminal on a call-related basis, the operation of the processors in the two machines is fairly autonomous with respect to their instantaneous activities. This is the situation for fault detection where the two systems are entirely independent. Each machine is responsible for initializing and carrying out its own fault-recovery actions and the level of the accompanying initialization will be solely determined by the conditions within the machine that detected the error. In effect, either machine is able to initiate any level of initialization on its own database independently of the other. However, as part of the initialization procedure, the hardware and software within the two machines must be synchronized so that the databases reflect a consistent set of calls. The calls that were destroyed in one machine must be reported to the other so they can be cleared from that system also. For example, a host-transient clear will destroy a number of calls that involve remote terminal lines. These calls must be cleared in the remote terminal so that periphery and call records are in agreement with those in the host.

The exact procedure for synchronizing the two machines will depend on which system has initiated the phase. Since the host is in charge of call control, its call records are regarded as the master copy and the remote terminal state is brought into agreement with its set of records. The host is therefore in charge of synchronizing the two machines.

Whenever a phase occurs in the host, the synchronization procedure is straightforward. The host will initialize its database and periphery and will then send initialization orders to the remote terminal to bring it into agreement with its updated records. When the remote terminal undergoes an initialization, it reports the level of the initialization to the host. In some instances, on a stable clear, for example, this is sufficient information to allow the host to initialize its database. In the case of a transient clear, it is also necessary to transmit a map of the lines that are in a transient state in the remote terminal. From the data specifying the initialization level and the map of transient lines, the host is able to update its call records and periphery. Once this is accomplished, it will conduct an initialization of the remote terminal in the same manner as for a host-initiated phase.

During a high-level initialization, interactions between the host and the remote terminal must be modified. For example, during the resynchronization of the host and remote path memory transient databases, it is necessary to prevent these databases from being accessed by call-processing routines. To implement this capability, the RSS is modeled as a finite-state machine with ten possible states. The state of each remote terminal together with a list of activities allowed in that state is maintained by the host in a database. Software, which functions differently based on the state of the remote terminal, checks this database to determine what action is appropriate in the given RSS state.

III. MAINTENANCE ENHANCEMENTS

Extension of telephone communication service through the 10A RSS brings with it an associated extension of maintenance capabilities. An obvious maintenance need relates to the SPUC/DL and data-channel facility between the No. 2B ESS and No. 10A RSS processors. Integrity and maintenance issues for this subsystem are dealt with in the companion article of this series, "Adding Data Links to an Existing ESS," by C. E. Ishman et. al.

Another area requiring maintenance enhancement is related to the addition of voice channels to the network architecture of the combined systems. Unlike other "links" of the No. 2B ESS network, the voice channels are nonmetallic, largely external to the central office environment, and are maintained by a separate craft force. These factors require the design to deal with issues of availability, transmission and noise performance, and trouble sectionalization. The channel maintenance software package, developed to address these issues, is described in the following sections.

Other maintenance additions, for telephone customer loop testing, are necessitated by the remoteness of the RSS and the nonmetallic network upon which these loops terminate. Testing capabilities, similar to those available for host-terminated lines, must be provided for RSS customer loop trouble detection and resolution. Such capabilities are described in the following sections on line maintenance. Finally, as a means of achieving test hardware economies in the RSS, the host diagnostic capabilities were expanded to test Dual-Tone Multifrequency (DTMF) receivers at the RSS as well.

3.1 Channel maintenance

Basic needs for the effective maintenance of RSS voice channels may be divided as follows:

1. The system must provide routine, automatically initiated tests that periodically diagnose individual channels to detect performance-affecting faults—particularly faults characterized by progressive and marginal degradation.

2. The above set of tests should also be initiated by call-processing programs in instances where in-process integrity checks indicate errors that could be caused by faulty channels.

3. The resolution of some hardware faults will require manual intervention. This is accomplished by: (a) allowing the above diagnostic tests to be executed upon demand, and (b) providing dc and ac test access to each channel for voltmeter-type testing. This latter facility has been provided through an enhancement of the existing central office Trunk Test Panel (TTP).

3.1.1 Channel diagnostics

The channel diagnostics, which play a major role in fulfilling all three needs, have been designed with a "start-small" philosophy; the sequence of tests is selected so as to confirm basic functions first, and then to build upon this knowledge in carrying out subsequent evaluation. The following sequence of tests is terminated when a failure is detected at any step:

1. Power cross test
2. False cross or ground test
3. Supervision test
4. Restore-verify test
5. Low line resistance test
6. Dial pulse test

7. ac far-to-near test
8. ac near-to-far test.

The power cross test, shown schematically in Fig. 7a, uses the same detecting circuit as do customer lines and is designed to detect crosses to central office battery or commercial power in the cabling between the No. 2B ESS switching network and the carrier terminal. By performing this test first, potential damage to other test circuits is avoided. The False Cross or Ground (FCG) test cannot use the existing test method since it is overly sensitive to the capacitative load seen at the input of the carrier's channel unit. Instead, the existing continuity and polarity test circuit is connected, as shown in Fig. 7b, to detect the existence of conductor-to-conductor and conductor-to-ground shorts (up to approximately 2000 ohms) in the same cabling.

The supervision test verifies the ability of the channel to pass on-hook and off-hook signals from the RSS remote terminal to the No. 2B ESS host. With the channel idle, on-hook supervision is first confirmed by applying loop battery toward the channel with the continuity test circuit (as diagrammed in Fig. 7c). After sending an off-hook order to the RSS (via the data channel), the central office current detector is scanned for the expected off-hook.

The restore-verify test, in a manner similar to that used with customer lines, verifies that the supervisory attending element can be reconnected to the channel and can detect the off-hook signal associated with a channel "origination." This is accomplished, as shown in Fig. 7d, by causing the RSS to transmit an off-hook signal toward the host when the host has its line attending element connected to the channel.

The low line resistance test is still another carry-over from host-line testing. Here, the objective is to detect the existence of high-resistance (up to approximately 15,000 ohms) crosses from conductor to conductor and from conductor to ground. While the intent of making such a test on a customer line is to prevent false ring tripping, the benefit to channel testing is to better characterize the nature of a fault. This test connection is shown in Fig. 7e.

Since originating RSS customers make use of central office digit receivers, the voice channels must be capable of conveying dialed digits, which are transmitted on the channel as a series of supervisory transitions. The dial-pulse test performs this function by: (1) connecting the central office end of the channel to a digit receiver, and (2) requesting the RSS to transmit a digit "0" (ten dial pulses). The test passes if the digit receiver shows that ten dial pulses have been received. The associated connections are shown in Fig. 7f.

The final two tests on the channel form a gross verification of its two-way transmission capability. The ac far-to-near test, shown in Fig. 7g, connects a source of milliwatt level 1-kHz tone at the remote end and a tone detector at the central office end. The complementary near-to-far test, shown in Fig. 7h, verifies transmission in the opposite direction and employs a similar technique. Since the tone detectors used in these tests are sensitive down to −30 dbm, the diagnostic results cannot confirm adherence to rigid (±1 dB) limits. The Remote Office Test Line (ROTL) feature has therefore been enhanced to perform accurate transmission measurements on voice channels as well as on trunks. This capability is discussed in the following section.

3.1.2 Automatic channel transmission testing

The ROTL feature, first implemented in the No. 2-EF-1 generic, was designed to allow automatic transmission testing of trunks by the Centralized Automatic Reporting on Trunks (CAROT) processor. Since the RSS channels have a functional role similar to that of local interoffice trunks, enhancement of the ROTL feature was selected as an efficient method for channel transmission testing. The resulting design, as implemented in the 2BE3 generic, proceeds along the following typical sequence in the testing of a channel:

1. The CAROT center connects to the ROTL access (incoming) port via the standard Direct Distance Dialing (DDD) network.
2. Priming data specifying the test parameters is sent from the CAROT, through the ROTL access port, to a Multifrequency (MF) receiver. This digit receiver is accessed via a normal central office connection from the ROTL test port, as shown in Fig. 8.
3. Upon receipt of the channel identity, the central office releases connections to the MF receiver, establishes a connection between the ROTL test port and the selected channel, and requests the RSS to connect a transponder to the remote end of the selected channel.

During this sequence of events, considerable data "handshaking" takes place between the CAROT and the No. 2B ESS—much of this communication in the form of bursts of "test progress tone" sent by the ROTL circuitry to the CAROT. The end result is a composite connection, as shown in Fig. 9, which permits the RSS transmission measuring transponder (termed a miniresponder) to interact with the CAROT in performing transmission measurements in either direction on the channel. The CAROT may then restore or remove channels from service based on the test results.

Since the CAROT-ROTL system can make loss and noise measurements accurate to 0.1 dB, this design provides a low-cost means of maintaining voice channel transmission characteristics within acceptable limits. Also, because the RSS miniresponder can be accessed in a similar fashion using a portable ROTL System Test Set, the same high-resolution measurements are available to the maintenance craft on a demand basis.

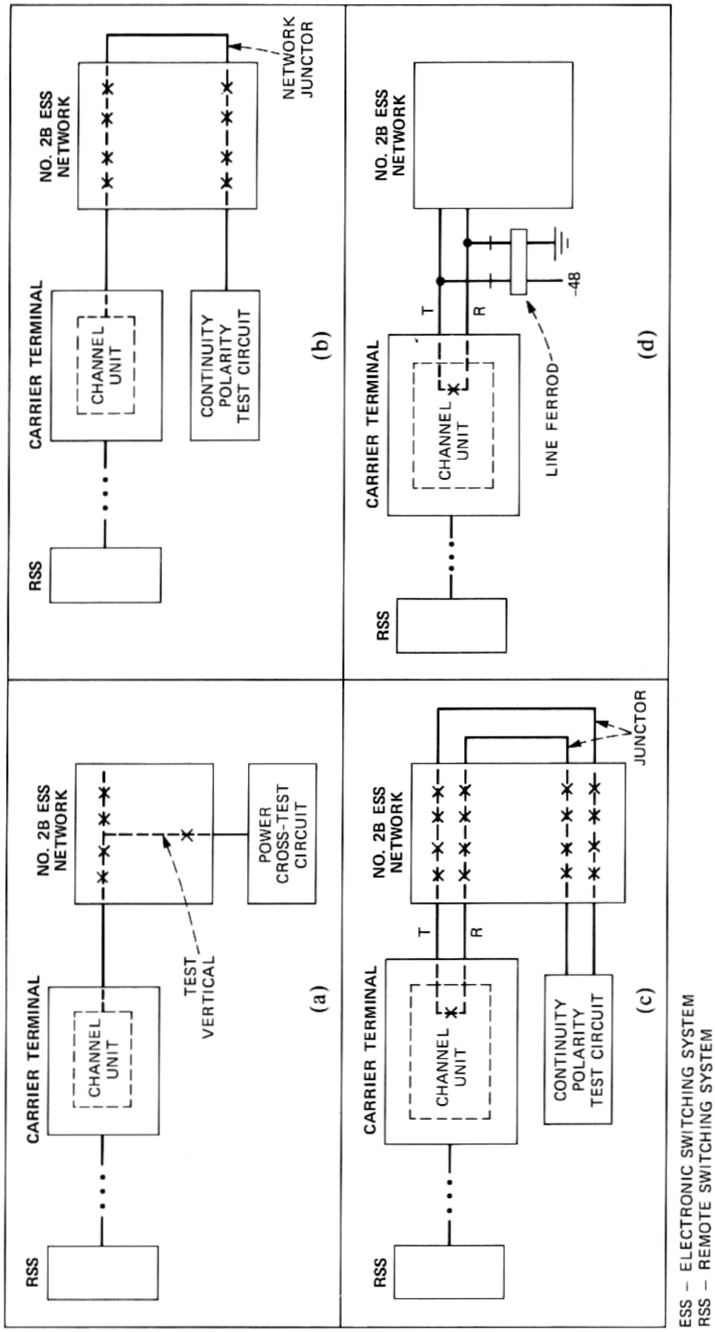


Fig. 7—Test configurations of channel diagnostic: (a) Power cross test. (b) False cross/ground test. (c) Supervision test. (d) Restore-verify test.

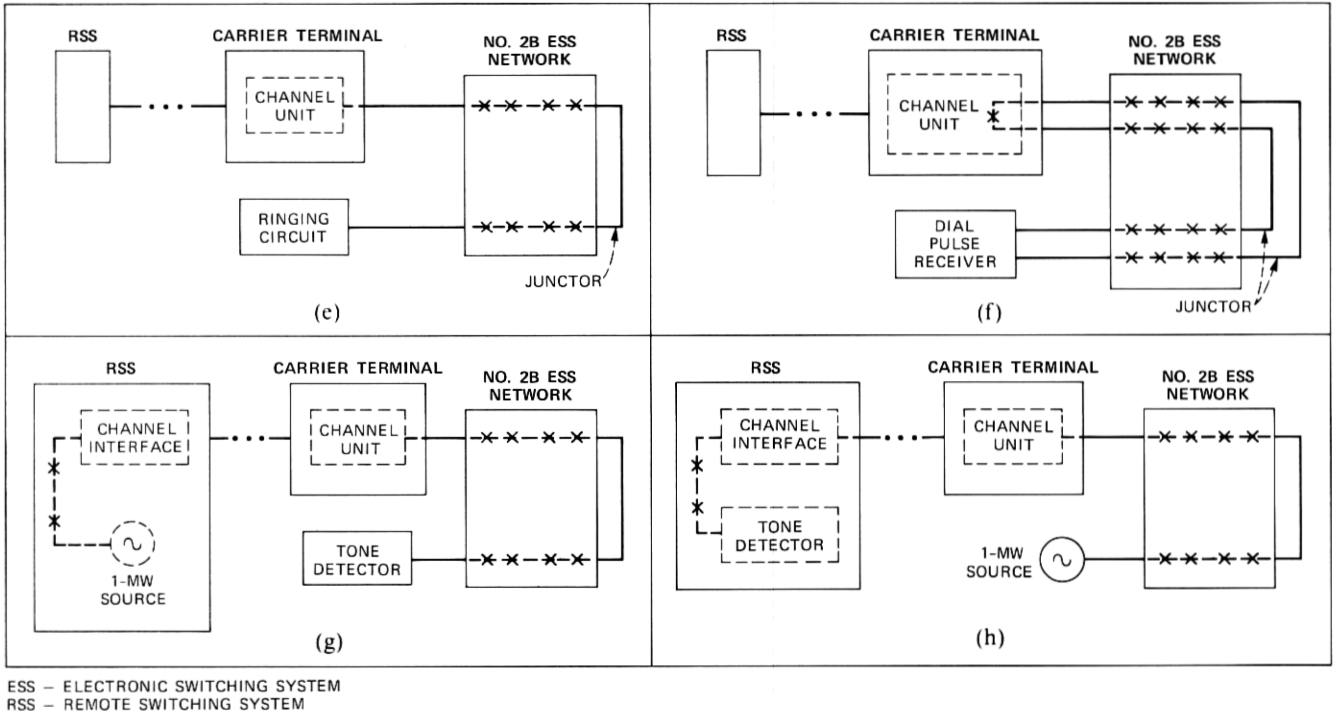


Fig. 7 (Continued)—Test configurations of channel diagnostic: (e) Low-line resistance test. (f) Dial pulse test. (g) AC far-to-near test. (h) AC near-to-far test.

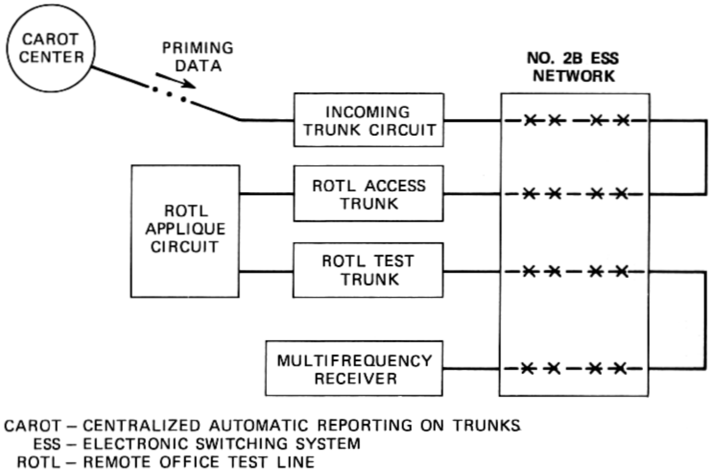


Fig. 8—Initial CAROT-ROTL connection.

3.1.3 In-process error detection

While routine testing is an essential ingredient in the channel maintenance package, means must also be provided to deal with channel-related failures encountered during the processing of a telephone call. Such a facility will properly dispose of channels with transient faults (which are not detected during periodic exercise) and channels with newly occurring faults.

Once a channel is implicated by a failure in call processing, automatic routines dispose of the channel as follows:

1. A diagnostic test is run on the channel and, if the test fails, the channel is removed from service (subject to previously specified numerical limits).

2. If the diagnostic passes or cannot be run because of resource blockage, a report of the incident is made to error analysis programs running at the RSS.

3. The error analysis routines, which accept failure input from the remote terminal as well as the host, evaluate the failure history of each channel. This evaluation is carried out using two algorithms: a peer group comparison and a "quick check."

4. If the peer group analysis indicates that a given channel's error rate is significantly higher than that of its peers, that channel will be removed from service (subject to the specified maintenance limits).

5. If the error analysis indicates a "quick check" failure (three successive channel errors occurring on the same channel), channel diagnostics are run on the suspected channel. Depending on the

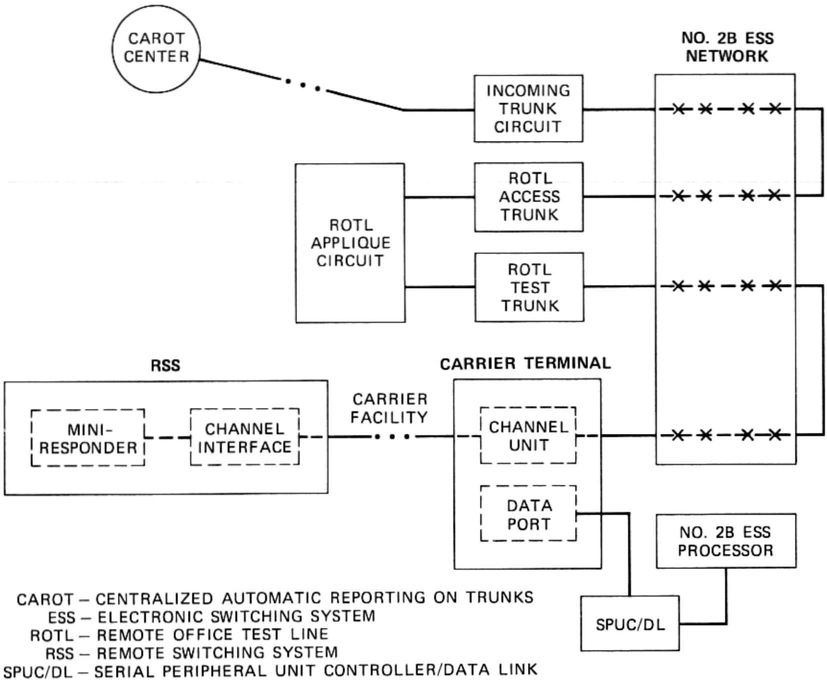


Fig. 9—Channel testing with CAROT and ROTL.

success or failure of the diagnostic exercise, the action taken is the same as that listed in (1) and (2) above.

3.1.4 Manually controlled channel testing

As previously mentioned, the ROTL capability for channel (a) testing may be used automatically (via CAROT) or manually. In addition, to provide direct metallic access to the channel and control of its associated circuit states, the central office trunk test panel programs have been modified to provide test functions similar to those available for trunks.

A typical sequence, which permits a loop-around connection of two channels, is itemized below. Such a connection is used in two-way transmission loss measurements on channels.

1. Using the panel-mounted telephone set at the trunk test panel, digits are dialed that direct a specified channel (a) to be connected to a source of milliwatt tone (i.e., a 102-type test line) at the RSS. This connection, shown in Fig. 10, permits the TTP transmission measuring set to determine the loss of the channel in the far-to-near direction.

While this channel is accessed, existing keys at the TTP may be used to change the circuit state of the channel at the RSS. (These

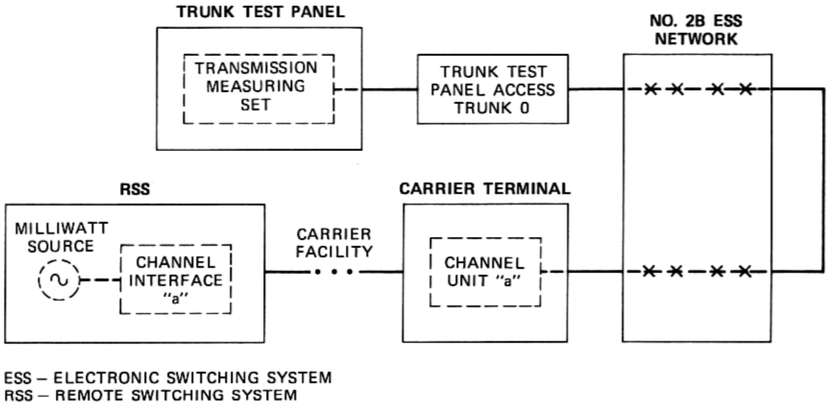


Fig. 10—Measurement of channel far-to-near loss.

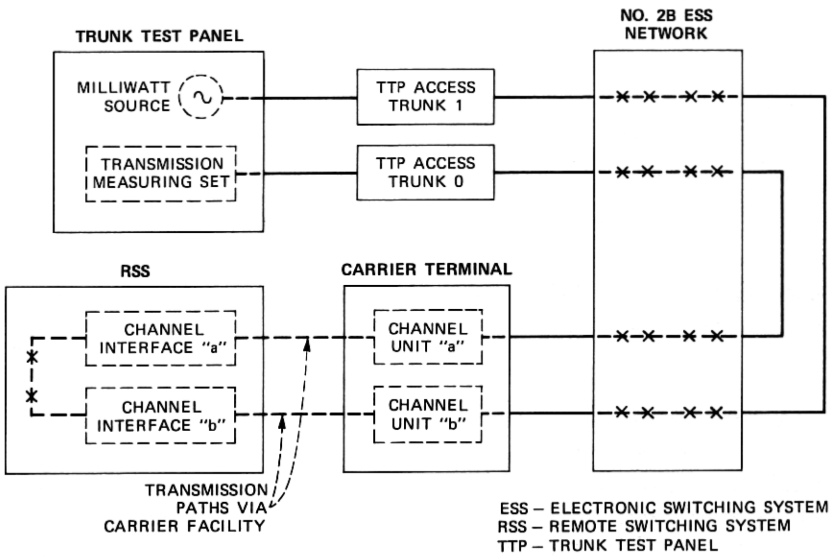


Fig. 11—Measurement of channel near-to-far loss.

states correspond to supervisory conditions, loss pads in and out of circuit, and the connecting of a balanced terminating network.)

2. Dialing a different test code and channel identity will cause a second channel (b) to be connected between another TTP access trunk and a loop-around connection at the RSS. This connection, shown in Fig. 11, has produced a configuration in which both ends of a two-channel transmission path are connected to the TTP. Now with a source of 0 dbM connected to the second channel and a transmission-

measuring set connected to the first, the near-to-far loss on the second channel may be determined.

3. When any channel is released from the TTP, several options are possible, depending upon the channel's previous status and the method of TTP disconnect. For example, if the channel had initially been out of service, TTP release will initiate a diagnostic test of the channel, restoring it to service if the diagnostic passes. If an initially idle (in-service) channel is released with the "make busy" key operated, the channel is unconditionally removed from service.

3.2 Line maintenance

Like channel maintenance, line maintenance may be subdivided into capabilities that: (1) by periodic, automatically initiated testing, attempt to detect progressive hardware degradation before service is affected; and (2) by allowing demand-type human-controlled testing, enable the maintenance craft to resolve the nature and location of faults. Automatic Line Insulation Testing (ALIT) occupies the first of these categories and, like its counterpart for host-terminated lines, detects high-resistance crosses and grounds on metallic pairs between the RSS and the customer premises. The second category of features includes an interface to the Local Test Desk (located in a centralized repair service bureau) and the station ringer test, which verifies the correct operation of the subscriber's station set.

3.2.1 Line insulation testing

If it were not for the nonmetallic network in the RSS remote terminal, line insulation testing could be performed by the host No. 2B ESS. Instead, the remote terminal is equipped with its own insulation testing circuit and the means to make a metallic connection between this circuit and all customer line terminations.

As shown in Fig. 12, the RSS ALIT circuit accesses the customer line via the Metallic Access Bus (MAB), the Line Test Access Bus (LTAB), and the Universal Service Circuit (USC), the latter in the "bypass" state. RSS firmware has exclusive control of this connection and testing sequence once a line and test parameters are specified via data order. That is, after the RSS receives a line test request from the host, the remote terminal selects an idle USC, connects the MAB, performs the test, disconnects from the line, and returns the test results in an acknowledgment message over the data channel. The test parameters received in the requesting message are similar to those specified for host line testing and permit testing sensitivities ranging from 80 kΩ to 5 MΩ. In addition to the line identity, the requesting order also includes a specification of the test mode. The mode, an operating company and craft option, selects the extent of the testing performed on an individual line. The choices are:

1. A foreign EMF (electromotive force) test, which looks for false power crosses to either conductor,

2. The TRG (tip-ring to ground) test, which measures resistance between the conductors (tied together) and ground,

3. Leakage test, which measures resistance between the conductors,

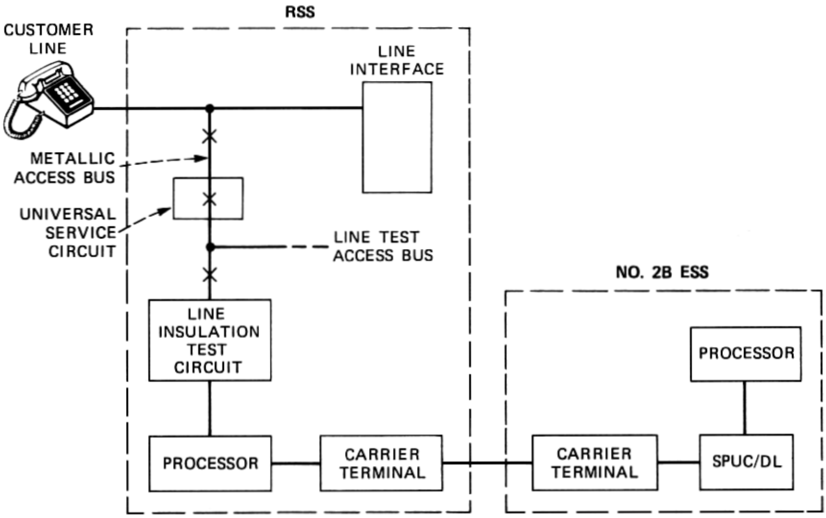
4. A "general" test, consisting of a sequential application of all the preceding tests.

RSS line insulation testing is included in the automatically initiated sequence and follows, RSS by RSS, the insulation testing of host lines. While the testing of RSS lines could have been done concurrently with the testing of host lines (since separate and dedicated hardware is involved), such a design would have increased the complexity of teletypewriter output messages, which identify line failures. By completing the testing of all lines in one entity (i.e., the host, or one of the RSSs) before proceeding to test lines in the next, a single "header" message serves to identify the entity for all line failures that follow. This has the added benefit of grouping those reports corresponding to their geographic locations.

3.2.2 Local test desk

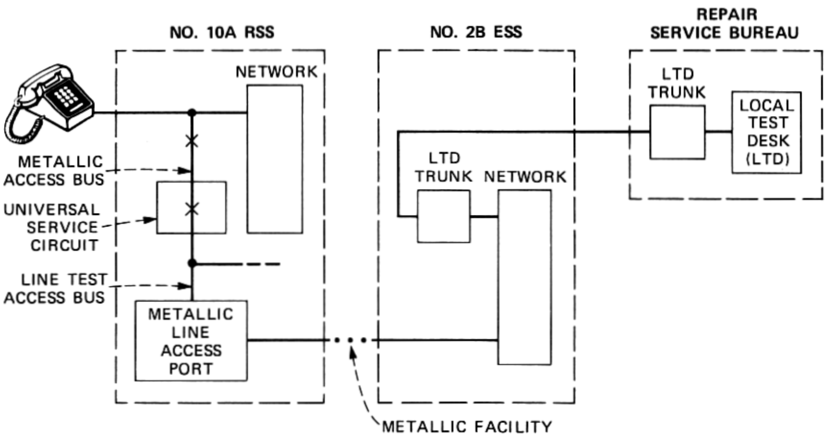
Demand-type line testing of host and RSS lines may be performed from a test cabinet, located within the No. 2B ESS central office, and from a Local Test Desk (LTD) at some remote location. To accommodate the variation of metallic and carrier facilities between the LTD and the host and between the host and an RSS, a number of options have been provided. The simplest case, from a design point of view, exists where there is a low resistance (less than 2600 ohms) metallic path from the LTD to the RSS customer's telephone. As shown in Fig. 13, the connection from the LTD to the host uses the standard No. 2B ESS—LTD incoming trunk circuit and a dedicated metallic facility between the RSS and the host. This metallic connection is completed in the RSS by connection through the Metallic Line Access Port (MLAP), the USC, and the MAB. LTD testing, which consists of various voltage and resistance measurements, may then proceed over the established connection.

Since such low-resistance facilities are expensive and rarely found in the rural and suburban environment, an alternative implementation is provided. This arrangement takes advantage of the Remote Testing System (RTS), which was previously provided to work over nonmetallic facilities between the LTD and the central office. The resulting design employs a multifrequency telemetry Remote Line Test (RLT) unit at the RSS. As shown in Fig. 14, the microprocessor-controlled



ESS – ELECTRONIC SWITCHING SYSTEM
RSS – REMOTE SWITCHING SYSTEM
SPUC/DL – SERIAL PERIPHERAL UNIT CONTROLLER/DATA LINK

Fig. 12—RSS line insulation testing.



ESS – ELECTRONIC SWITCHING SYSTEM
LTD – LOCAL TEST DESK
RSS – REMOTE SWITCHING SYSTEM

Fig. 13—Metallic option for the local test desk.

RLT has metallic access to the customer line via the MLAP, USC, and MAB. The connection of the LTD and RLT provides for test requests being sent to the RLT and for test results being returned to the LTD. Supervisory signals from the LTD (e.g., disconnect) are

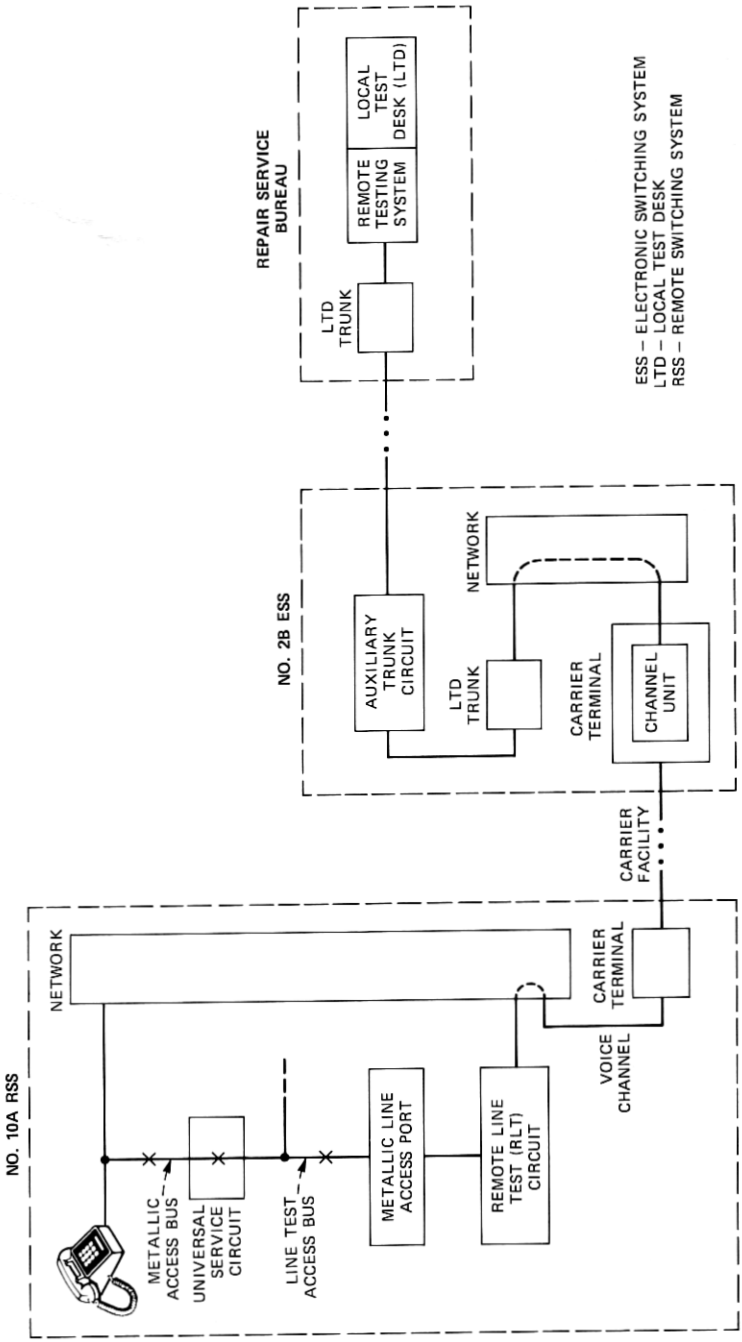


Fig. 14—RLT option for the local test desk.

intercepted by the auxiliary trunk circuit in the No. 2B ESS, since the host is responsible for maintaining and tearing down the entire connection. The RLT and the auxiliary trunk circuit at the host have therefore functionally replaced the central office equipment associated with the remote testing system.

3.2.3 Differences between host and RSS LTD line testing

Since the two testing options (RLT or MLAP) require different trunk circuit hardware at the central office, the incoming call from the LTD may appear on either type of trunk. This requires the LTD operator to know the testing option associated with a particular RSS, and it requires the host No. 2B ESS to verify that the incoming LTD trunk type agrees with the testing option of that RSS. This latter confirmation is performed when the line's directory number is translated into an RSS identity.

The Local Test Cabinet (LTC), normally located near the distributing frame in the central office, is not equipped with the remote testing system. Thus, the LTC is incompatible with RLT-equipped RSSs and can test RSS lines only if that RSS is equipped with the MLAP option.

Another difference between LTD/LTC testing of host versus RSS lines concerns the action taken if the line is found busy when the initial test connection is made. When a host line is being tested, a suitably marked incoming LTD trunk can be connected to the busy line by means of the no-test facility of the No. 2B ESS switching network. If the host line becomes idle during the LTD connection interval, the line is marked busy (again) and the LTD is reconnected using a normal network path. In the analogous RSS case, the LTD is bridged onto the existing line connection in the RSS network. Since the software line status may not be changed, a camp-on request is registered at the RSS. When the line becomes idle, the pending LTD request causes the line to be rebusied, but no path reconfiguration is required.

One of the LTD functions is to verify whether a given line can originate. This is done on host lines by requesting the line attending element (ferrod) to be reconnected to the customer loop and then placing a resistive bridge across the loop. When the off-hook condition is sensed, dial tone is returned to the tester via the LTD trunk circuit, as shown in Fig. 15. Such operation is not possible with RLT testing of an RSS line since returning dial tone through the LTD trunk would open the LTD-RLT path and prevent further communication between them. Instead, when a line origination test is to be performed on an RSS line, that line is connected to a digit receiver in the host (shown in Fig. 16). The digit receiver then performs the functions of: (1)

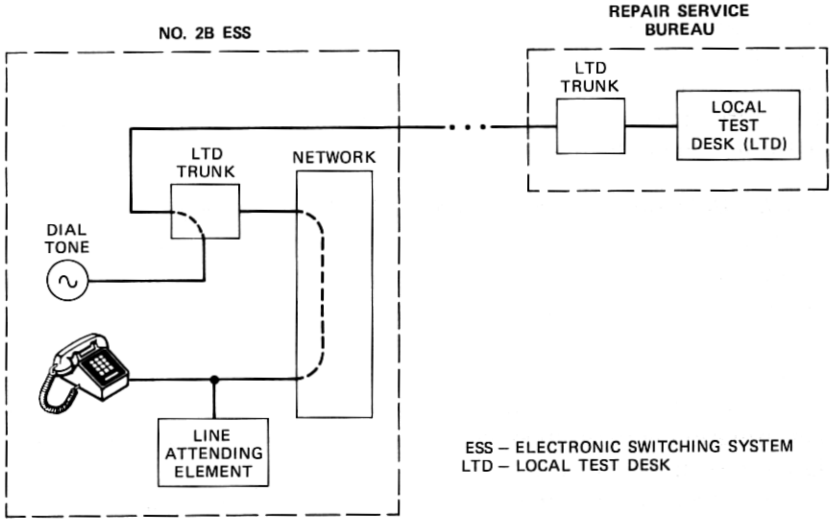


Fig. 15—Local test desk line origination test for host lines.

detecting supervisory changes on the line, and (2) returning dial tone when the off-hook signal is detected. Dial tone attachment and removal can then be monitored by the LTD, which is connected to the customer's line.

3.2.4 Station ringer testing

The station ringer test is under the control of the customer station and permits the verification of:

1. Station dialing capability
2. Party-identifying ground (in the off-hook state)
3. On-hook leakage, and
4. Ringing code and ring trip.

Since the sequence of actions required to perform these tests is well known to craft accustomed to testing host lines, a basic requirement for the service was that the craft interface remain unchanged for RSS lines.

The procedure, and its implementation for RSS lines, is diagrammed in Fig. 17. The test sequence begins by dialing a special code, normally a unique NNX, followed by the last four digits of the line's directory number. The RSS line is then connected to the host's station ringer test circuit via the same channel that had previously been connected to the customer digit receiver. The first step, the off-hook resistance test, is initiated with a switchhook flash by the tested line. A data-link order sent to the RSS causes the resistance test to be made by a

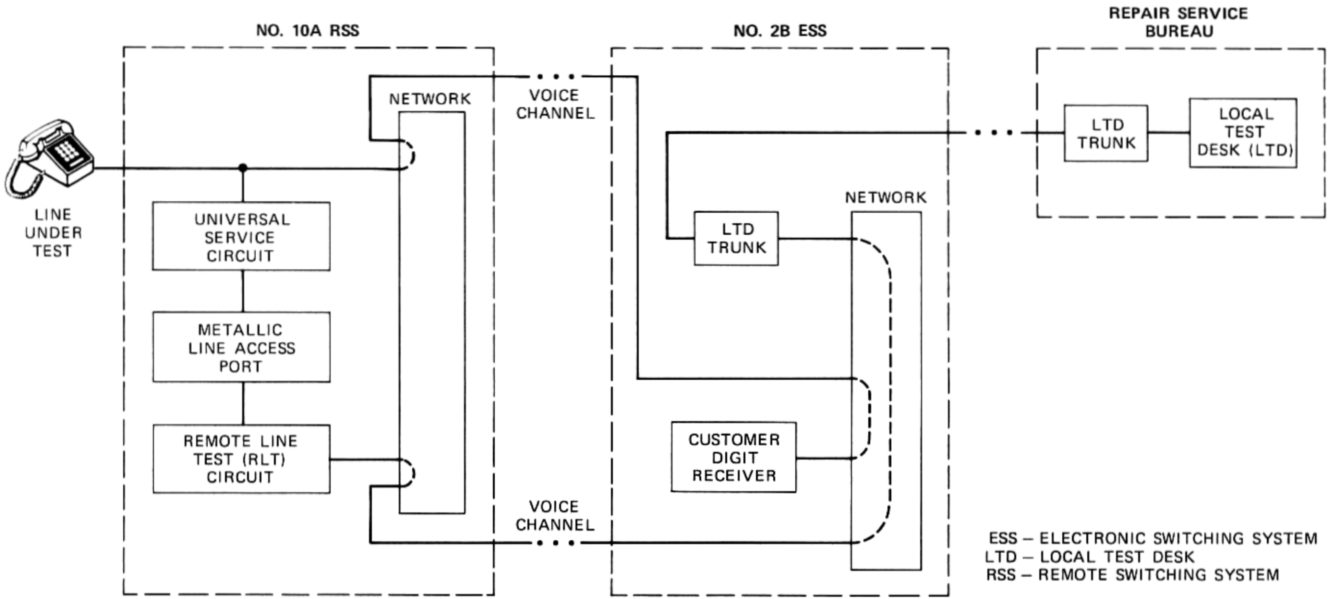


Fig. 16—Local test desk line origination test for RSS lines.

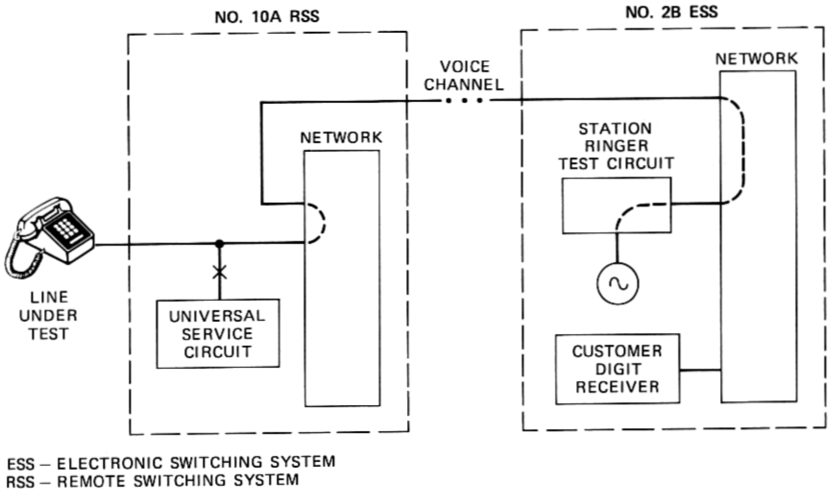


Fig. 17—Station ringer testing for RSS lines.

USC. The results of this test are then indicated with a steady vs. interrupted tone returned by the station ringer test circuit.

The second test on a (noncoin) customer line is an on-hook leakage test and is initiated by on-hook supervision. Again, the USC at the RSS is directed to perform the resistance test; its results are returned to the host via data-link message. If the resulting leakage measurement is acceptable, terminating ringing is applied to the line by the USC. Ringing indicates the result of the leakage test as well as confirming that the station set ringer is functional and that the correct ringing code is applied. The final test, ring trip, is made if the station set is taken off-hook within the next 3 minutes. As in the previous instances, since the ringing is applied by the RSS's USC, the tripping of ringing must be reported to the host via a data message.

At this point, the station ringer test circuit is connected to the line under test. A station set on-hook would idle all connected resources or, if a repeat test is desired, a switchhook flash causes the off-hook resistance test to be made, starting another test cycle.

3.3 DTMF receiver testing

As we mentioned previously, the RSS is capable of "stand-alone" operation should its interface with the host be lost. This feature therefore required that DTMF receivers be equipped in the RSS so that customers utilizing *Touch-Tone** dialing may be served. To

* Trademark of AT&T.

achieve design economies in the RSS, a DTMF test circuit (like that in the host) was not provided. The alternative to manual routine testing was to provide an automatic diagnostic using the host's DTMF test circuit.

Such operation is possible because the stand-alone circuits (including the DTMF receivers) have access to the RSS voice channels via a link "multiple" arrangement shown in Fig. 18. This network design, originally intended for mutually exclusive connections of RSS lines to either channels or stand-alone circuits, permits a voice channel connection to a DTMF receiver without use of the RSS network junctors. The resulting configuration during the execution of the diagnostic involves the host's DTMF receiver test circuit, as shown in Fig. 19.

Maintenance software is furnished to diagnose one or more receivers on a demand or automatic basis. For the automatically initiated case, diagnostics of the RSS receivers are carried out one RSS at a time following the corresponding diagnostics for host digit receivers.

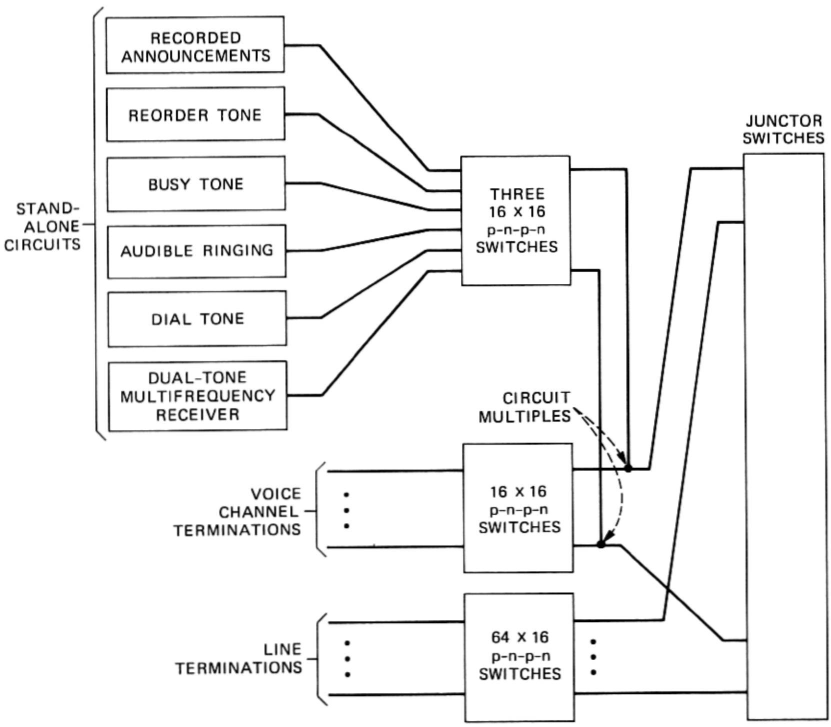


Fig. 18—Stand-alone circuit network connections.

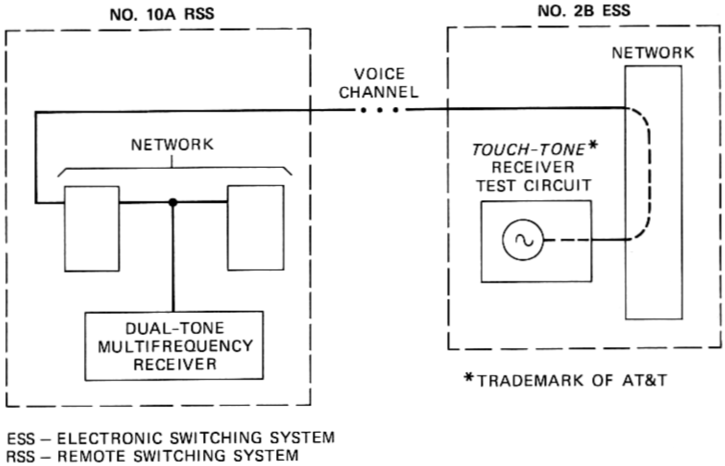


Fig. 19—*Touch-Tone* receiver diagnostic connections.

IV. REAL-TIME PERFORMANCE

This section discusses two aspects of the RSS feature: the real time required to process an RSS call and the effect of RSS on the call capacity of an office.

The many components of an RSS call were determined by performing extensive measurements in the system laboratory. The results of this study indicated that an ESS line-to-RSS line call requires about 33 percent more processor real time than a host line-to-line call. Similarly, an RSS line-to-ESS line call requires about 49 percent more real time and an intra-RSS call (reswitched-down) requires about 113 percent more processor real time than a host line-to-line call.

Because of overhead associated with each RSS hosted by a No. 2B ESS and real-time factors associated with each RSS call, the No. 2B ESS call capacity is reduced with an increasing proportion of RSS traffic. The amount of this reduction is a function of the number of RSSs hosted and the RSS traffic.

When this project was still in the initial planning stages, it was evident that providing RSS capabilities would affect the host office capacity. Throughout the entire development of the feature, considerable emphasis was placed on performance issues. This included such items as: planning effective communication strategies between the host and SPUC, minimizing RSS-related work in non-RSS segments of code, and optimizing and fine tuning frequently executed functions. The result of these activities is a relatively small penalty in host capacity due to RSS.

V. ACKNOWLEDGMENTS

The authors wish to acknowledge the outstanding efforts of the many individuals at Bell Telephone Laboratories, American Telephone and Telgraph, and Western Electric who have contributed to this work.

AUTHORS

Donald W. Brown, B.S.E.E., 1962, University of Louisville; M.S.E.E., 1964, New York University; Bell Laboratories, 1962—. Mr. Brown has been engaged in the system design of proposed Signal Transfer Points for the Common Channel Interoffice Signaling network. Since 1976, he has worked on system and program design for the Remote Switching System. He was past Supervisor of the RSS Remote Terminal Maintenance Group and is now Supervisor of the Cell Maintenance Group for the Advanced Mobile Phone Service system.

John J. Driscoll, B.S.E.E., 1964, Clarkson College of Technology; M.S.E.E., 1966, Stevens Institute of Technology; Bell Laboratories, 1964—. Mr. Driscoll has been engaged in various aspects of electronic switching system development. Initially, he worked on maintenance and call-processing software development for No. 2 ESS, and later he worked on exploratory studies and call-processing software development for the Advanced Mobile Phone Service. He was involved with call-processing and administrative software development on the No. 1 ESS and the No. 2B ESS RSS systems. Presently, he is working on No. 5 ESS performance evaluation and capacity improvement.

Frederick M. Lax, B.S.E.E., 1974, University of Notre Dame; M.S.E.E., 1975, Massachusetts Institute of Technology; Bell Laboratories, 1974—. From 1975 to 1979, Mr. Lax was engaged in software development for No. 1/1A ESS call-processing and No. 10A Remote Switching System (RSS) feature development. In 1979, he was named Supervisor of the RSS First Application/General Availability Features Group and, subsequently, Supervisor of the No. 5 ESS Audit Systems Group. He is presently Head of the No. 5 ESS Software Recovery Department.

Michael W. Saad, B.S.E.E., 1964, M.S.E.E., 1965, University of Illinois; Bell Laboratories, 1966—. M. Saad first worked in the maintenance software area for the Automatic Voice Network (AUTOVON) system. His No. 4 ESS experience includes call-processing design, peripheral error-analysis strategies and design, peripheral-unit-configuration program design, and No. 4 ESS translations for international switching. Mr. Saad was supervising the 2B ESS RSS Call-Processing Group and is currently supervising the Administrative Services Group for No. 5 ESS.

J. G. Whitemyer, B.S.E.E., 1963, University of Akron; M.S.E.E., 1965, Ohio State University; Bell Laboratories, 1963—. Mr. Whitemyer joined Bell Laboratories in 1963 and worked in the Local Crossbar Switching Laboratory where he was involved in the development of Custom Calling Services and *Picturephone** Service on the No. 5 Crossbar System. He was promoted to

---

* Registered service mark of AT&T.

Supervisor in 1970 and, shortly thereafter, took responsibility for the system design of No. 3 ESS. Subsequent assignments included responsibility for circuit and microcode design of the 3A Central Control, field support for the No. 3 ESS and No. 2B ESS, and software design for the No. 2B ESS. He is presently Supervisor of the 2BE4 Project Management and Software Group. Mr. Whitemyer holds patents relating to the implementation of telephone switching features and telemetry.