

## **No. 4 ESS:**

# **Prologue**

By A. E. SPENCER, Jr.

(Manuscript received February 23, 1977)

No. 4 ESS, a high-capacity, toll and tandem switching system, is the largest single system development ever undertaken by the Bell System. It is also the vehicle by which electronic switching was first introduced into the Bell System long distance telecommunications network. During 1976, four No. 4 ESS offices were placed in service—the first in Chicago in January, the second in Kansas City in July, and the third and fourth in Jacksonville and Dallas in December.

No. 4 ESS had its origin in the 1950s in both research and planning. Research had provided the fundamental time-division switching techniques on which No. 4 ESS is based, and planning revealed that a very large switching system would be required to cope with expected growth.

Preliminary development began in 1968. By 1970, a specific plan was laid out. This plan required development of a high-speed processor, a new device interconnection technology, a digital switching network and associated transmission terminal equipment, a large body of software, new manufacturing facilities, and new installation and operating procedures.

A system as large and complex as No. 4 ESS would have been difficult (if not impossible) to achieve without the benefits of the integrated Bell System structure. Building on a background of solid research, the integrated design and manufacturing team worked with systems engineers, with switching, transmission, and device development engineers, with planners from AT&T headquarters, Long Lines, and the operating telephone companies, and with members of the manufacturing, documentation, and training and installation forces over a period of 7 years to introduce No. 4 ESS.

The advantages to the Bell System and its customers clearly justify this tremendous effort. Through No. 4 ESS, operating and maintenance costs for toll switching systems are expected to be reduced to one-third of what they otherwise would have been, building and floor space requirements will be cut to only a quarter of what they would have been,

terminal costs for digital transmission facilities terminated on No. 4 ESS will be reduced to the point that T-carrier systems will prove in at zero length, and finally, the stored program control will introduce a flexibility for new features as yet undreamed of by telecommunications customers.

In addition, the No. 4 ESS offers new opportunities for integration of switching and transmission. The switching system now reaches out beyond its normal limits to help maintain connecting transmission terminals. Furthermore, dealing with PCM multiplexed channels carrying a number of talking paths offers additional opportunities for improved performance, reduced capital expenditures, and reduced installation, maintenance, and administration costs.

This issue begins with an overview of the No. 4 ESS system and provides a brief history of the project and the background for the other papers. While omitting details, the collection of articles provides a comprehensive overview of No. 4 ESS. To limit the volume to a manageable size, the 1A Processor and 1A Technology have already been covered in a separate issue.

It is impossible to adequately acknowledge the contributions of everyone involved in a project of the magnitude of No. 4 ESS—people from many organizations of Bell Telephone Laboratories, Western Electric, AT&T, Long Lines, and the operating telephone companies all participated in important ways. The authors of this volume would like to express their gratitude to all of these people for the unity of purpose and free communication which overcame the complex organizational interfaces and technical problems, and permitted No. 4 ESS to exceed initial objectives and to be completed on schedule.

## No. 4 ESS:

# System Objectives and Organization

By A. E. RITCHIE and L. S. TUOMENOKSA

(Manuscript received February 8, 1977)

*This article is an introduction to a series of articles that describe the No. 4 Electronic Switching System. Objectives and the organization of the system are given and the overall operation is explained.*

## I. INTRODUCTION

### 1.1 Background

A new toll Electronic Switching System, No. 4 ESS, has been developed to meet the expanding needs of the telecommunications network, including the effects of continuing toll message growth. The first four No. 4 ESS offices were placed in service during 1976, starting with the Chicago 7 office on January 17.

The purpose of this article is to serve as an introduction to the ten more technically detailed articles that follow. This introduction highlights the objectives of the new system in comparison to existing Bell System toll, or trunk, switching systems and outlines the features and characteristics of No. 4 ESS that differ from previous electronic switching systems, particularly the No. 1 ESS.

Expectation of a trunk switching version was implicit in the early planning and development of the first Bell System local electronic switching system, which came to be known as No. 1 ESS. It was clear even then (late 1950s) that if the anticipated improvements in flexibility, reliability, and economy were realized, these advantages would be equally applicable to toll and tandem switching.

The opportunity to test this thesis came early. The first local No. 1 ESS was cut into service in May of 1965.<sup>1</sup> Even before this initial cutover, upon the urgent request of the United States government, development was started of a four-wire combined line and trunk switching system based upon No. 1 ESS and designed for use in a special noncommercial switching network. This version of No. 1 ESS is now the principal switching system used in the government-controlled AUTOVON (Automatic Voice Network) in the contiguous United States. Although it had

Table I — Bell System toll switches, January 1, 1967

| Systems                   | Erlangs | Attempts/<br>Hour | Total<br>Number |
|---------------------------|---------|-------------------|-----------------|
| No. 4A crossbar           | 6,200   | 116,000           | 73              |
| No. 1 crossbar tandem     | 1,700   | 47,000            | 237             |
| No. 5 crossbar local/toll | 1,100*  | 20,000*           | 407             |
| Step-by-step intertoll    | 330*    | 5,000*            | 500             |

\* Nominal

been expected that the special four-wire No. 1 ESS might evolve into a commercial toll ESS, this did not occur. The primary reason for this was that growing requirements for switching capacity appeared to exceed the inherent capability of the specialized AUTOVON switch.

### 1.2 Planning studies

Serious study specifically aimed at determining Bell System needs for a toll ESS began at Bell Laboratories early in 1966. At that time the principal Bell System toll switches consisted of the systems listed in Table I. Of these several types of systems only one, No. 4A crossbar, provided full toll capability, including a four-wire network, for service as a control switching point in the network hierarchy.

The early studies addressed the problem of determining the maximum traffic, or call-handling, capacity that should be set as an objective for the new system. Traffic capacity has three interrelated dimensions which must be maintained in reasonable balance:

- Trunk connections or terminations
- Office network load
- Call attempts per hour

In principle, optimum traffic capacity can be determined as a function of nationwide network structure, traffic load, and economics; in turn the desired traffic capacity has a profound effect upon switching system architecture and technology. Alternatively, the state of the switching art may set a limit upon capacity lower than otherwise desirable.

For these reasons, system studies and development studies proceeded in parallel, with a continual interchange of information and mutual effort to match the desirable with the possible. Historical trends and projections into the future indicated a continuing growth of toll message traffic at a rate of 8 to 9 percent a year. When this growth was translated into switching requirements and matched against the capacity of the largest available toll system, No. 4A crossbar, it was clear that, by the late 1970s, many metropolitan areas would require a multiplicity of toll switching

offices. This results in network inefficiencies and administrative complexities, both of which carry economic penalties. In order to quantify these effects and relate them to an objective system capacity, two major study efforts were undertaken.

One pioneering study investigated in detail what came to be known as the multimachine penalty. The penalty accrues from the lowered efficiency of "splintered" trunk groups, additional interoffice trunking, more traffic routed up the hierarchy, and double-switching. The studies were conducted on model metropolitan areas based on New York and Los Angeles.

The second study was directed at a projection of the economics of future toll switching installations in each of 241 metropolitan areas in the contiguous United States. Data were gathered for each area on existing installations, the current traffic load, and forecasts of future traffic growth. From these parameters, a computer model was constructed which could accommodate switching offices with any postulated capacity, cost, and traffic characteristics, including multimachine penalties. By means of this model, comparative costs could be determined over any span of years on an area-by-area or a systemwide basis.

The early development studies that paralleled the systems studies followed No. 1 ESS principles closely, extending them to define a multiprocessor structure controlling a multiplicity of suboffice four-wire ferreed switching networks. It was estimated that a set of six No. 1 ESS-type processors would handle 200,000 call attempts per hour providing an electronic toll office of at least twice the capacity of the No. 4A crossbar system. The price was estimated to be well below the No. 4A crossbar price at comparable traffic sizes.

Information derived from the study models, taking into account the development data, led to the conclusion that a good and reasonable capacity objective for a new toll system would be a design that could grow to two to three times the size of No. 4A crossbar. Greater size would be useful in a few areas of the country, but the major benefits would be achieved at the three times 4A level. Overall Bell System savings accruing from a system of the projected size and estimated cost were very large and justified a quick approval of standard development.

## **II. EVOLUTION OF REQUIREMENTS AND SYSTEM PLAN**

During the early planning period it was assumed that service features of a new system would, at least in an initial development, be an extension and improvement of the toll features of No. 4A crossbar, that maintenance and administrative features would be enhanced, and that the new CCIS (Common Channel Interoffice Signaling)<sup>2</sup> system would be incorporated in the design. When approval was given to start actual devel-

opment (1968), work on the elaboration and refinement of requirements was stepped up to match the needs of the development program.

In any large project, particularly when it is based on a new technology such as electronic switching, it is neither possible nor desirable to formulate final objectives and requirements independent of design and development.

If original objectives are sound and challenging, it is likely that technical breakthroughs will occur during the course of development and that objectives can be broadened and requirements can be extended. This was particularly true in the No. 4 ESS project.

There were three major technical advances that appeared early in the program which profoundly affected the course of development. These were:

(i) A new central processor design<sup>3</sup> of such high capacity that system objectives could be met (and eventually exceeded) with a system architecture based on a single processor pair instead of the multiprocessor arrangement originally envisaged. This carried with it a basic design technique and methodology which came to be known as 1A Technology<sup>4</sup> and which eventually permeated the entire system. The processor, designated the 1A Processor, was equally applicable in a larger version of the local No. 1 ESS.

(ii) A time-division switching network of such high capacity, small size, and basic economy that it displaced the ferreed network that was initially planned.<sup>5</sup> A major advantage of this network, which made extensive use of 1A Technology, was that it matched the accelerating trend toward use of digital transmission facilities.<sup>14</sup> As a result, it became possible to design No. 4 ESS as an integrated switching-transmission system with hitherto separate functional organizations cooperating in the development. This new interface opened the way to many new opportunities for improved efficiency and economy.

(iii) Enhanced trunk maintenance and record-keeping capability based on a minicomputer. An auxiliary facility known as the circuit maintenance system,<sup>6</sup> together with a related new trunk test position, made possible a much more efficient interface between the craft force and the trunk test and maintenance operations. The system incorporates mechanized record keeping to handle the large amount of continually changing office information related to trunk and facility interconnections.

As the system plan of No. 4 ESS evolved to take advantage of the concepts outlined above, it became possible to extend the system objectives and formulate system requirements in a new framework. Most important, the capacity objectives could now be converted to requirements and explicitly stated as three times 4A. Furthermore, it soon be-

Table II — Major service features

|   | No. 4A<br>Crossbar  | No. 4 ESS<br>Objectives |
|---|---------------------|-------------------------|
| Capacity                                    |                     |                         |
| Call attempts/hour*                         | 116,000             | 350,000                 |
| Network load, erlangs†                      | 6,200               | 28,000                  |
| Trunk terminations                          | 22,400 <sup>§</sup> | 110,000                 |
| Digit capacity                              | 14                  | 14                      |
| Translation (electrically alterable)—digits | 3 or 6              | 1 to 10                 |
| Code conversion                             |                     |                         |
| Digit deletion                              | 3 or 6              | 1 to 10                 |
| Digit prefixing                             | 1 to 3              | 1 to 10                 |
| Automatic alternate routes                  | 7                   | 13                      |
| Class marks—fixed                           | 16                  | 1/trunk group           |
| Common Channel Interoffice Signaling        | ✓                   | ✓                       |
| International gateway                       | ✓                   | ✓#                      |
| CCITT No. 6 signaling                       |                     | ✓#                      |
| Centralized Automatic Message Accounting    | ✓                   | ✓                       |

\* Level for engineering the office on a "10 high day," busy hour average.

† Level for engineering at 0.5% first trial matching loss.

§ Typical mix of one-way and two-way trunks.

# Not to be available initially.

came possible to expand the network load and termination requirements to a higher level and to specify that the network should be designed to operate on an essentially nonblocking basis. The new capacity requirements and a feature list for No. 4 ESS are shown in Table II, in comparison with No. 4A crossbar.

It might be observed here that the capacity levels achieved in the final design were as follows:

|                       |                        |
|-----------------------|------------------------|
| Call attempts/hour    | 500,000 (550,000 peak) |
| Network load, erlangs | 47,200                 |
| Trunk terminations    | 107,520                |

It will be noted on Table II that, except for the large increase in capacity, the basic toll switching features of No. 4 ESS are essentially an extension of No. 4A crossbar features. Beyond these new capabilities, however, major improvements were planned for the new system in terms of cost, performance, ease of installation, maintenance and administration, network management features.

*Cost:* A primary objective was to keep the basic cost of the system below that of No. 4A crossbar in the 4A size range. Above single 4A size, large cost savings were inherent in the system because 4E incremental costs would be much lower than a second and third 4A. Reduced multimachine penalties also would generate first-cost savings.

*Performance:* Overall network objectives for fast call setup and dis-

connect (made feasible by CCIS) established rigid switching time requirements on No. 4 ESS. These resulted in substantial impact on the design. Objectives were also established for various types of irregularities, based on current system performance, surveys of customer reactions to the irregularity when it does occur, and the difficulty of implementation. In addition the current ESS objective of no more than 2 hours of office down time per 40 years was applied, and objectives were established for maximum incidence of system errors such as call cutoffs or loss of calls in process of being set up. These objectives were then apportioned to the system components involved and applied to the No. 4 ESS design.

*Installation:* A reduced installation interval was planned in terms of plug-connected frames and expanded factory testing.

*Maintenance, Engineering, and Administration:* With availability and cost of skilled manpower becoming a matter of increasing concern, recognition of the need to hold down the staff to operate the office led to numerous objectives to enhance the maintenance and administrative features. These include:

- People-oriented, error-resistant, and efficient personnel interfaces.

- Automatic generation of operations reports by the system itself.

- Modular equipment and data concepts to allow trunk rearrangements on a group (12) or digroup (24) basis.

- Easy-to-understand and use interactive commands to alter the data base for routing and other translation changes.

- Elimination of many of the paper records by storing them in the system.

- Use of plug-ins through the system to facilitate installation and maintenance.

- Provision of automatic error analysis techniques.

- Software-controlled job assignments.

*Network Management:*<sup>12</sup> In addition to the traditional network management features, No. 4 ESS was planned to provide improved control and surveillance functions. For example, one objective was to detect on a real-time basis hard-to-reach codes and take action to block calls to such codes. In addition to restricting traffic in overload situations, the system should also act to expand routing opportunities. These features are intended to cope with the increasingly complex network and to maintain the overall efficiency in presence of traffic surges that can occur in overloaded portions of the network.

In addition to the objectives in the above major feature areas, which are detailed in subsequent articles, there were a number of other objectives such as designing the system so it would facilitate the addition of future features. Also, in 1971, the objective was set to cut over the first



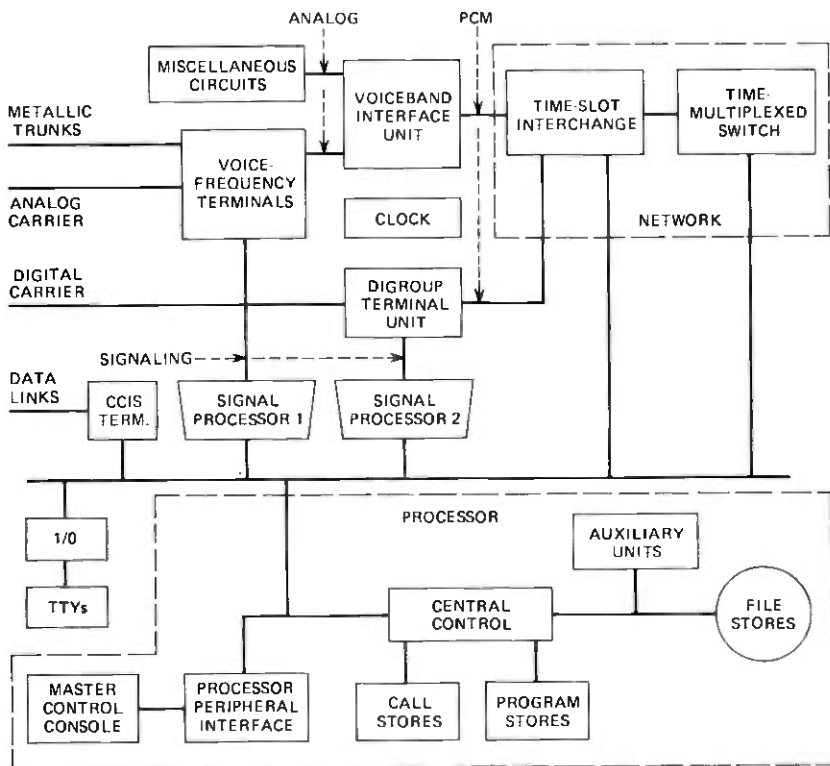


Fig. 1—No. 4 ESS office.

system over in the beginning of 1976. These objectives as well as the great majority of the other objectives that were set for the system, have been met.

### III. SYSTEM ORGANIZATION

#### 3.1 Outline of system plan

The general system organization is illustrated in the block diagram (Fig. 1). No. 4 ESS, like No. 1 ESS, uses a high-speed electronic central processor (1A Processor) operating with stored-program control to control the actions of the central office on a time-shared basis. However, although No. 4 ESS shares this basic concept with No. 1 ESS, there are a number of rather fundamental differences in the two system plans. The most significant of these is the use of the solid-state PCM time-division network. The decision to develop No. 4 ESS around a time-division switch was made early in 1970 after extensive studies and comparisons with other alternatives. Some of the major characteristics of this network are:

A time-space-space-space-space-time switching configuration organized in two types of frame: the *time-slot interchange* and the *time-multiplexed switch*.

High capacity, which, as mentioned above, not only meets but exceeds the system objectives.

Low blocking (virtually nonblocking), which eliminates need for trunk rearrangements for load balancing.

A T-carrier-compatible DS-120 input format—an 8.192 megabits per second PCM bit stream that accommodates 120 voice-frequency channels.

Small floorspace requirement due to use of the solid-state techniques.

Use of plug-in units and connectorization to ease installation and maintenance.

Solid-state technology (1A Technology) throughout.

In addition, a *clock*, accurate to 5 parts in  $10^{10}$  over a 1-week period, times the office and the digital lines that home on No. 4 ESS.

The DS-120-format digital inputs to the switching network are provided by two types of transmission equipment: *voiceband interface units* and *digroup terminal units*.

In the voiceband interface unit<sup>7</sup> the analog signals are sampled 8000 times per second and converted to standard ( $\mu$ 255 companded) 8-bit PCM frames. One hundred twenty channels are grouped together in a 128-time-slot format. The *voice-frequency terminal* units extract supervisory and dial pulse information and, conversely, convert the information they receive from *signal processor 1* to dial-pulse and supervisory information. Each voiceband interface frame includes seven active units and a spare (840 trunks).

In the digroup terminal units<sup>7</sup> five T1 carrier lines are combined to 120-channel, 128-time-slot PCM format, which is the same as the output of the voiceband interface units. The digroup terminal also extracts from the T-lines the necessary control information, passes it to *signal processor 2* and conversely converts the information it receives from signal processor 2 into the T-lines. Each digroup terminal frame contains eight active units and a suitable spare (960 channels). The signal processors perform a number of simple time-consuming functions and thus deload the main processor. These functions include routine scanning, collecting of dial-pulse digits, timing, etc. Signal processor 1, which interconnects with the voiceband interface frame, has individual scan and signal distribute points for the analog trunks. In signal processor 2, these individual points are not required, since the access is directly to the 120-channel bit stream. Both signal processors have additional scan and distributor points for administrative and maintenance functions. Signal

processor 1 handles up to 4096 analog trunks and signal processor 2 up to 3840 digital trunks.

The 1A Processor,<sup>1</sup> in conjunction with peripheral processors, with a 700-nanosecond central control cycle and a powerful order structure, allows the call capacity of No. 4 ESS to be achieved without multiprocessing. Another advantage of the 1A Processor, which plays a role very similar to the No. 1 Processor in No. 1 ESS, is electrically writable stores.

The *CCIS terminal*,<sup>2</sup> which, except for its bus access circuitry, is common to the terminal used in No. 4A crossbar, handles the routine and repetitive tasks associated with the CCIS (Common Channel Inter-office Signaling) data link.

A minicomputer-controlled Circuit Maintenance System (CMS) has its initial application in No. 4 ESS. CMS provides storage for records such as circuit layout record cards and assists in administration of work flow (e.g., by forming trouble tickets and test position work lists).

In addition to the above major new frames, the system includes some smaller developments, e.g., a sit-down *Master Control Console* (MCC), a 51A test position that replaces the 17-type test position, etc., as well as existing hardware, such as channel banks and test lines. All frames are 7 feet 2 inches high. Power distribution to the switching frames and the new transmission frames is at 140 volts; to the other frames it is still at 48 volts.<sup>13</sup>

#### IV. SOFTWARE

In No. 4 ESS most of the control, call handling, administrative, and maintenance functions are provided by the stored program.<sup>8-10</sup> The initial program (so called 4E0 generic) was the largest program yet developed for Bell System electronic switching systems. It consisted of about 400,000 instructions, 400,000 diagnostic test words, plus other miscellaneous control data, bringing the total to about 1.4 million 26-bit words. With the features added by the 4E1 generic, the program grew by 45,000 instructions and 70,000 diagnostic test words. The primary objectives of the design were:

- Real-time efficiency
- Simple personnel interface
- Defensive design
- Ease of modification

The design of the software system represented a major challenge not only because of its sheer size but (i) because it was developed in parallel with the hardware components of the system and (ii) because entirely new and difficult problems had to be solved in the design of the recovery

software for the network, signal processor, and the transmission interface frames (voiceband interface and digroup terminal). Despite these difficulties, a good-quality program was developed and completed on schedule. Some of the major characteristics of the program system are:

High real-time efficiency (as indicated above, the design not only met but substantially exceeded the capacity objective)

Elimination of timed interrupts to schedule high-priority work

A structure that allows many of the display and control pages to be designed independently of the generic

Maintenance control that allows up to three diagnostics to be executed in parallel

Automatic memory administration by the system

Software-generated test calls

## V. STATUS

The first four No. 4 ESS offices were installed in 1976 and are now providing excellent service.<sup>11</sup> Seven more offices are scheduled for service, including the Rego Park office in New York. This office will be the first to go into service with the 4E2 generic, which will provide local tandem features, the maintenance software for a new digital echo-suppressor frame, and many enhancements to the administrative and maintenance features. Work is also already underway on the "gateway" features to provide international service. This generic, called 4E3, will first see service in the Broadway 24 office in New York in the second quarter of 1978. Beyond these committed developments, other work is underway to achieve an even lower-cost system and to provide additional features.

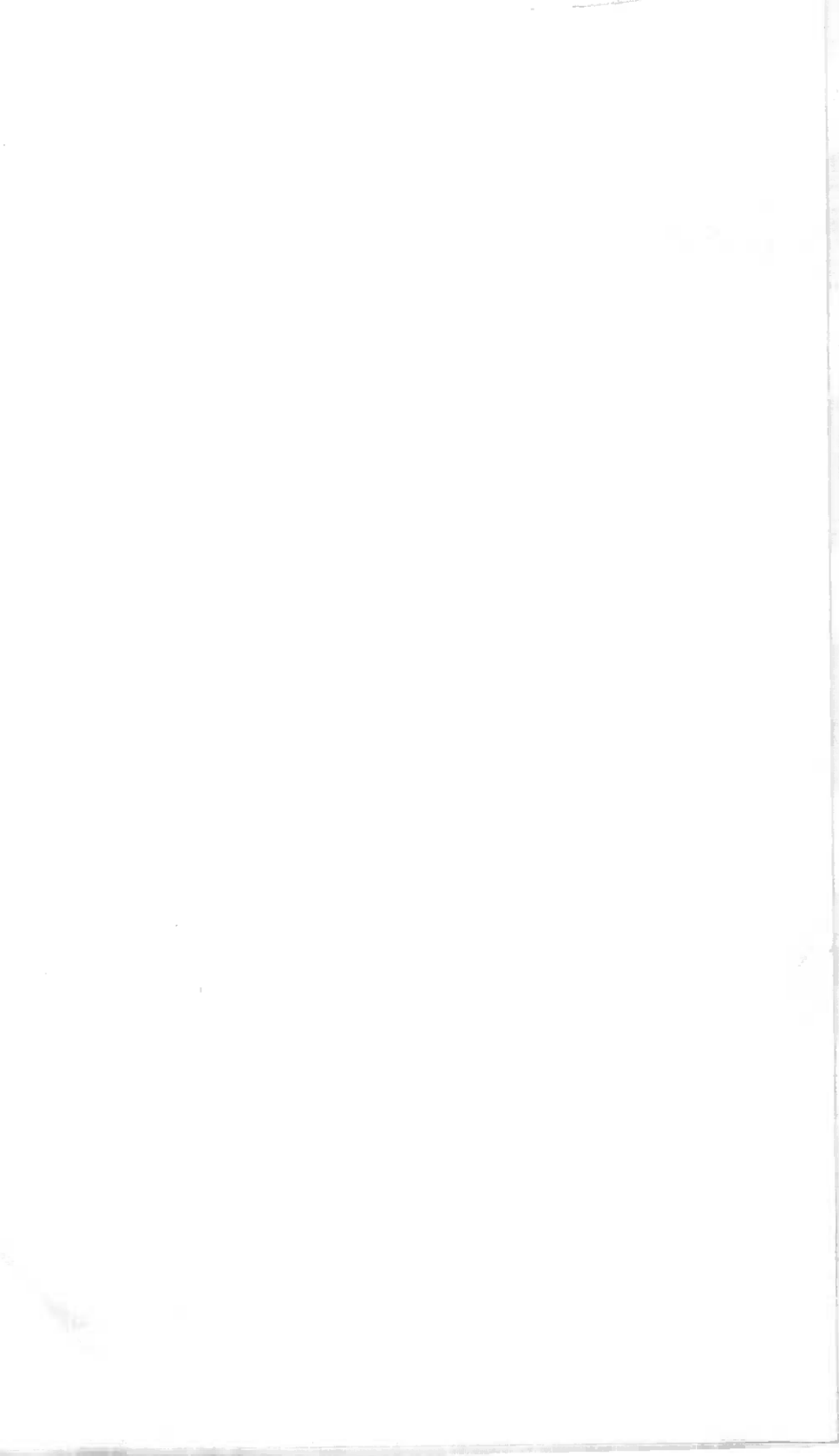
## VI. SUMMARY

This paper has presented a general outline of the objectives and organization of No. 4 ESS and thus serves as an introduction to the papers that follow.

## REFERENCES

1. Special B.S.T.J. issue on No. 1 ESS, September, 1964.
2. Special B.S.T.J. issue on CCIS, to be published.
3. R. E. Staehler, "1A Processor: Organization and Objectives," B.S.T.J., special issue on 1A Processor, 56, No. 2 (February 1977), pp. 119-134.
4. J. O. Becker, et al., "1A Processor: Technology and Physical Design," B.S.T.J., special issue on 1A Processor, 56, No. 2 (February 1977), pp. 207-236.
5. J. H. Huttenhoff et al., "Peripheral System," B.S.T.J., this issue, pp. 1029-1056.
6. J. A. Giunta et al., "Data/Trunk Administration and Maintenance," B.S.T.J., this issue, pp. 1203-1238.
7. J. F. Boyle et al., "Transmission/Switching Interfaces and Toll Terminal Equipment," B.S.T.J., this issue, pp. 1057-1098.

8. T. J. Cieslak et al., "Software Organization and Basic Call Handling," B.S.T.J., this issue, pp. 1113-1138.
9. M. N. Myers et al., "Maintenance Software," B.S.T.J., this issue, pp. 1139-1168.
10. G. A. Sellers et al., "Program Administration, Test, and Evaluation," B.S.T.J., this issue, pp. 1239-1278.
11. H. B. Compton et al., "System Integration and Early Office Planning," B.S.T.J., this issue, pp. 1279-1296.
12. T. V. Greene et al., "Network Management and Traffic Administration," B.S.T.J., this issue, pp. 1169-1202.
13. H. J. Luer et al., "System Power," B.S.T.J., this issue, pp. 1099-1112.
14. W. L. Ross et al., "Switched Digital Network Plan," B.S.T.J., this issue, pp. 1297-1320.



## No. 4 ESS:

# Peripheral System

By J. H. HUTTENHOFF, J. JANIK, JR., G. D. JOHNSON,  
W. R. SCHLEICHER, M. F. SLANA, and F. H. TENDICK, JR.

(Manuscript received July 8, 1976)

*The No. 4 ESS peripheral system economically performs essential switching and signaling functions. A modern time-division switching network has been developed to meet rapidly growing traffic demands and provide a basis for the all-electronic telecommunications network of the future. Common-channel interoffice signaling (CCIS) provides high-speed interprocessor communication via data link. Signal processor units perform routine signaling tasks such as scanning and digit collection, effectively increasing the main processor's capacity. To a greater degree than in previous systems, the vital man-machine interfaces are enhanced with sophisticated interactive capability.*

## I. INTRODUCTION

Electronic switching systems, in modern terminology, are composed of two parts, the central processing unit and the peripheral system. In No. 4 ESS, the central processing function is performed by a new high-speed integrated circuit unit, the 1A Processor, described in a special issue of the B.S.T.J.<sup>1</sup> The essential interconnection, signaling and interface functions of the No. 4 ESS peripheral system are the subject of this paper.

The switching network developed for No. 4 ESS features a new PCM time-division architecture that effectively integrates modern switching and transmission technologies. Interoffice trunks are handled via a multiplexed format, with no need to derive individual channels, and with no need for much of the per-channel equipment required in previous toll switching machines. Significant economies in administration and maintenance also result from this configuration. The multistage time-space-time topology chosen enables a large, high-traffic-occupancy

switching network to be constructed, as required in large metropolitan area switching centers. The planning, design, and operation of this new switching network is detailed in Section II.

Several special interoffice signaling equipment units have been developed for No. 4 ESS. These recognize and process information that identifies the address or destination of a call or any change of status of the call. A new Common Channel Interoffice Signaling (CCIS) unit is being introduced, permitting direct intercommunication via a data link between modern stored-program controlled switching offices. In addition, two new signal processor units, SP1 and SP2, are being provided with No. 4 ESS to relieve the 1A Processor of many repetitive time-consuming tasks such as counting dial pulses, timing critical intervals, or assembling several multifrequency signaling digits for presentation to subsequent address-determining programs. SP1 is designed with a multiplicity of scan and signal distributor points for sensing and controlling up to 4080 analog trunks. SP2 is designed to interface effectively with the bit-stream signaling used in digital carrier systems. A description of these signaling interfaces is given in Section III.

Section IV explains the bus system used to exchange orders and information between the 1A Processor and the multiplicity of peripheral equipment units.

Much effort went into the planning and development of the man/machine interface facilities provided in No. 4 ESS. Specialized operating and maintenance centers located outside the equipment rooms are equipped with sit-down consoles and with interactive Cathode-Ray Tube (CRT) displays having access to very large information data bases. These man/machine interface features are outlined in Section V.

Development of No. 4 ESS could not have occurred without the almost concurrent development of a family of highly reliable, microscopic-size, silicon integrated circuits, which permitted the design of new switching frames with over 100,000 logic gates. Section VI covers the equipment design philosophy used in No. 4 ESS.

The following sections of this paper provide a more detailed explanation of the No. 4 ESS peripheral system.

## **II. SWITCHING NETWORK**

### **2.1 *Principal characteristics***

The switching network is the center of interest in the overall planning and design of a new toll electronic switching system, and has important operational and economic influences on the future of the long-distance telephone business.

Long-distance telephone usage is expected to increase about 150 percent in 10 years, and to continue this trend into the '90s. Much of this



increased usage will originate in large metropolitan centers. This rapidly growing, concentrated traffic demands a new high-capacity toll switching system with a switching network much larger than present units.

Conventional space-division switching networks tend to become less efficient and more expensive as their size increases. The goal of No. 4 ESS is to be cost-competitive with existing systems while also meeting the large size requirement. This goal clearly indicated the need for a nonconventional approach to the design of a new toll switching network.

Only after several years of study and planning was the development of the final network begun. Starting in 1968, Bell Laboratories conducted an extensive evaluation of several network types that appeared to be suitable for modern message switching service. Space-division networks employing sealed metallic crosspoints such as the ferreed were studied, as well as those with solid-state PNP crosspoints. Delta modulation was considered as a means to overcome the variable attenuation of the solid-state crosspoints. Time-division networks were extensively evaluated, especially PCM digital hybrid types employing time switching and single- and multiple-stage space switching configurations. Logic designs and feasibility models were completed and cost estimates were made.

The various network types were compared according to the following criteria:

(i) The network should be at least three times larger than that of the present 4A crossbar system.

(ii) The network should be substantially nonblocking to eliminate load-balancing and minimize administrative effort.

(iii) The network should function efficiently in the rapidly growing digital transmission environment, as well as with conventional analog facilities.

(iv) The network must be cost-effective, both in terms of first cost and in terms of annual charges for maintenance and administration.

(v) The network must operate harmoniously and efficiently with the stored program processor.

As a result of this extensive evaluation and other system studies, the hybrid time-space-time network shown in Fig. 1 was selected for use in No. 4 ESS.

This network was chosen because:

(i) It satisfies all requirements.

(ii) Transmission plant is moving toward increased use of digital facilities.

(iii) Installation effort and floor space are reduced.



Fig. 1—No. 4 ESS network.

(iv) The digital interface leads to future cost reductions in terminal equipment.

The time-switching functions take place in time-slot interchange (TSI) units and the space-division switching functions take place in time-multiplexed-switch (TMS) units. A full-size network comprising 64 TSI and 8 TMS frames can handle over 107,000 four-wire trunks and service circuits. This is a folded, single-sided configuration where any network port can be connected to any incoming, outgoing, or two-way trunk. No double connections are required for two-way trunks.

The 1A Processor resident software controls the network as shown in Fig. 2. This software has the responsibility for path selection, for path setup and takedown, and for administering the map which lists the busy-idle states of all time slots and network links.

The network contains independent memory units which, by repetitively cycling 128 times per PCM frame interval, maintain all speech paths once they are set up by the processor. These network memories duplicate the path linkages contained in the processor's map. Network connections are maintained autonomously, even with momentary processor outages. The hardware-software interface and the path selection process were carefully designed to minimize processor usage.

A network traffic simulator was developed to determine the effect of various path-hunt algorithms on network blocking and to determine the processor efficiency in selecting paths. The simulator modeled a full-size network. Shown in Fig. 3 are the results of the simulation. The figure

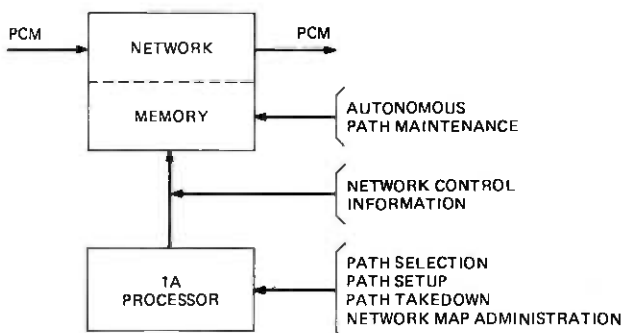


Fig. 2—No. 4 ESS network control.

of timing pulses, which are distributed to the TMS and TSI frames and through their clocking circuits to the transmission and toll terminal equipment for further use. In addition, the network clock frame contains a system clock unit providing timing signals readable by software in the 1A Processor, as well as time-of-day at the master control console.

The network clock (NCLK) contains four 16.384-MHz quartz crystal oscillators, each having a long-term stability of 1 part in 10 billion per day and a lifetime stability of 1.6 parts in 100 million. The oscillators normally run in a mode wherein one oscillator is designated as master, with the remaining three oscillators operating in the slave mode, phase locked to the master oscillator. Although normally arranged in an established hierarchy with the 00 clock chain oscillator designated as the master, any oscillator can be master and the clock can operate with as little as one good oscillator. The master oscillator selects control, and the monitor circuit in each chain automatically determines from error inputs whether its associated oscillator is healthy or defective, and thus determines whether the oscillator in chain 00 should be switched to chain 01, 10, or 11. In general, the highest healthy oscillator in the hierarchy is selected as master to the other oscillators, which are then phase-locked to it.

The phase-lock circuits detect and interpret phase and signal-level errors to ensure that the oscillator output is within  $\pm 5$  degrees deviation of the master. Pulse-shaping and frame-pulse generation circuits serve to combine the 16.384-MHz signal from one clock chain with that of the other clock chain in the bay, and shape the resultant signal into a symmetrical square wave. In addition, a counter in the frame pulse generation circuit counts the pulses and generates a signal every  $\frac{1}{8000}$  second to eliminate the 2048th pulse from the outgoing pulse train, thus defining the 8-KHz PCM frame pulse to the TMS and TSI. The clock cable drivers distribute the duplicated clock signal to TSI and TMS units over predetermined lengths of coaxial cable.

### **2.3.2 Time-multiplexed switch frames**

The TMS frame (Fig. 6) is organized as a simplex two-stage  $256 \times 256$  switch array, with each stage composed of  $16 \times 16$  arrays of crosspoint logic gates. Each TMS is contained in a 7-foot high, 4-foot 4-inch wide two-bay frame and consists of two peripheral bus units, a controller and bus interface unit, a control unit, eight-switch units, and miscellaneous power and filter units.

The TMS frame provides no duplication of switches within the frame. Each TMS frame has a mate frame which performs the same switching function. If either frame should fail, its mate performs the switching function without loss of calls. There may be one, two, or four pairs of

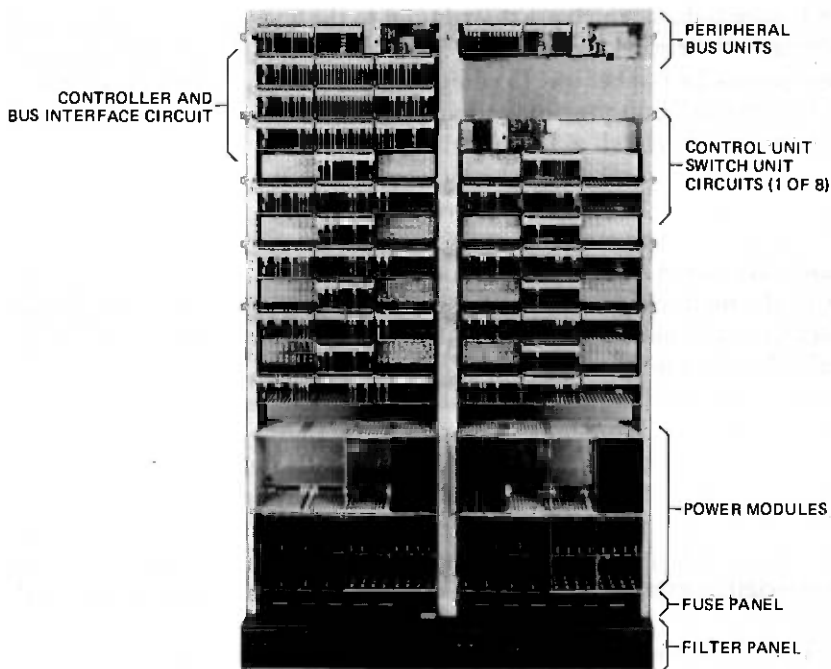


Fig. 6—Time-multiplexed switch frame.

mated TMS frames in a No. 4 ESS office. Moreover, circuit packs may be added to the switch units in four discrete steps so that the individual frames can grow in capacity.

In a full size No. 4 ESS office, four pairs of TMS frames are provided in a duplicated two-stage 1024 input by 1024-output switch array, providing 1024 unidirectional paths between input and output in each time slot. Each connection is capable of providing a path for an 8-bit PCM sample of data (representing a voiceband signal) in each time-slot. Each network connection through the TMS frames is made in terms of a pair of unidirectional paths in one of the 128 time slots, sharing the paths on a repeating basis at an 8-kHz rate. A total of 135,000 network paths ( $1024 \times 128$ ) are possible in a four-TMS-pair office, with each path being set up cyclically 8000 times per second.

The PCM samples arrive as part of a pulse train from the TSI frame over coaxial cables, where the samples from any one conversation, arriving once every 125  $\mu$ sec, are interleaved in 128 time slots with the samples from other conversations, and enter the switch units where cable receivers reconstitute the signals and translate them to 1A logic levels. The signals are then passed through the appropriate crosspoint gates in the two stages of TMS switching, and are presented to coaxial cable

drivers where they are amplified and transmitted to the other TSI frames over coaxial cables.

The switches are controlled by information contained in time-slot memories. The information for the connections is placed in these memories by the call-processing programs, which (via instructions over the peripheral unit buses) direct the frames to set up the paths connecting the incoming and outgoing calls. This path information is decoded in the controller, which then inserts the connection information into the time-slot memories where it is read out to provide a path through the network in a given time slot. The memories are accessed from the controller for a directed read or write for call-handling purposes in one half of each time slot, and are read in the second half of each time slot, under control of a counter synchronized to the network clock, to provide the path information for the next time slot.

The TMS does no retiming of the PCM data; rather, the TMS simply closes the data path and allows the data to pass through. This requires precise control of delay, which is provided by accurately cut coaxial cables between TSI and TMS frames and between NCLK and TSI/TMS frames.

The duplication existing between the paired TMS frames is further accounted for by a matching interface, provided between mated pairs of TMS frames in order to verify that the two frames are operating in step and also to further verify that the frames are operating properly. A mismatch of control data between the two frames causes an error indicator to be set in the frame detecting the mismatch.

### ***2.3.3 Time-slot interchange frames***

The Time-Slot Interchange (TSI) frames provide the initial time-space (T-S) and final space-time (S-T) switching stages of the No. 4 ESS time-division network. The TSI frames accept incoming PCM samples from the analog and digital facilities (voiceband interface frames and digroup terminals) over coaxial cables which carry the signals in a DS-120 format, wherein 120 8-bit PCM channels are time-multiplexed in a 128 time-slot, 16-bit per time slot, 16.384-MHz channel. The eight time slots not used for PCM channels are used for maintenance functions. A TSI frame interfaces with 14 DS-120 links, and thus is capable of handling 1680 trunks.

The receiving portion of the TSI buffers the incoming DS-120 links to allow synchronization of the incoming data with the network timing, then decodes and decorrelates the data, and performs the initial T-S switching prior to transmitting the data samples to the TMS frame.

After passing through the TMS frame, the data return to the transmitting portion of the same or another TSI, where the final S-T switching is performed on the data. After passing through these stages, the

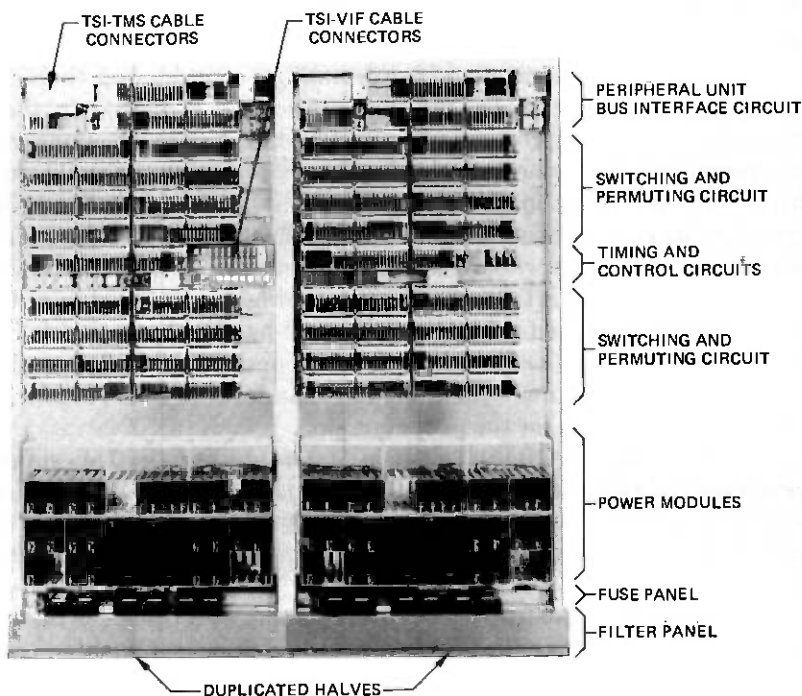


Fig. 7—Time-slot interchange frame.

transmitting portion of the TSI then recorrelates and reloads the data into outgoing DS-120 links, where the data are then transmitted to the appropriate analog and digital facilities.

The TSI frame is a 7-foot-high, 6-foot 6-inch wide frame, which contains duplicated peripheral bus interface units, duplicated timing and control units, two duplicated switching and permuting circuits, and power and miscellaneous units (Fig. 7).

The TSI frame is controlled by programs stored in the 1A Processor; the peripheral unit bus provides the means by which the 1A Processor accesses and controls the TSI. Thus, the peripheral bus interface unit sends information to, and receives information from, the peripheral unit bus system, and gives these instructions to and receives timing from, the timing and control unit.

The timing and control unit performs a number of important functions which control and sequence the operation of the TSI frame, and particularly the switching and permuting circuits. The timing and control unit receives a 16.384 MHz signal from the network clock frame, and uses this signal to drive a counter whose phases are decoded to provide timing and addressing for all autonomous operations performed within the TSI

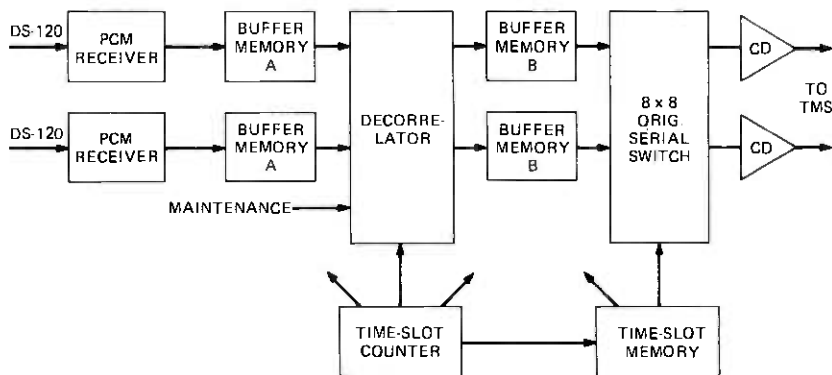


Fig. 8—Switching and permuting circuit (receive).

frame. In addition, the 16.384 MHz clock is also converted to an 8.192 MHz signal and transmitted, via a DS-120 link, to the transmission facilities connected to that TSI frame. The timing and control unit also provides decoding and sequencing for control and maintenance orders received via the peripheral unit bus, and records occurrences of autonomous errors detected in the frame for subsequent maintenance and diagnostic purposes. In addition, the unit provides control access (read/write) to the time-slot memories and busy/idle map memories in the switching and permuting circuits. Finally, matching of data between the duplicated timing and control units is provided to perform a vital diagnostic capability within the TSI frame.

Data arrive at the receive side of the switching and permuting circuit (Fig. 8), via the serial DS-120 link. This terminates on the PCM receiver, where the data (8-bit PCM) samples are converted to a 9-bit parallel form by addition of a parity bit, and placed in a first buffer memory (buffer memory A), at the address determined by the word count of the incoming PCM data. This provides a retiming function which allows independent operation of the transmission facility and the switching network. The reading of buffer memory A, as with all subsequent functions performed on the data in the TSI frame, is under the control of the time-slot counter in the frame.

The next operation is the performance of the deloading and decorrelation of the incoming data streams. For this operation, the data are read from buffer memory A in synchronism, transferred across the decorrelator in parallel format, and written into buffer memory B. The location from which the data are read from buffer memory A, and the location into which the data are written into buffer memory B is the count of the time-slot counter—i.e., at time slot  $x$ , words are read from buffer memory A location  $x$  and written into buffer memory B location  $x$ . The decorrelator may be thought of as an  $8 \times 8$  switch with a fixed algorithm wiring

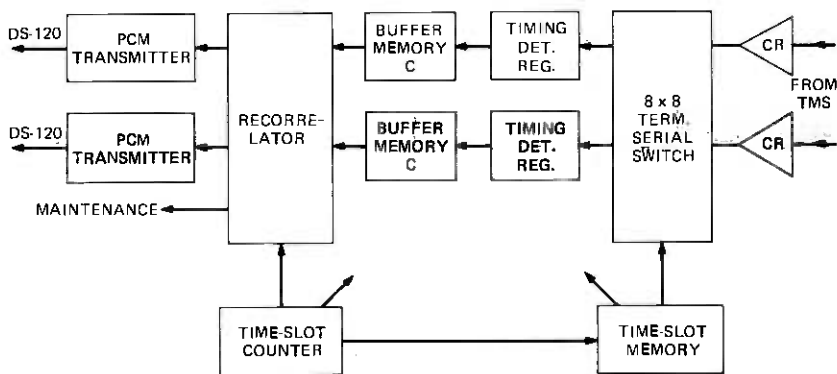


Fig. 9—Switching and permuting circuit (transmit).

pattern. Seven DS-120 links terminate on each switching and permuting circuit. The eighth input to the decorrelator is used only for maintenance and diagnostic purposes. Eight outputs from the decorrelator are available for access to eight buffer memory B's.

Operationally, the decorrelator spreads the traffic incoming on each DS-120 link equally over the eight buffer memory B's, so that each buffer memory B will store one-eighth of the samples from each input. Since there are seven inputs, each with 120 trunks, each buffer memory B will contain the samples from seven-eighths of 120, or 105 trunks, and will contain only 15 trunks from each DS-120 input. This deloads the network input and minimizes any effects of correlation of trunk seizure and holding times such as might occur with a large trunk group (perhaps as large as 120 trunks) arriving on a single DS-120 link.

When it is determined that a new network connection is to be established, call-processing programs resident in the 1A Processor are entered which search for an appropriate idle path through the network. Information specifying the time slot to be dedicated to the new connection, as well as the space switch linkages, is sent to the TSI via the peripheral bus interface. Through the control unit, this information is interpreted and stored in the time-slot memory. As the time-slot counter advances, the contents of the time-slot memory are read out, one cell at a time. A group of these bits defines the address of the buffer memory B to be read in that particular time slot, thus time-switching the incoming PCM channel information. A second group of bits provides the information needed to control the  $8 \times 8$  space switch that directs the PCM word from buffer memory B to the TMS frame.

The PCM signal generally returns from the TMS to the transmit side of a different switching and permuting circuit from one in which it originated. As shown in Fig. 9, the signal is passed through the fourth stage of space switching—an  $8 \times 8$  time-shared switch—then through



a timing detector and register where moderate amounts of time slips between paths are accounted for, and into buffer memory C, where the second stage of time-switching is performed. A corresponding time-slot memory associated with buffer memory C contains the data which selects the path through the  $8 \times 8$  switch and the information for the time switch in a manner analogous to that discussed previously.

The block diagram (Fig. 9) shows the transmit portion of the switching and permuting circuit. After the data are stored in buffer memory C, the data are read out synchronously by the time-slot counter, passed through a recorrelation switch which performs an inverse mapping of the decorrelation function, and sent, via a transmitter (which conditions the signal into a DS-120 format), to the transmission facilities, voiceband interface frame or digroup terminal.

### III. SIGNALING INTERFACES

#### 3.1 Overview

Interoffice signaling is the information that identifies the address or destination of a particular call or change of status of a call. In No. 4 ESS, this signaling is monitored and controlled via the signal processors and CCIS terminal groups which are, in turn, controlled by the 1A Processor via the peripheral unit bus. There are two types of signal processor frames: Signal Processor 1 (SP1) which can handle signaling for up to 4080 analog trunks, and Signal Processor 2 (SP2), which can handle signaling for up to 3840 digital trunks. The signal processors can process both Dial Pulse (DP) and Multi Frequency (MF) signaling formats. Each CCIS terminal group frame can process common channel interoffice signaling for up to 24,000 trunks.

Much of the work done by a switching system in processing calls is associated with supervisory scanning and with the collection and transmission of the address digits. In No. 4 ESS, the signal processors and CCIS terminal groups autonomously execute these repetitive and time-consuming data processing functions performed by the central control in previous electronic switching systems. This eliminates the overhead that would exist if the data processing functions were done by program control in the 1A Processor, allowing a higher call-handling capacity. Call-processing programs resident in the 1A Processor maintain control over the overall handling of a call. Decisions as to how many digits are to be collected on MP or DP calls, what action to take on seizure or disconnect, etc., are determined there.

The CCIS signaling equipment for No. 4 ESS will be described in a later special B.S.T.J. issue on CCIS and will not be described in detail here. The signal processors will be presented by first giving a detailed description of the SP1 design. Since many parts of the SP2 design are

identical to SP1, the SP2 will then be presented by describing only those parts which are different from SP1.

### **3.2 Signal Processor 1**

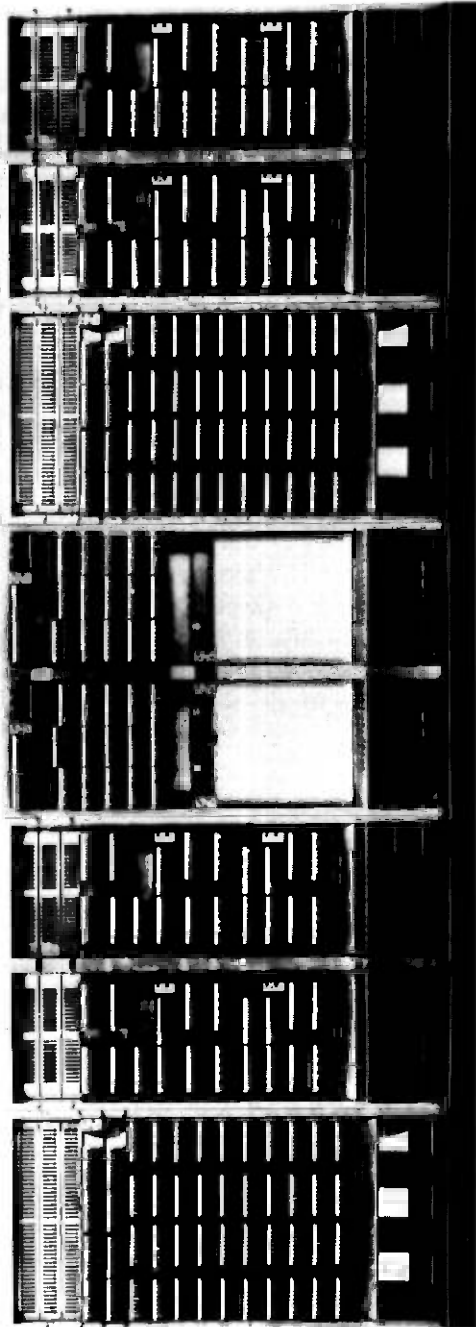
#### **3.2.1 Description of Signal Processor 1**

The SP1 is a duplicated, wired-logic processor which performs the various scan, distributing, and digit-reception tasks for analog trunk facilities. Furthermore, it is a directed processor under control of CC. In order to execute these tasks, the SP1 control interfaces internal matrix-access circuits and communicates with CC via the peripheral unit bus. Each SP1 has 4096 simplex scan and 4096 simplex distributor points for control of up to 4080 trunks. These are designated universal scan and distributor points and connect to the E and M leads, respectively, of the trunks in the Unitized Terminal Equipment (UTE) facilities and miscellaneous trunk frames. A scan point and distributor point are assigned to each trunk.

In addition 2048 scan and distributor points, designated miscellaneous points, are used for control of service circuits and miscellaneous circuits and alarms. Included in this category are MF transmitters and receivers. MF signaling equipment is not assigned to a single trunk but is instead shared by many trunks on an "as needed" basis. No. 4 ESS uses a new integrated circuit MF receiver and transmitter. Trunks requiring MF receivers or transmitters are connected to these units through the switching network for the time required to receive or transmit a call and they are then released for use by other trunks. Up to half of the miscellaneous distributor points can be pulse points, transformer coupled to supply 500-ns control pulses for peripheral control functions. All scan and signal distributor points provide dc isolation to prevent ground loops.

The SP1 interfaces with new and existing signaling equipment through its scan and distributor points. Autonomously every 10 ms the SP1 scans all 6144 scan points, stores changes of status and digits received in internal buffers, and performs nearly all timing functions on MF and DP reception and transmission. Upon command the SP1 also executes directed orders from the 1A Processor to empty the internal buffers containing collected digits, seizure reports, and disconnect reports, and to operate distributor points or read groups of 16 scan points. CC also can load internal work lists with digits to be transmitted to other offices. Once loaded, the digits are transmitted autonomously by the SP1.

In addition to handling signaling in a No. 4 ESS office, the SP1 monitors and controls miscellaneous circuits in an office such as alarm circuits and power switches. This provides the interface that allows CC to control the entire periphery. Two signal processors in each No. 4 ESS office are



DISTRIBUTOR  
APPLIQUE

DISTRIBUTOR  
AND  
SCANNER MATRIX

SIGNAL  
PROCESSOR  
CONTROL

DISTRIBUTOR  
APPLIQUE

DISTRIBUTOR  
AND  
SCANNER MATRIX

Fig. 10—No. 4 ESS signal processor.

designated as base SPs; they are initialized via CC pulse points. The base SPs in turn supply duplicated pulse points to initialize or "bootstrap" the remainder of the periphery on a per frame basis.

A photograph of the SP1 frame appears in Fig. 10. The right-hand matrix and applique are optional, and, within each matrix frame, growth is provided by groups of circuit packs for 1024 points. The matrix frames will be described first, and will be followed by the control units. The matrix frames provide the actual interface with the equipment to be controlled or monitored, while the control unit provides the actual data storage and processing.

### **3.2.2 Distributor and scanner matrix**

The distributor and scanner matrix frames are simplex units accessed by a duplicated control unit via duplicated access circuits. The access accepts a 9-bit address and reads out a row of 16 points. The distributor access contains additional circuitry so that a row of 16 flip-flops, storing the status of 16 distributor points, is selected and may be individually set or reset. The interface of the matrix with the duplicated access is an AND function so that only when the access circuits from both controllers agree is a row selected. A successful selection generates an ALL SEEMS WELL output. An ALL ZEROS TEST checks for stuck conditions, especially a stuck ALL SEEMS WELL. This prevents access-circuit faults from causing service disruptions.

### **3.2.3 Signal Processor 1 control**

The functions of the control can be divided into two classes—directed and autonomous. A directed function is the result of a PU bus order from CC and immediately results in some action and/or a response back to CC. Directed functions, for example, would include reading an output buffer, reading the present state of 16 scan or distributor points, writing up to 16 distribute points, and reading or writing data or a control memory word.

Autonomous functions are the repetitive and time-consuming functions performed by the SP1 that remove the heavy burden from CC. These functions are performed on a 10-ms cycle as shown below.

The 10-ms base-level cycle (Fig. 11) was selected to meet the timing precision of MF and DP outpulsing and still allow the SP1 to process a reasonable number of scan points each cycle. The first half of each cycle is devoted to MF outpulsing, universal scan, and DP digit reception, while the second half is devoted to MF outpulsing, miscellaneous scan, MF reception, and DP outpulsing. Note that MF timing must be performed every 5 ms or twice each cycle to meet both domestic and international signaling requirements.

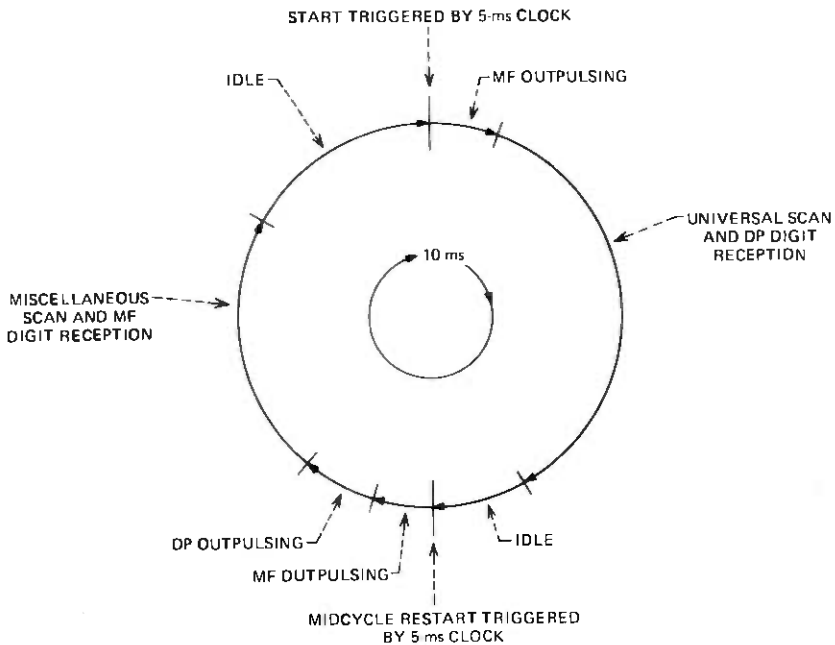


Fig. 11—SP base-level timing.

Figure 12 is a block diagram of the SP1 control, whose organization is similar to that of a stored program processor. There is a bus structure that provides access through the control unit for all the various subunits. The autonomous sequencers are the heart of the control. An executive sequencer is the highest-order sequencer and activates three autonomous task sequencers in the order and frequency specified by the 10-ms base cycle. It also performs sanity checks to ensure that each task has been

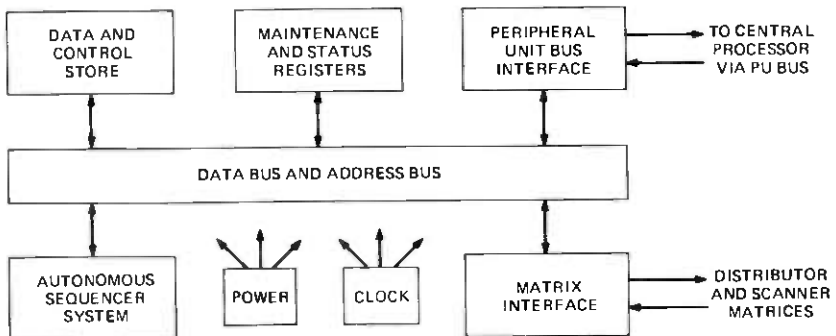


Fig. 12—Signal processor control.

completed the correct number of times. The scan and digit reception sequencer, one of the three task sequencers, performs both the universal and miscellaneous scan tasks and in addition performs both MF and DP reception. The other two task sequencers are the DP outputting sequencer and the MF outputting sequencer. Each task sequencer, when called, performs its task for all associated elements and returns control to the executive sequencer upon completion. The three task sequencers in turn may call in the matrix sequencer or memory sequencers which return control to the calling task sequencer upon completion.

The peripheral unit bus interface contains the cable drivers and receivers which interface the parallel data PU bus to the SP1 control. It listens for PU bus orders destined for the SP1 and carries them out on an interrupt basis. An internal clock supplies appropriate timing pulses. A countdown circuit provides synchronized 5-ms, 10-ms, and longer outputs for various purposes such as interdigital timing.

### **3.2.4 Data and control store**

The data and control store is used by each of the autonomous task sequencers for storage and retrieval of processing data, for storage of processing instructions from CC and for buffering reports to CC. The store logic structure is conventional except for insertion masking and address increment logic. Also, counters are used to keep addresses (pointers) for list entry points.

The store contains four buffers of varying lengths: the seizure, digit, high-priority and low-priority buffers. They store autonomously collected information destined to be sent to CC. Each buffer is a first-in, first-out list. The type of information put in each is indicated by its name, with the high-priority buffer generally containing answer and disconnect signals. The CC interrogates these buffers periodically and requests the information. The store also contains work lists used by the autonomous sequencers for DP and MF reception and outputting.

### **3.2.5 Trunk status bits**

Information for each trunk assigned to an SP1 is contained in its trunk status bits. The trunk status bits occupy three-quarters of the store and control the scan and digit reception sequencer on a per-trunk basis. Six bits of each word are permanently assigned to each of the scan points in the scan matrix. These bits are designated T1 through T6. The bits for four trunks are stored in one memory word. T1 is the state of the scan point at the last time the SP1 scanned the point. Thus the SP1 can recognize and report a change of state. T2 and T3 tell the SP1 what to do if there is a change of state. These bits are set by CC via a PU bus order. The report code specifies whether the SP is to ignore the change of state

or report changes in a buffer. For example, if the scan point is in the universal field and the change of state is to off-hook, the report will be put in the seizure buffer instead of the low-priority buffer. If the digit buffer is specified, the SP1 will create an entry on the dial-pulse reception work list and begin to report digits rather than individual state changes for that trunk.

T4 and T5 are used for hit timing if T6 has been set by CC. If the change of state persists for 30 to 40 ms, the change of state will be reported. This feature is used primarily to protect against false seizures.

A block of eight words is reserved for SP1 maintenance programs. Specific patterns are stored in these words at the time a unit is placed in service. Whenever an interrupt occurs, the words are accessed to provide a quick check of peripheral-access, internal-bus, and memory-access circuitry.

### **3.2.6 Maintenance and status registers**

Four register groups provide the means for controlling SP1 controller configuration and service status, recording failure information and reporting errors to the CC, and forcing special maintenance circuits which enable CC programs such as SP diagnostics to detect and resolve circuit faults. Error detection and maintenance circuitry is distributed throughout the SP1. The error-source register records the source of an error discovered by these circuits. The hardware checks in the simplex category are parity failure of store information, invalid sequencer states, DP work lists out of order, etc. Duplex checks are primarily matches between registers and sequencers of the duplicated control.

The action on detection of an error is to record the error and stop. At regular intervals, CC polls all units to locate any unit in trouble. Fault recognition programs then determine the faulty unit. Read and write access plus clock control via PU bus maintenance orders permit quick checks by fault-recognition programs, and intensive, accurate error location by maintenance programs.

## **3.3 Signal Processor 2**

### **3.3.1 Description of Signal Processor 2**

Each Signal Processor 2 (SP2) monitors and controls signaling for a total of 3840 digital trunks via four digroup terminals.<sup>2</sup> The SP2 performs data processing functions—scanning, digit reception and outpulsing, etc.—and interacts with the 1A Processor in the same manner as the SP1. Hence, the SP2 is similar to the SP1 with two principal exceptions. First, instead of physical scan and distributor point circuits for the universal points, the supervisory state of each trunk is stored as a bit in 512 words

of additional memory. Whenever the autonomous data processing functions need to read or write universal supervisory states, the SP2 matrix sequencer accesses the data memory. Second, the SP2 acts as the communication link to provide, in effect, an extension of the peripheral unit bus to the digroup terminals for maintenance and control messages.

### ***3.3.2 Supervisory data transfer and timing***

The incoming supervisory state of each trunk is extracted from the T-carrier line facility by the digroup terminal (DT). Upon request by the SP2, the data is sent to the SP2 on a serial data link to be stored in the SP2 data memory. Similarly, and simultaneously, outgoing supervisory data is sent from the SP2 to the DT to be injected into the outgoing T-carrier data stream. The timing of the data transfer is such that the average delay is less than 3 ms, which means that the overall delay with the SP2 is less than with the SP1 because there is essentially no delay in the DT compared to relay operate delay in the metallic trunks and the delay of single frequency sets on the analog trunks associated with an SP1.

### ***3.3.3 DT control and maintenance messages***

The SP2 acts as a communication link between the DTs and the peripheral unit bus. Program commands for a DT from the 1A Processor, such as protection switch or error source readout requests, are sent on the peripheral unit bus to the SP2 and temporarily stored in memory. At the start of data transfer to the intended DT, the SP2 first transmits the message and then starts the data transfer. Similarly, a message from the DT is transmitted to the SP2 and placed temporarily in memory.

When signaling for digital trunks is via CCIS, the DT acts as a transmission multiplex and terminal without supervisory signaling. For this situation the SP2 acts as a control and maintenance message link to the peripheral unit bus for up to 12 additional DTs.

### ***3.3.4 Error detection and redundancy***

The DT and the SP2 contain error-detection and configuration circuits and both have duplicated control circuits. Normally DT controller zero is configured to SP2 controller zero and similarly with controller one. In this mode there is continuous matching in the SP2 of all information and sequencer action. In the event of a fault, either DT controller can be configured, by peripheral unit bus messages to the SP2, to either half or both halves of the SP2.

### ***3.3.5 Supplementary matrix***

In a largely digital office there still exists a need for miscellaneous physical scan and distribute points for such functions as recognition of



fuse alarms and for controlling multifrequency transmitters and receivers. A supplementary matrix frame is available as an option for the SP2. The supplementary matrix contains 1024 scan and 1024 distribute points arranged to correspond directly with the first 1024-point miscellaneous matrix of the SP1 scan and distribute matrix.

#### **IV. PERIPHERAL UNIT BUS SYSTEM**

The peripheral unit bus provides the control and data path between the 1A Processor and most peripheral frames of the No. 4 ESS. The bus consists of four duplicated bus groups: the enable address bus, the write bus, the reply bus, and the control bus. The enable address and the write buses convey instructions and information in parallel form from CC to the peripheral units, while data is sent from a peripheral unit to CC via the reply bus, also in parallel form. Control and maintenance information is transmitted to and from the peripheral units over the control bus.

Because of the limitations of the length of the bus and the numbers of frames that could be connected to a single bus, a Peripheral Unit Bus Branching (PUBB) frame is provided to extend the peripheral unit bus. The PUBB is a fully duplicated frame, with separate units for PU bus 0 and PU bus 1. It is growable to four units per bus with four bus branches per unit. Loop-around circuitry in each unit allows fault-recovery programs to check the integrity of each branch up to the output of the PUBB frame. Since all signals entering the PUBB are regenerated internally and transformer-coupled to the outputs, the PUBB also provides isolation of the periphery from the 1A Processor.

Outgoing signals pass through serial bus receivers in the peripheral units being serviced. Bypass resistors on the receiver circuit pack connectors assure bus continuity when a receiver pack is removed for repair. Shunt cable drivers in each peripheral unit send data back to CC. The last peripheral unit on a bus branch contains terminating resistors to preserve the transmission characteristics of the bus. Because all connections to the bus are transformer coupled, the bus provides complete ground potential isolation between units. Care is taken in engineering an office to assure that critical frames do not all appear on the same bus branch.

All No. 4 ESS peripheral units are alerted via coded enabling, whereby each unit has a unique name and listens to the bus at all times. When an order is sent over the peripheral bus, only the peripheral unit whose name matches the name accompanying the order will respond.

#### **V. MAN/MACHINE INTERFACES**

The craftsman has several facilities with which to interface No. 4 ESS. The major facilities are the Master Control Console (MCC), the

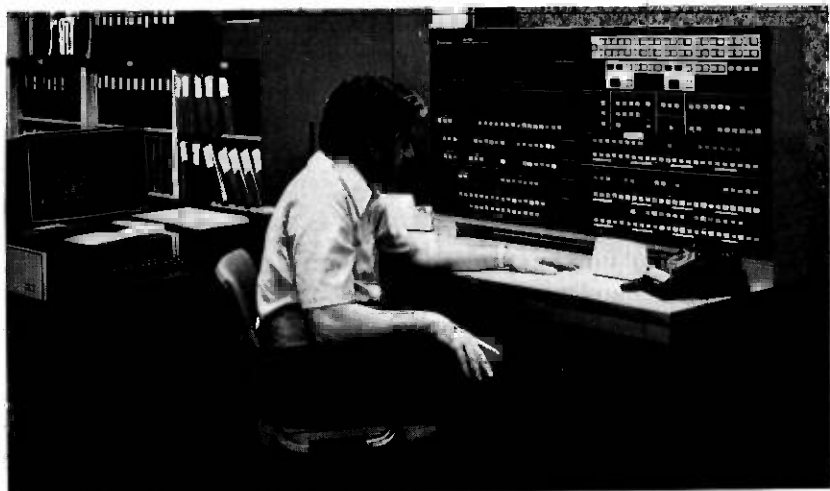


Fig. 13—Master control console.

*DATASPEED*® 40 terminals, the power control switches located on the peripheral units, and the office alarm system.

The Master Control Console (Fig. 13) is located in the Maintenance Operations Center (MOC) and consists of an alarm, control, and display console that serves as a direct communications link between the craftsperson and the system. It indicates by means of alarms and status indicators the current condition of the system. Controls are provided so that manual recovery of the system can be effected if the automatic recovery capability is unable to do so. The displays and controls have been designed to be as simple as possible and care has been taken to avoid any arrangement of controls and displays that might mislead or confuse the craftsperson. An equipment status panel on the MOC provides the craftsperson with the overall status of the system. Any unit out-of-service for any reason, including diagnostics, will activate an indicator lamp for that unit type. By depressing the key associated with the indicator, the craftsperson is furnished with a printout of the member numbers of all units of that type which are out-of-service.

The *DATASPEED* 40 terminal is the major man/machine interface between the craftsperson and the peripheral units. From the terminal it is possible to remove a unit from service, diagnose it, and restore it to service. This can be done via several I/O channels located throughout the No. 4 ESS office. The primary terminals used are those located in the MOC area. In addition, two beltline I/O channels are distributed to all peripheral units so that the craftsperson can interface with the system at the frame being repaired. Messages to and from the craftsperson are simple and concise. In the case of a circuit-pack replacement, the pack

type and frame location identified by diagnostics are given to the craftsperson.

The craftsperson also has limited control over a peripheral unit with the power control switches located on the peripheral units. Operation of the switches requests removal of a frame from service. An indicator lamp on the switch shows whether the request is being serviced and the state of the frame (out-of-service). Similarly, it is possible also to request restoral of a peripheral unit to service, which will automatically cause diagnostics to be run on the unit. If the unit passes diagnostics and is restored to service, all indicator lamps will be extinguished. Finally, manual power alarm tests on the unit power converters can be performed from the power control switch.

The craftsperson is alerted to failures in the system by the No. 4 ESS office alarm system. This newly designed system consists of a series of alarm grids, each monitoring a specific area of the periphery. Alarm outputs, both audio and visual, can be limited to only the area being monitored and its associated work center or can also be directed to other areas or work centers via software controls. Grids can also be joined into larger grids via software to accommodate a reduced work force at non-peak hours. The status of all grids is displayed in the MOC. Audible alarm outputs are included to indicate software alarms as well as hardware or power alarms.

## **VI. EQUIPMENT DESIGN**

Development of the No. 4 ESS peripheral equipment would not have been possible without dramatic advances in miniaturization of electronic circuitry. This rapid advance was based upon the development of Silicon Integrated Circuit (SIC) technology. The fact that a minute SIC chip may contain as much or more circuitry than was previously contained in an entire plug-in circuit pack has had a tremendous impact on the development of No. 4 ESS equipment, permitting the design of frames which contain over 100,000 electronic gates.

In order to produce the 1A Processor, with high-speed operation at low power levels, and with small signal swing and low noise margins, it was necessary to develop a hardware technology capable of meeting these needs. This is called 1A Technology hardware and is described in detail in another B.S.T.J. special issue.<sup>3</sup>

Since much of the circuitry in the No. 4 ESS peripheral area is composed of high-speed digital logic gates which are similar to the 1A Processor circuitry, it was decided that the No. 4 ESS equipment also would use the 1A Technology hardware to the greatest extent possible consistent with the requirement of the peripheral area. This permitted increased economies in manufacture due to greater quantities of similar

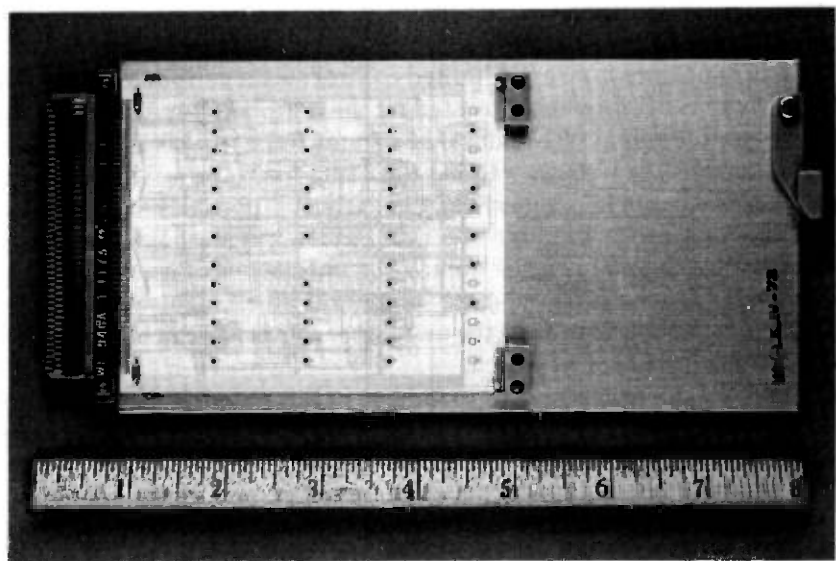


Fig. 14—Logic circuit pack.

items of manufacture, and it was not necessary to develop, design, and maintain two similar but different hardware systems.

Stated briefly, the 1A Technology hardware system is based upon a family of Collector Diffusion Isolation (CDI) chips. These CDI chips are all identical in external dimensions (0.050 inch by 0.050 inch over the tips of the beam leads) with each chip having 28 beam lead contacts. The family of codes consists of six general purpose codes and seven with specific functions. Up to 52 of these SIC chips may be mounted upon a large ( $3\frac{1}{4}$  inches by 4 inches) ceramic substrate and chips are interconnected with thin-film conductors. The standard substrate has predetermined locations for 52 SICs and 841 locations for arrays, each containing up to seven beam crossovers to facilitate wiring. The substrate is mounted upon an aluminum plate (7.31 inches by 3.67 inches by 0.062 inch), which acts as a heat sink and is fastened to an 82-pin connector. The bulk of the No. 4 ESS peripheral area circuitry has been built on 149 different codes of this type of circuit pack (Fig. 14).

Discrete circuit packs also were developed to cover that circuitry which was made up of discrete components or of combinations of SICs and discrete components. These circuit packs are packaged on conventional  $\frac{1}{16}$ -inch-thick epoxy glass double-sided wiring boards measuring 3.67 inches by 7 inches. Interconnections between sides are provided by plated-through holes. The SICs are accommodated on the discrete circuit packs by small Hybrid Integrated Circuits (HICs). A HIC consists of a small ceramic substrate which mounts one or more SICs along with its

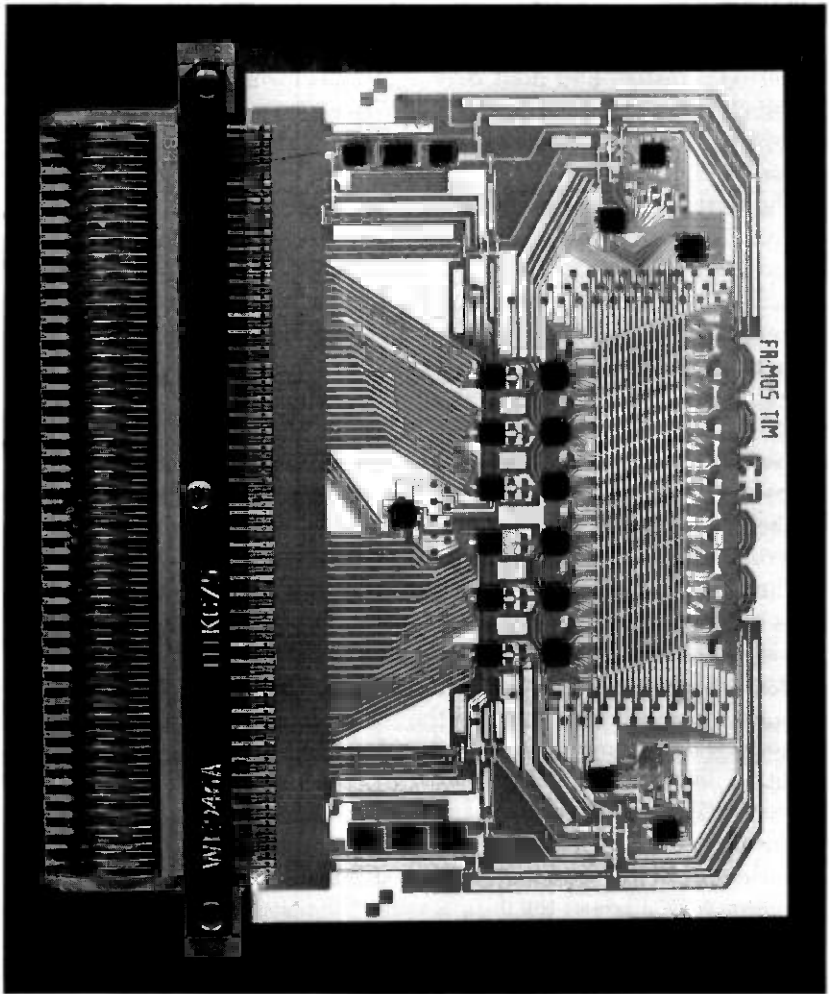


Fig. 15—Solid-state memory pack.

thin-film circuitry; in addition a HIC may contain thin-film resistors. The substrate then has a conventional lead frame added to it, which is used to mount the HIC to the epoxy glass board through holes in the board. Discrete components, which usually have axial leads, are mounted in the conventional fashion. The circuit pack can be equipped with either a 42-contact or an 82-contact plug connector. To date, 35 of those codes have been designed for use in the No. 4 ESS peripheral area.

As a special case, four solid-state memory packs have been developed to provide localized storage in the No. 4 ESS peripheral area. These circuit packs are organized as either 128- or 256-word by 10- or 12-bit arrays.

The memory function is provided by a silicon gate MOS chip which contains a 128-word by 2-bit array of static memory cells. Decoding, buffering, sensing, and digit driving are performed by bipolar chips. These chips are mounted upon a half-size ceramic substrate which is fastened to an aluminum heat sink and provided with a connector, as shown in Fig. 15.

The next level of assembly is the unit which is built up of a number of circuit packs assembled on a common mounting structure and interconnected at the backplane to perform one or more specific functions. Generally in a unit the circuit packs are located on  $\frac{1}{2}$ -inch horizontal centers or multiples thereof and 4-inch vertical centers. Occasionally  $\frac{3}{4}$ -inch horizontal spacings are used.

Interconnections between the circuit-pack connector terminals in a given unit are accomplished by two methods. Multilayer boards are used for power and ground connections and for some signal connections. Most of the signal connections, however, are made with 30-gauge wire, which is machine-wrapped. Electrically sensitive leads are applied manually, generally as tight twisted pairs or miniature coaxial cables.

The final level of packaging is the frame, which consists of two or more units mounted upon a common structural framework. Intraframe wiring consists of twisted pair or miniature coaxial cables either loosely run via guides or in cable harnesses. Almost all of the interframe cabling is connectorized. All of the cables that carry telephone messages in the high-speed time domain are coaxial cables, while the rest are switchboard cable.

From the initial conception of the frame design to the acceptance tests at the central office level, thermal considerations were of the utmost importance. In the early designs, analytical studies using computer programs were made of the temperature rise within the frame under the maximum central office ambient temperature. The first models of the frames were tested at elevated temperatures in the hardware laboratory. As the System Laboratories were completed, the most heat-sensitive frames were tested in a system environment. Finally, major portions of the first two central offices were tested under operating conditions at elevated temperatures. All of these tests have confirmed the soundness of the original thermal design.

## VII. SUMMARY

In this paper we have described the major items of the No. 4 ESS peripheral system, with particular emphasis on the switching network and the signaling interfaces. Much effort went into the planning and organization of these peripheral items to assure their compatibility with analog and digital transmission facilities and with existing and forward-looking signaling arrangements. To achieve this integration re-

quired the cooperation of many people in the transmission and switching organizations, as well as the gradual demolition of what once was a sacred boundary between the two disciplines.

#### REFERENCES

1. R. E. Staehler, "1A Processor: Organization and Objectives," B.S.T.J., special issue on 1A Processor, 56, No. 2, pp. 119-134.
2. J. F. Boyle et al., "Transmission/Switching Interfaces and Toll Terminal Equipment for No. 4 E.S.S.," B.S.T.J., this issue, pp. 1057-1098.
3. J. O. Becker et al., "1A Processor: Technology and Physical Design," B.S.T.J., special issue on 1A Processor, 56, No. 2, pp. 207-236.





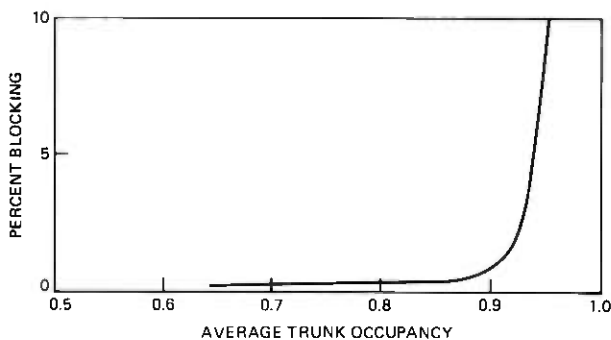


Fig. 3—No. 4 ESS network blocking.

represents the blocking curve of a full-size network. It shows the steep slope characteristic of highly efficient networks. The toll switching system requirements are:

0.5 percent blocking at 0.7 occupancy

10 percent blocking at 0.9 occupancy

The performance of the No. 4 ESS network is considerably better than these requirements.

Figure 4 shows the time (percentage of typical call setup time) used by the path-selection process. At high network occupancy, the usage amounts to about 10 percent. The efficiency of this process, as well as its stability at high loads, is an important factor in helping to achieve a very large call-handling capacity.

## 2.2 Topology

An initial decision was made to employ 1A Technology for all logic circuits in the time-division network. Based on the speed of these

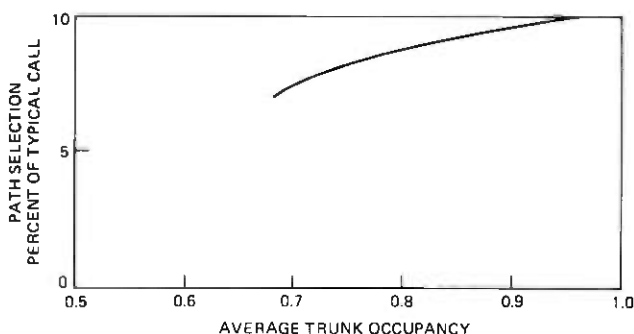


Fig. 4—No. 4 ESS network path-selection time.

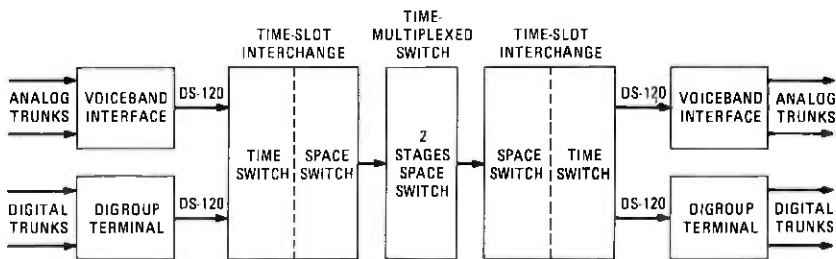


Fig. 5—No. 4 ESS time-division network topology.

semiconductor devices and on the characteristics of newly available solid-state memory units, another decision was made to operate the network at a rate of 128 time slots per PCM frame (125  $\mu$ sec).

One of the early network topologies featured a combined concentrator, time switch and PCM codec. This was augmented with a three-stage space switch employing high-speed logic-gate switching elements. Such a network design, when concentrating 140 trunks on 128 time slots, was expected to be capable of handling about 72,000 trunks, each operating at 0.7 occupancy.

As development proceeded, the need for a still larger network was recognized. It was also decided to separate the PCM codec and time-switch functions to permit future digital signal processing (such as echo suppression) independent of switching. This led to the final network topology shown in Fig. 5. In this arrangement, the four-stage space-switch functions are distributed in two separate frame types—a time-multiplexed switch frame, and a time-slot interchange frame. The time-switch functions reside in the time-slot interchange frames, and the PCM functions are contained in a voiceband interface frame, which performs a purely sequential encoding of 120 trunks onto a fixed-format data link. This allows the PCM function to be located near the trunks, and provides a universal interface data link between transmission and switching frames—the DS-120 data link, which carries 120 trunks of PCM information on a 128 time-slot fixed-format signal. In addition, a direct digital switching interface is provided by digroup terminal units which multiplex five T1 digital carrier signals and connect (via a DS-120 link) to the switching frames in the same manner as the voiceband interface units. This network, at maximum size, can terminate 107,000 trunks and service circuits, each operating at 0.9 occupancy.

## 2.3 Description of switching network frames

### 2.3.1 Network clock frame

The network clock frame provides the basic timing and synchronization for the switching network. It generates a stable, accurate source

## **No. 4 ESS:**

# **Transmission/Switching Interfaces and Toll Terminal Equipment**

By J. F. BOYLE, J. R. COLTON, C. L. DAMMANN, B. J. KARAFIN,  
and H. MANN

(Manuscript received July 28, 1976)

*The digital time-division switching network of No. 4 ESS imposes new requirements on the transmission/switching interface, while at the same time presenting opportunities for major improvements and economies. Two new terminals—the Digroup Terminal (DT) and the Voiceband Interface (VIF)—perform the interfacing for digital and analog transmission systems, respectively. Access to the switching network itself is via serial PCM links, each accommodating 120 voice-frequency channels. The DT terminates digital facilities and performs the multiplexing/demultiplexing necessary to interface the PCM links at the switch. The VIF terminates four-wire, voice-frequency analog trunks and performs analog-to-digital and digital-to-analog conversion and multiplexing/demultiplexing to interface with the switch. Both terminals incorporate extensive maintenance hardware and reconfiguration capability.*

*Unitized Facility Terminals (UFTs) are used for a variety of functions—attenuation, signaling conversion, etc.—that must be performed on analog trunks between the basic facility distributing frame on the one side and the VIF and signal processor on the other. The use of UFTs, a standard switching/transmission interface, and modular equipment arrangements and floor plans result in substantial savings in floor space, cabling, and distributing frame requirements, when compared with traditional switching/transmission systems.*

## **I. INTRODUCTION**

The nature of No. 4 ESS, together with evolving equipment concepts, has resulted in a streamlined switching/transmission interface with

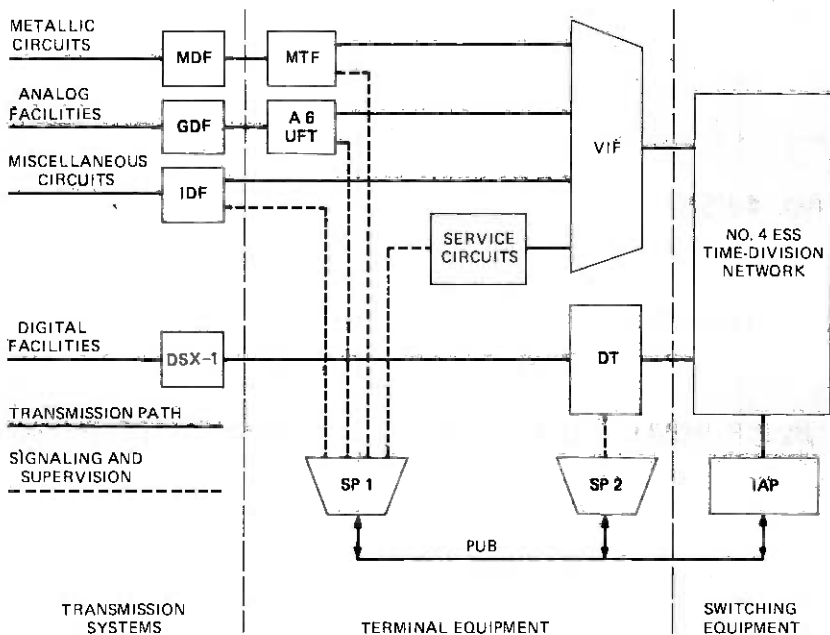


Fig. 1—Transmission/switching interface system.

advantages in cost, space, and maintenance effort. The terminal equipment and arrangement that compose this interface are the subjects of this paper. The interface system is shown in Fig. 1.

The streamlining of the terminal arrangement is clearest in the case of digital transmission facilities. The No. 4 ESS switching network is, of course, a digital time-division network. Serial links into and out of the network use the DS-120 format—an 8.192 Mb/s PCM stream that accommodates 120 voice-frequency channels. Terminals for digital transmission facilities terminate the facility and provide the multiplexing/demultiplexing to interface the transmission format with the DS-120 format. In particular, there is no need to derive each analog channel. Avoiding the conversion step eliminates unnecessary signal degradation as well as the need for a host of per-channel equipment: trunk circuits, distributing frame appearances, etc.

The Digroup Terminal (DT) is the interface for digital facilities in No. 4 ESS. The DT consists of up to eight Digroup Terminal Units (DTUs). Each DTU terminates DS-1 level signals (1.544 Mb/s, 24 VF channels), providing multiplexing of five DS-1 signals for a DS-120 port of the time-division network. The DT also extracts/inserts signaling information from/to the DS-1 streams. Signaling information is then exchanged between the DT and the Signal Processor 2<sup>1</sup> (SP2) by means of a serial 2.048 Mb/s link, again avoiding per channel operations and hardware.

Figure 1 shows the DSX-1, a DS-1 cross-connect frame, as well as the peripheral unit bus connection between the SP2 and the 1A processor.<sup>2</sup> Note that the DT is the only frame between the cross-connect and a port of the switching network. An overview of the DT architecture appears in Section 2; a detailed description appears in Section 3.

The interface between No. 4 ESS and analog transmission facilities is accomplished by the Voiceband Interface (VIF). The VIF contains up to seven Voiceband Interface Units (VIUs). Each VIU terminates 120 four-wire voice-frequency channels and performs the analog-to-digital and digital-to-analog conversion necessary to interface with the DS-120 link of the time-division network.

The four-wire channel at the VIU is a pure message channel; i.e., signaling, with the exception of MF digits, is stripped from the trunk at the unitized terminal and converted to a standard format—looped E and M—before being passed to the Signal Processor 1 (SP1).<sup>1</sup> The interface format, then, between the analog transmission plant and the switch is a DS-120 PCM stream with looped E and M signaling. Again we note the absence of a trunk circuit in the switch. An overview of the VIF appears in Section 2; the detailed description appears in Section 4.

The variety of functions to be performed on analog circuits between the distributing frames and the VIF and SP are efficiently accomplished with Unitized Facility Terminals (UFTs). For the case of carrier facilities, the A6 UFT family provides, in a single package, the channel units, signaling converters, attenuation pads, and maintenance access. The metallic terminal frame (MTF) terminates the wide variety of metallic facilities and provides conversion to standard levels and signaling format for the VIF and SP1. The arrangement is shown in Fig. 1. The unitized concept is discussed in Section 5.

The combination of unitized terminals, a standard analog interface, and the “natural” simplicity of the digital interface lead to particularly efficient cabling arrangements and office layouts as discussed in Section 5. An example is presented in which this combination of traditional switching and transmission functions into a unified whole results in pronounced savings in floor space, cabling and distributing frames.

## II. THE INTERFACE TERMINALS

The basic interface between the transmission plant and the No. 4 ESS switching network is provided, as we have said, by two frames—the DT and the VIF. While the DT interfaces the digital transmission plant and the VIF interfaces the analog plant, structurally the two frames have much in common. In this section we consider some of the similarities as well as some differences in the two frames. In Sections 3 and 4 the frames are discussed individually and in detail.

Many of the similarities in the two frames derive from the fact that both the DT and VIF interface with the switching network using DS-120 PCM coaxial links. The DS-120 is an 8.192 Mb/s serial data signal accommodating 128 encoded voice-frequency channels. (Each 4-kHz voice channel is sampled at 8 kHz and encoded at 8 bits/sample for a rate of 64 kb/s.) Only 120 of the 128 channels are actually used to carry traffic. The remaining eight channels are used to maintain the coaxial links. The choice of a 128 time-slot format with 120 channels is natural in that 120 is a multiple of both the standard 12-channel analog group and the 24-channel digroup (DS-1) used in digital transmission, while 128, being a power of two, is convenient for binary logic.

Seen then from the Time Slot Interchange (TSI),<sup>1</sup> the initial and final stages of switching, the transmission plant is uniformly the same. It is a DS-120 link, whether there is a VIF or a DT on the transmission side.

Each voiceband interface unit (VIU) provides the interfacing between 120 four-wire, voice-frequency, analog channels and a single DS-120 port (two coaxial lines, one for each transmission direction). Similarly, each digroup terminal unit (DTU) interfaces five digital links of 24 channels each to a single DS-120 port. To a first-order approximation, a VIF or DT can be viewed as a frame containing a number of independent units with a duplicated central control.

The structure of a multiplicity of units with a control permits the use of a switchable spare unit to provide service protection. In other words, when a working unit is discovered to be defective, a spare unit is switched in to carry the traffic without interruption. Both the DT and VIF use a 1-for- $n$  protection strategy, i.e., one spare unit for a frame containing  $n$  working units. (For DT,  $n = 8$ ; for VIF,  $n = 7$ .) With automated fault detection and diagnosis, repair of a faulty unit is rapid. The probability, then, of losing service reduces to the probability of two units in a frame failing at nearly the same time. With highly reliable circuitry, service outages are expected to be rare.

The use of a 1-for- $n$  protection strategy affords considerable hardware savings when compared with full duplication. The strategy arises directly from the multiplicity-of-units architecture. By contrast, when the function of a frame cannot be divided into a number of identical, independent units, duplication is often the only viable protection strategy.

Both the DTU and VIU are designed to process 128 VF channels. As we have said, only 120 of these channels are used to carry traffic. Internally, in both units, the spare time slots are used in the fault-detection process. Test signals originating in the controllers enter the units in the spare time slots, are processed by the units as traffic and compared with expected results. Deviations result in alarms and ultimately in protection switching of the unit.

A difference in the two frames is that, since the DTUs process only digital signals, the test signals that the digroup terminal controller (DTC) distributes are digital, while the voiceband interface controller (VIC) must distribute analog tones to test A-to-D conversion, etc.

A primary function of the controllers in both DT and VIF is unit maintenance. Both controllers supply test signals to the units, collect failure data from the units, and control protection switching. To accomplish these tasks, both frames require clock synchronization between units and controller. Both the DTC and the VIC receive office timing from the switch on duplicated clock links, derive necessary waveforms, and distribute timing to the units. The DTC and VIC control 960 and 840 trunks respectively, and hence are designed to be extensively self-checking. They are duplicated to prevent service outages.

When the controllers detect faults either in the units or in themselves they report to the central control (CC)—the 1A Processor.<sup>2</sup> Software residing in the 1A processor determines and directs any reconfiguration—protection switching, controller choice, etc.—and diagnoses the failure. Like other frames in the No. 4 ESS office, an attempt is made to diagnose the fault to a small number of replaceable circuit packs.

The controllers must communicate with the processor for the alarm, configuration and other functions. Neither the DTC nor the VIC require the capacity of the full Peripheral Unit Bus<sup>1</sup> (PUB)—the system by which many of the switch frames communicate with the processor. The VIC receives orders from CC indirectly via the SP1;<sup>1</sup> it does reply directly on the reply (PURB) and control (PUCB) portion of the PUB. The DTC communicates with the SP2<sup>1</sup> via the 2 Mb/s link. The SP2 in turn communicates with CC via the PUB, acting as a buffer for a number of DTs.

A major difference between the DT and VIF is that the DT must process signaling information while signaling is stripped off analog trunks at the UFT. This difference is reflected in more complex equipment and cabling arrangements for analog trunks. It is also reflected in a difference in the controllers. The DTC exchanges signaling information with the SP2 and hence has a call-processing role. The 2 Mb/s link between the DTC and SP2 carries both service and maintenance information. The VIC, on the other hand, is strictly a maintenance controller and exchanges no service information with the CC.

In summary, the gross architectures of the DT and VIF are very similar. Both frames are composed of a number of independent units that interface the switch using DS-120 links, a spare unit for service protection and a duplicated control. The controllers in both frames receive synchronizing clock from the switch, supply test signals to and collect failure data from the units, control protection switching, and communicate with the central processor with the help of the signal processors.

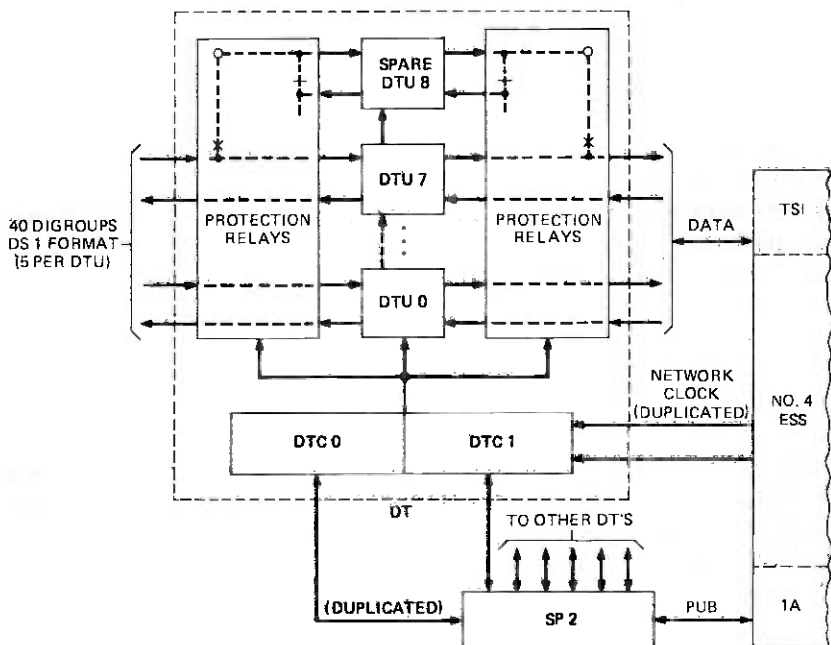


Fig. 2—Digroup terminal block diagram.

In spite of these similarities, there are, of course, important differences between the frames. There is the obvious difference that the DT interfaces digital facilities while the VIF interfaces analog facilities. A second major difference is that the DT has a signaling function that is absent in the VIF.

A more detailed look at the individual frames appears in the next two sections.

### III. DIGROUP TERMINAL

#### 3.1 General

The digroup terminal provides the digital processing necessary to terminate DS-1 level digital signals and the multiplexing and demultiplexing required to interface these signals with time-slot interchange ports of No. 4 ESS. As shown in Fig. 2, a DT comprises up to eight normally active digroup terminal units, a switchable spare DTU, and two nearly identical digroup terminal controllers. Each DTU interfaces five 24 channel DS-1 level signals with a single 120-channel TSI port, resulting in a maximum DT termination capability of 40 DS-1 level signals or 960 channels. Clock signals are transmitted to the DTCs from the No. 4 ESS network clock via a TSI. The DTCs maintain the DTUs and provide the



means for high-speed serial exchange of signaling information and DT maintenance messages with Signal Processor 2 (SP2) memory. SP2 processes the signaling information and interfaces with the 1A Processor over the peripheral unit bus. SP2 does not process DT maintenance messages, but instead shuttles them between the DTs and DT maintenance programs resident in the 1A Processor. Up to 16 DTs interface with a single SP2. Up to four of these DTs can terminate DS-1 level signals from trunks that contain imbedded signaling bits; the remaining DTs terminate DS-1 level signals from CCIS trunks.

Detection of service-affecting faults is accomplished autonomously within the DT by hardware techniques, and such faults result in the autonomous generation of DTU or DTC failure-summary messages. These are interpreted by the DT fault-recovery program, which is resident in the 1A Processor, and the faulty subsystem is isolated by reconfiguring the DTUs or DTCs to preserve the existing calls. The DT diagnostic program is later executed to resolve the failure to a small set of circuit modules.

Physically, a digroup terminal frame is a 7-foot by 4-foot, 4-inch structure, as shown in Fig. 3. The circuit packs are standard 1A Technology packs.<sup>3</sup> The DT is connectorized to facilitate frame installation, and all external connections are made at the connector panel. The protection switch relays and outgoing line equalizers associated with the DTUs are provided on separate panels. The fuse and alarm panel consists of individual DTU and DTC power switches, out-of-service indications and protection switch status. The communication panel provides telephone and teletypewriter connections and a key to request a frame diagnostic. A DTU occupies three shelves and each controller occupies two shelves.

The remainder of Section 3 consists of a description of the operational and maintenance features of the digroup terminal. First the operational features of the digroup terminal unit will be described, with emphasis on new No. 4 ESS features, such as slip synchronization and common control digital processing. Next, the operational features of the digroup terminal controllers will be treated with emphasis on the novel time-division techniques by which signaling information and maintenance messages are processed and exchanged with SP2. Finally, the maintenance of the DT is discussed, with emphasis on the hardware and software concepts that are unique to the DT.

### **3.2 Digroup Terminal Unit**

A DTU comprises a DS-1 interface, a unit processor, and a TSI interface, as shown in Fig. 4. The DS-1 interface synchronizes the DS-1 level signals to the No. 4 ESS network clock, regenerates the incoming DS-1 level signals, and performs the necessary formatting, multiplexing, and

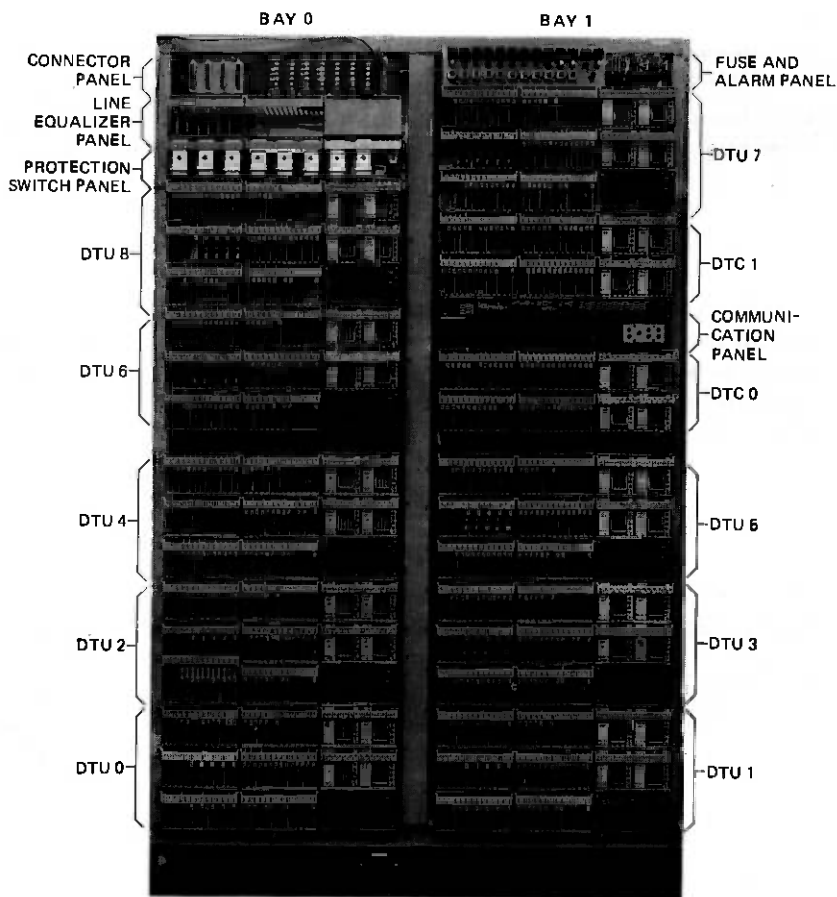


Fig. 3—Digroup terminal frame.

demultiplexing to interface the time-multiplexed 120-channel stream of the unit processor. The functions within the unit processor are sequentially performed on each digroup or channel in the multiplexed stream on a common-control basis. These functions include framing, signaling extraction and insertion, and digroup failure alarming.

Common control processing is important to the economic and maintenance objectives of the DT. In addition to allowing multiple digroups to share the logic associated with the digital processing functions, common control permits this circuitry to be functionally tested in real time. The result is an efficient processing structure for multiple digroups in which faults can be detected absolutely, with no circuit duplication required. Within the unit processor, the digital information is multiplexed in a time-slot format which is identical to that which exists within the time-slot interchange of the No. 4 ESS network.

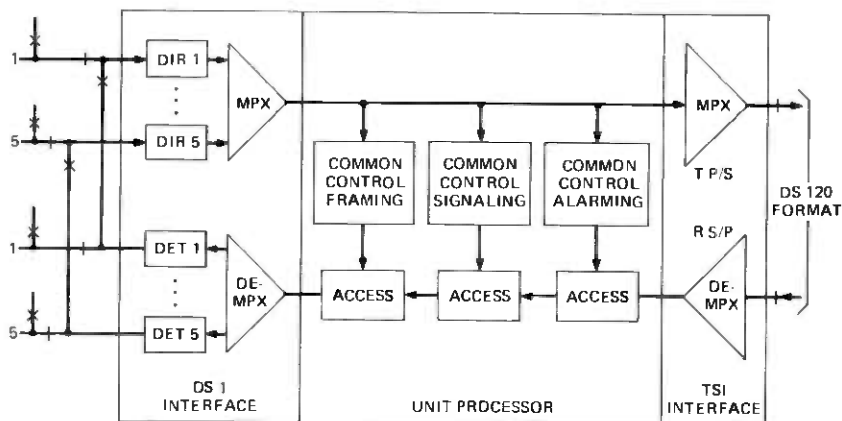


Fig. 4—DTU block diagram.

Within the DTU, the TSI interface performs the formatting and some additional multiplexing and demultiplexing to interface the TSI data port with the DS-120 format.

Figure 5 illustrates the data format at various points within the DTU. A single channel within a DS-1 level signal occupies  $5.18 \mu\text{s}$  on a serial bipolar stream. Within the unit processor, a single channel is compressed to 976 ns in a parallel format, with each digroup compressed to  $23.4 \mu\text{s}$  and stacked sequentially on the time-multiplexed stream. At the output of the TSI interface, the eight data bits of a single channel occupy 976 ns of an 8.192 Mb/s (16.384 megabaud) serial data stream.

### 3.2.1 DS-1 interface

The DS-1 interface receiver is shown in more detail in Fig. 6. In the incoming direction, the data stream is regenerated, monitored for bipolar violations, and converted to a parallel, word-organized format. The recovered clock drives digit and word counters which define the 24 channels. The word counter generates an address which is used to write the receive stores. Two 24 by 10 receive stores, designated the A store and the B store, are provided for each digroup, and these are written alternately with complete frames of data. The read address is provided by a decoding of DT clock with reads 24 channels from each digroup sequentially and stacks the digroups according to the time multiplexed format of Fig. 5.

The synchronization plan for No. 4 ESS requires that each incoming DS-1 level signal be frequency-locked to No. 4 ESS network clock. For trunks with DTs at both ends, frequency lock is ensured by the synchronization of No. 4 ESS network clocks. For trunks with a DT and a channel bank at opposite ends, frequency lock is achieved by loop timing

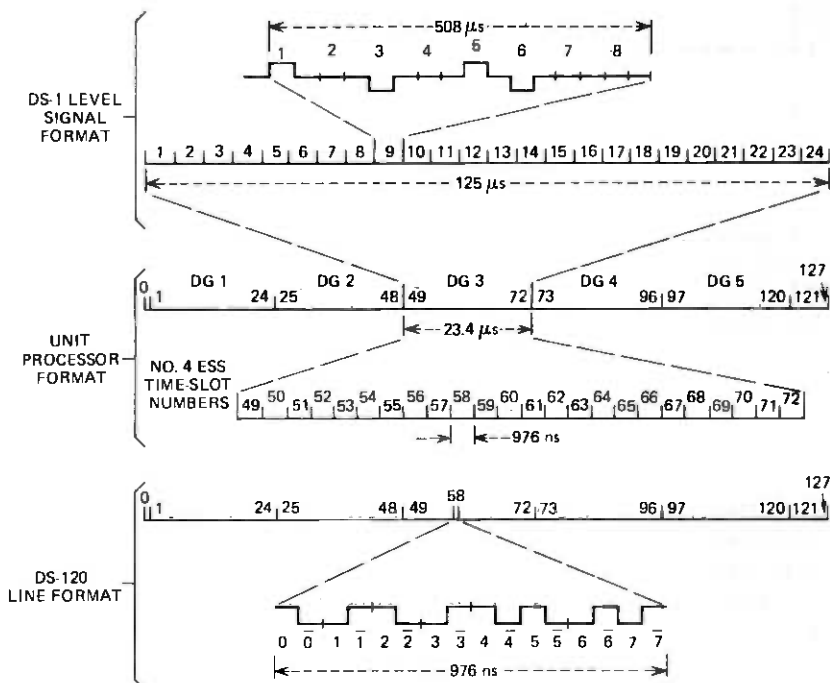


Fig. 5—DTU data formats.

of the interfacing channel banks. However, because of phase fluctuations on digital transmission lines or imperfect line-frequency synchronization, a slip occasionally occurs at the DTU. A slip for a digroup is defined as a deletion or repetition of exactly one frame of information for that digroup sent from the DTU to the No. 4 ESS time-division network. The direction of slip depends on the relationship between the No. 4 ESS and DS-1 level signal frame rates. Normally, the A and B stores are alternately read, but when an instantaneous difference between the No. 4 ESS and DS-1 level signal frame rates make an underflow or overflow imminent, the A store is read twice in succession. Depending on the relationship between office and line frame rates, this double read either deletes or repeats an entire frame of information for that digroup.

In the outgoing direction, A and B per-digroup stores are used to allow the 40 DT digroups to be aligned in phase. This permits direct comparison of outgoing frame and subframe bits which are common for all five digroups, thereby simplifying fault detection. For the transmit stores, the write address is provided by a decoding of DT clock which sequentially writes the 24 channels for each digroup according to the time-multiplexed format of Fig. 5. The read address is provided by a decoding of DT clock which establishes the outgoing-line word rate. In the outgoing

direction the A store is read while the B store is written and vice versa. Since write and read timing are phase and frequency locked, no slip is possible in the outgoing direction. The words read from the transmit stores are serialized, converted to bipolar format and transmitted as DS-1 level signals.

### 3.2.2 Unit processor

Each common control function is a sequential machine which consists of a recirculating memory and combinational logic to determine the next states and outputs from the present states and inputs. As data for each digroup appears on the time-multiplexed data stream, the state variables appropriate for that digroup are clocked to the output of the memory and new state variables are loaded into the memory input while output data is generated. In this way, the combinational logic processes each digroup sequentially, with the individual digroup states retained in memory. The result is a sharing of the combinational logic by all digroups, including test digroups, which leads to the significant maintenance advantages discussed in Section 3.4.

For incoming DS-1 level signals, detection of framing integrity is accomplished by examining the framing information in each digroup for the framing code. Once a digroup loses frame, hunting for the framing position is accomplished by examining data bits over several frames and searching for the framing code. Since each digroup is processed independently, any configuration of in-frame and out-of-frame digroups can occur, but all digroups are processed concurrently with the status of the individual digroups retained in memory.

In a similar manner, the common control signaling extraction function examines the incoming subframe information in each digroup for the subframe code, and it loads the signaling frame Digit 8 (D8) for all 24 channels into a recirculating 128-hit E-lead store.<sup>4</sup> Signaling bits are extracted for each digroup independently in this way, but the signaling store is frozen for digroups which are out of frame or which have a mutilated subframe code.

The common control digroup failure alarm function monitors and times framing losses for local alarms and all zero incoming Digit 2's (D2) for remote alarms by incrementing and decrementing error-timing states.<sup>4</sup> If the error-timing states are incremented past a threshold, the digroup is set to the local or remote alarm state. Similarly, removal of the alarm condition is timed by the error-timing states toward a threshold which results in the removal of the local or remote alarm state.

In the outgoing direction, the common control framing and signaling circuits insert framing bits, subframe bits and signaling frame D8 bits simultaneously for all digroups under control of DT clock. The signaling

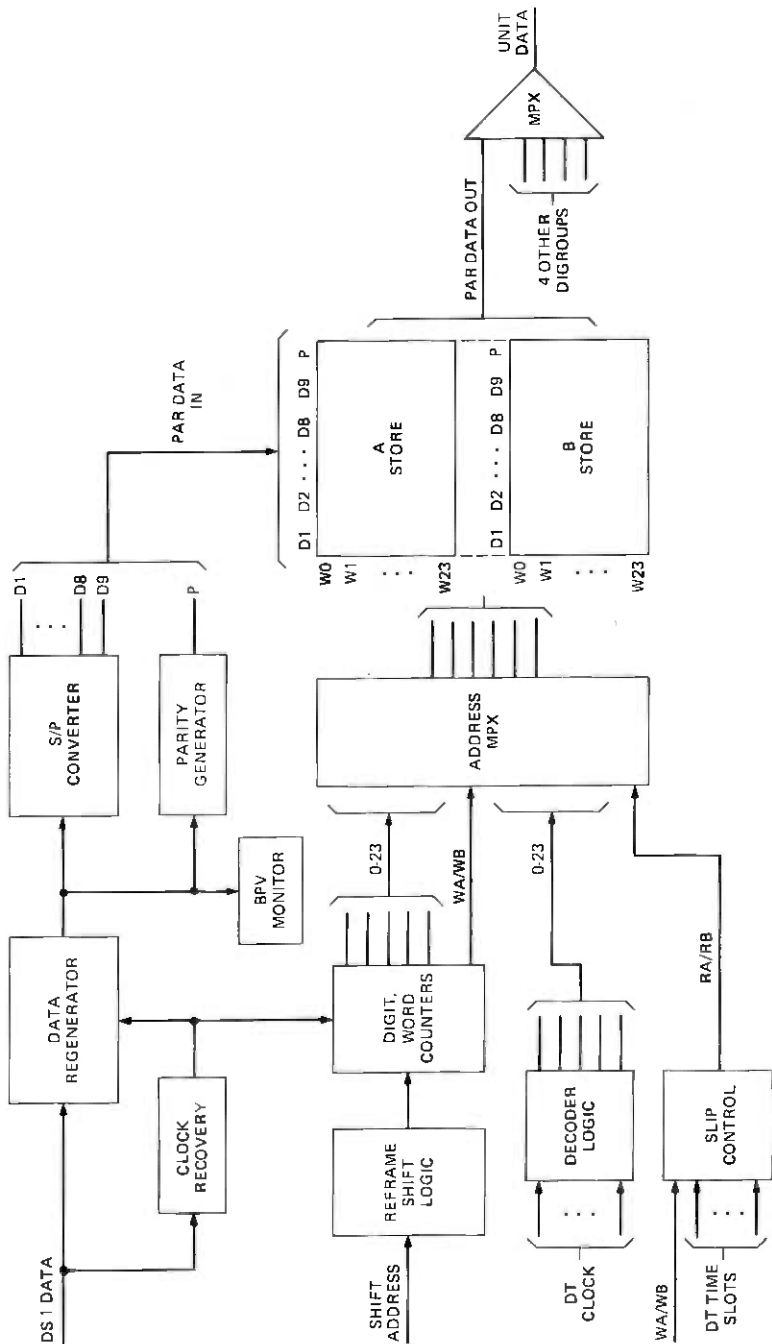


Fig. 6—DS-1 interface receiver.

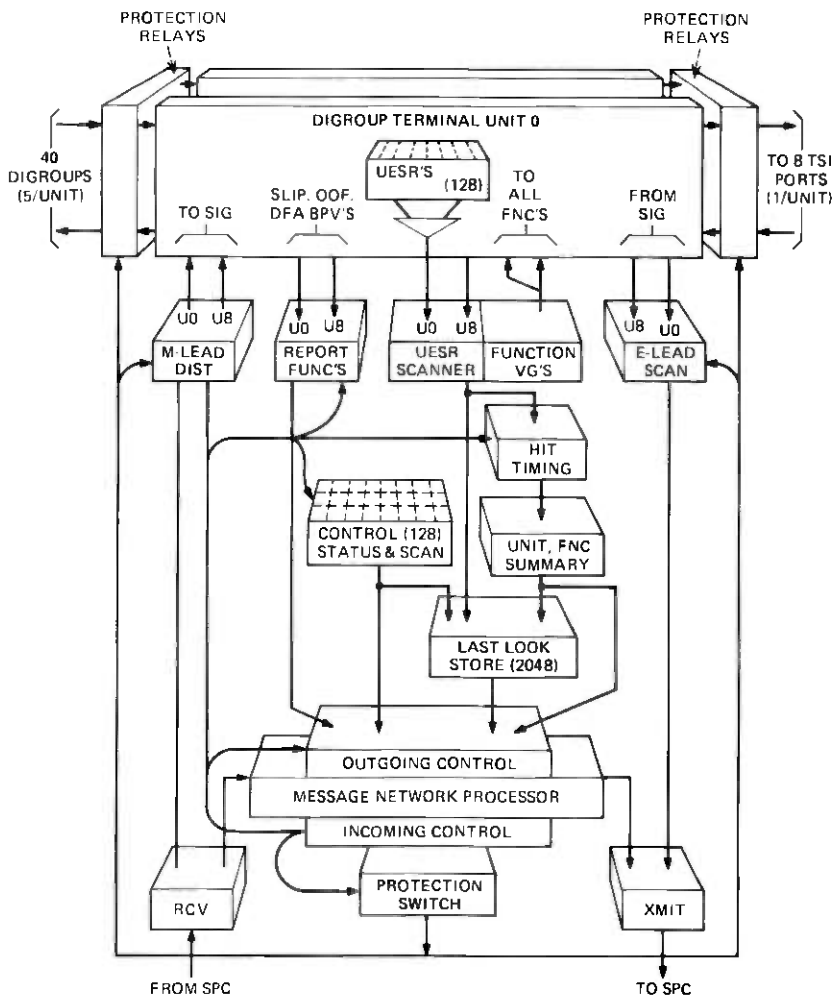


Fig. 7—Digroup terminal controller.

frame D8 bits are obtained from a recirculating 128-bit M-lead store which is updated periodically by SP2 via the DTCs. The common control digroup failure-alarm circuit also forces outgoing D2 bits low for digroups in local alarm.

### 3.3 Digroup Terminal Controllers

The DTCs are synchronous controllers that process signaling, supervision, and maintenance information on a serial time-division basis. Figure 7 shows a block diagram of the information processing hardware of a DTC. The controllers scan E-lead signaling bits from the individual

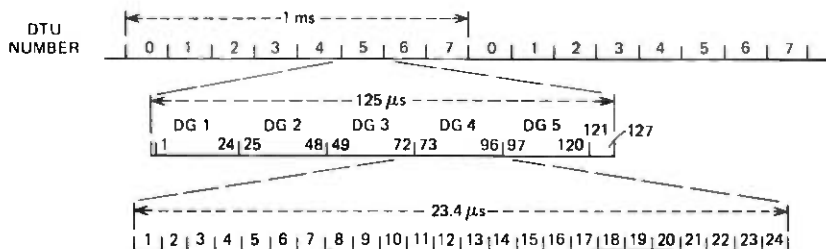
DTUs and send these bits to SP2 in a serial, high-speed stream. The controllers also receive a serial, high-speed M-lead stream from SP2 and provide the means to distribute this information to the individual DTUs. In addition, the controllers contain vector generators which exercise the functions of the DTUs. These vector generators are read-only memories which supply data bits for a test digroup and reference bits that constitute the expected results of the processing of these test data bits. The data bits are designed to exercise all single faults in the DTU functions, and deviations of the function outputs from the reference outputs constitute detected faults, which are loaded along with other alarms into DTU error-source registers. The controllers scan maintenance alarm information from the DTU error-source registers, hit-time the data, and summarize the alarms according to DTU and function. DTC alarms and status are also scanned, and all maintenance information is organized into a serial high-speed error-data stream at the last look store.

The controllers also provide the means to generate, queue, and distribute DT maintenance messages that pass between the DT and DT maintenance programs within the 1A Processor. These maintenance messages include autonomous and elicited alarm messages about the DT and about the T1 lines it terminates. T1 line report information is provided by the report function control which resides within the DTCs. The DTCs also provide the means to execute DTU protection switch orders which are sent from the DT maintenance programs in the 1A Processor.

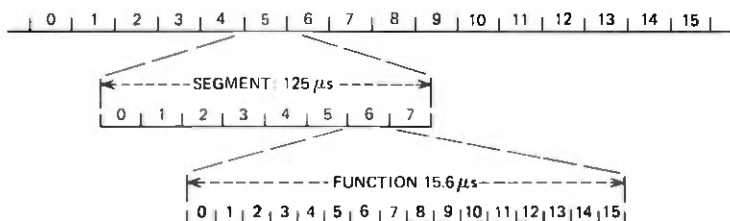
### **3.3.1 Signaling exchanges with SP2**

Signaling exchanges with SP2 are accomplished by multiplexing the E-lead information for eight DTUs into a serial stream, establishing a bidirectional high-speed communication link periodically with SP2, and demultiplexing the received M-lead stream to update the individual DTUs. DT clock organizes the E-lead bits into a continuous 1 Mb/s serial stream by scanning the 128-bit recirculating E-lead store of each DTU for 125  $\mu$ s. As shown in Fig. 8, each channel is assigned a specific 976-ns time slot, which occurs once every millisecond. Normally, the spare DTU is not scanned; however, when a DTU is protection-switched, the multiplex sequence is modified to insert the E-lead information from the spare DTU instead of that from the switched DTU. At the DTC transmitter, the signaling data bits are interleaved with address, opcode, and parity bits, as shown in Fig. 9, to establish a serial 2 Mb/s data stream between the DT and SP2. The address bits identify the location in SP2 memory into which the E-lead bits are to be written. The opcode bits identify the 32-bit word to SP2 as E-lead information, and the parity bit is used in DT/SP2 link maintenance.





(a) SCANNED E AND M SIGNALING DATA FORMAT



(b) SCANNED ERROR DATA STREAM FORMAT

Fig. 8—Time-slot assignment of signaling and maintenance data.

Normally, SP2 initiates a communication with each DT once every 10 milliseconds. Once SP2 is synchronized to the DT, rows of SP2 scan memory are written with 16 bits of data using the addresses which were interleaved with the data hits. In addition, SP2 signal-distribute memory is read to gather rows of 16 bits of data to be sent to the DT. The address used for the read operation is the address used for writing, but it is modified to account for the processing delay of the SP2, so that the SP2

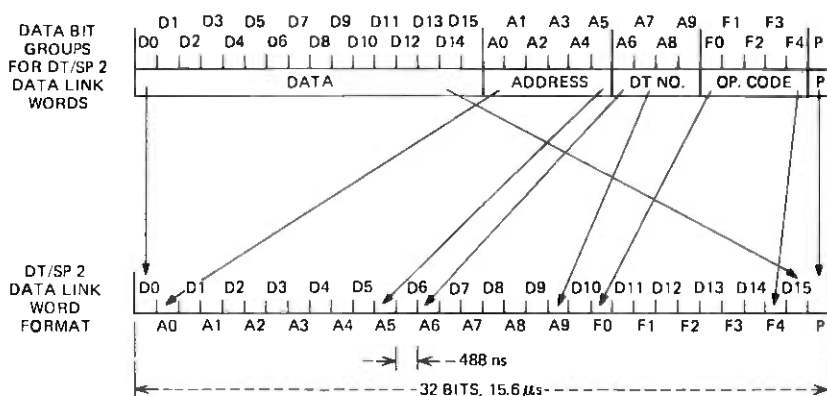


Fig. 9—Data link word format.

to DT word always has the same address as the word traveling in the reverse direction. The opcode bits identify this SP2 to DT word as M-lead information, and the parity bit is added for link maintenance. The SP2 stores the address of the initial E-lead word and terminates the communication when this address is reencountered.

At the DT receiver, the received M-lead words are by design already synchronized with DT clock in both frequency and phase. The DT receiver checks the words for proper DT number, address, and parity. If the opcode indicates M-lead data, DT clock distributes the data bits serially, in real time, to the M-lead store of the DTU indicated by the address. When a DTU is protection-switched, the demultiplex sequence of Fig. 8 is modified to load the M-lead store of the spare DTU instead of that of the switched DTU.

### 3.3.2 DT error-data stream

The processing of DT failure information and maintenance messages is similar to the processing of signaling. As shown in Fig. 7, maintenance data from the alarms within the DTU are collected into error-source registers (ESRs) located in the unit processor. DTU clock organizes the alarm bits into a serial stream by scanning 128 ESR positions in 125  $\mu$ s. DTC clock scans these DTU alarm streams with a 2-ms cycle, as shown in Fig. 8. Each 125- $\mu$ s portion of the cycle is designated a segment, and alarms from DTU0 through DTU8 occupy segments 0 through 8 on the error-data stream. Each of the 8 groups of 16 bits within a segment is designated a function, and each function physically corresponds to a 16-bit ESR in the DTU. The basic DTU alarm bits are hit-timed by the DTC; that is, a failure must be present during three 32-ms periods out of eight such periods in order to be considered a hard failure. The purpose of this hit-timing is to eliminate No. 4 ESS maintenance activity due to transmission phenomena such as noise bursts or loss of data on incoming DS-1 level signals and DTU protection switches. The hit-timed DTU failure information is summarized and inserted into segment 15 on the error-data stream. Thus, if a DTU alarm persists longer than the hit-timing interval, the unit summary indicates the failing DTU.

In a similar manner, DTC status bits and alarms are scanned and inserted into the error-data stream in segment 13 and 14, respectively. The DTC alarm information in segment 14 is summarized by function as shown in Fig. 8. The error-data stream is held in a 2048-bit recirculating last look store which provides all health and status information of the DT.

The processing of DS-1 level signal degradations is performed in an analogous manner. As shown in Fig. 7, bipolar violation samples, slip occurrences, framing state, and remote and local alarm state emerge from the DTUs in serial streams which are further scanned and processed in

report function control. Report function control occupies functions six and seven in segment 12 on the error data stream.

### **3.3.3 DT maintenance messages**

The generation and distribution of maintenance messages is controlled by the message network. Of the outgoing messages, only the DTU summary, DTC summary, and report function messages can be generated autonomously; all others must be requested by DT maintenance software, which is resident within the 1A processor. Outgoing messages are queued in a 128-bit recirculating message-request store. Each bit of the message-request store corresponds to a 16-bit function on the error-data stream, and a logical 1 appearing at the message request store output can gate 16 bits from the error-data stream directly into the message network. Detectors monitor the input and output of the last look store and load the message-request store during segments 14 and 15 if changes are detected. Similarly, the message request store is loaded when DT clock matches the segment and function identifier accompanying a single or multiple message request received by the message network from DT maintenance software. Scanning for new outgoing messages always ensures that DTC status messages, DTC alarm messages, DTU alarm messages, and report function messages are assigned decreasing priority.

Within the message network, the 16 outgoing message data bits are concatenated with a 16-bit message identifier, and the resulting 32 bits are sent to the DTC transmitter where they are formatted into two words of the form of Fig. 9. The opcode bits identify the resulting DT/SP2 link word as message-half one or two and as having high or low priority. The priority of DT messages is strictly a function of the processing time requirements for the message. The messages ultimately reach DT maintenance software via the SP2 high- or low-priority buffers which have nominal unloading times of 10 and 100 ms, respectively. When SP2 initiates a communication with the DT, a DT maintenance message or an idle message is concatenated with the signaling exchange as additional DT/SP2 link words. The outgoing DT maintenance message is loaded into the SP2 high- or low-priority buffer, depending on the link opcodes. The opcodes are retained within these buffers, and when the buffers are unloaded, the message is sent to DT maintenance programs resident in the 1A Processor. In the reverse direction, DT maintenance software loads DT maintenance messages into a special SP2 buffer for subsequent transmission to the DT. At the DT, an incoming DT maintenance message is loaded into the message network and the 16 data bits are distributed to the specified function during the next full 2-ms DT clock cycle. Each incoming message is distributed during a specific time slot or set of time slots during the distribution cycle. After the distribute cycle is complete, the message network scans for new outgoing messages.

### **3.4 Maintenance**

Detection of service-affecting faults is autonomous within a DT, and such failures result in the autonomous generation of DTC or DTU failure summaries. These are interpreted by the DT fault-recovery program and the faulty subsystem is isolated by reconfiguring the DTCs or DTUs in time to preserve the existing calls. The DT diagnostic program is later executed to resolve the failure to a small set of circuit modules. The diagnostic program also searches for latent faults within the DT and is scheduled periodically in addition to responding to detected faults.

#### **3.4.1 Fault Detection**

Within a DTU, parity and hardware test vector techniques are used to detect service-affecting faults. Within the DS-1 interface, parity over address plus data is generated and carried along with the data in both directions of transmission. In addition, common signals in the outgoing direction, such as subframe and framing codes, are compared in appropriate time slots to reference signals. Within the unit processor and TSI interface, test vectors are inserted in spare time slots to exercise the processing functions, and the function outputs are compared to reference vectors, which constitute the expected results. As shown in Fig. 5, time slots 0 and 121 through 127 of the time-multiplexed data stream are not assigned to active digroups. Instead, these eight time slots constitute a short test digroup which receives data bits and control inputs from vector generators within the DTC. These vector generators are read-only memories which are addressed by DT clock. The contents of the ROMs are designed to exercise completely the service-affecting single faults of the processing functions. The outputs of the processing functions are compared in the test digroup time slots to reference vector inputs from the DTC.

These fault-detection techniques are extended to the processing functions of the DTCs. Segment 9 of the error-data stream contains test vectors used to exercise the error-data-stream processing circuitry. In addition, test vectors are applied to the receiver, message network, report functions, and other DTC functions during appropriate portions of the operating cycle. Outputs from the functions are compared to expected results in these spare time slots to detect service-affecting DTC faults.

#### **3.4.2 Fault recovery**

When a service-affecting DTU fault occurs, the DTU error-source registers are loaded with an alarm pattern. This ESR data is hit-timed in the DTCs, and a DTU failure summary is sent to the DT fault-recovery program in the 1A Processor. The DT fault-recovery program interprets the failure summary and sends a DTU protection switch request message

to the DT. The DT message network distributes the message to the protection switch control which switches the spare DTU into the position of the failed DTU, thereby completing fault-recovery action.

When a service-affecting DTC fault occurs, the DTC error-source registers are loaded with an alarm pattern, and a DTC failure summary is sent to DT fault recovery programs. These programs interpret the failure summary, perform software hit timing and send configuration request messages to the DT to configure to the healthy DTC. The message network distributes these messages to the DTC status bits which configure the frame to the requested controller, thereby completing fault-recovery action. When DTU or DTC fault recovery action is completed, a DT diagnostic program is scheduled.

### **3.4.3 Fault diagnosis**

The DT diagnostic program elicits and collects DT ESR information by sending single and multiple message requests to the DT and then maps these ESR alarm patterns into a small set of circuit modules. The diagnostic program also detects latent faults by inverting parity checks and comparator inputs and observing the maintenance system's ability to detect and report faults. The diagnostic also exercises the various DT configurations.

### **3.4.4 DS-1 level signal degradations**

Sixteen bit messages that report degradations on the incoming DS-1 level signals are generated by report function control and sent to the message network. Within the message network, these 16 data bits are concatenated with a 16-bit message identifier, and the resulting 32 bits are sent to the transmitter where they are formatted as a pair of DT/SP2 link words. Two kinds of report messages can be sent for each digroup. The first contains digroup status, such as remote and local alarm state, framing state, and slip state. The second contains a bipolar violation measurement. A digroup status message is loaded into the outgoing message queue on slip occurrence, framing loss occurrence, remote or local alarm state change or on request. A bipolar violation message is loaded if the bipolar sample indicates a sample bipolar violation rate exceeding  $10^{-6}$  violations per bit or on request.

The report messages are sent to the 1A Processor resident DT maintenance programs. These programs record digroup local and remote alarm status and remove trunks from service under alarm conditions. In addition, slip rates, framing loss rates and bipolar violation rates are computed. If the rate computation indicates that the maintenance or out-of-service limit for that digroup is exceeded, the facility craft are notified via CMS-1A.<sup>5</sup>

## IV. VOICEBAND INTERFACE

### 4.1 Frame overview

#### 4.1.1 Function

The Voiceband Interface (VIF) is the standard interface for voice-frequency circuits in No. 4 ESS. A VIF terminates 840 four-wire analog trunks. Typical terminations are analog carrier trunks, metallic trunks, and service circuits. The principal functions of the VIF are analog-to-digital and digital-to-analog conversion of voice-frequency circuits, and formatting of the digital data to be switched by the time-division network.

The VIF consists of seven service Voiceband Interface Units (VIUs), a switchable spare VIU, and a duplicated maintenance controller (VIC). Each service VIU terminates 120 analog trunks and interfaces with the time-slot interchange (TSI) by a pair of coaxial cables using the standard DS-120 digital format.

The overall VIF structure is illustrated in Fig. 10. The VIUs convert voiceband signals to properly formatted digital streams in one transmission direction and reconstruct voiceband signals from digital streams in the opposite direction of transmission. Processor access to the VIF is provided by the Peripheral Unit Bus System, partially via the signal processor. The VIC controls the protection switching relays and provides timing, control, and maintenance functions for the VIUs.

#### 4.1.2 Physical characteristics

The VIF equipment is contained in a triple-bay, 6-foot, 6-inch wide by 7-foot high frame (Fig. 11). The major components are the eight VIUs and a pair of controllers (VICs). Each VIU is a three-shelf, 18-inch high assembly including power converters. The two VICs including power converters are housed in a four-shelf 22-inch high assembly in the middle bay. Voice-frequency and PCM switching relays used to configure the spare VIU in place of any of the service VIUs are housed in the top two shelves, 14 inches high, of each bay. Connectorized cabling is used for voice frequency, PCM, and processor interfaces.

#### 4.1.3 Technology

VIF circuit functions that are primarily digital use standard 1A Technology<sup>3</sup> circuit packs, multilayer back-planes, and wiring. The maintenance controller uses this technology exclusively. However, a large portion of the VIU functions are analog, which leads to use of a mixture of technologies. 1A Technology is used for digital functions and a combination of hybrid integrated circuits and discrete components mounted on epoxy boards is used for the inherently analog circuitry. An example

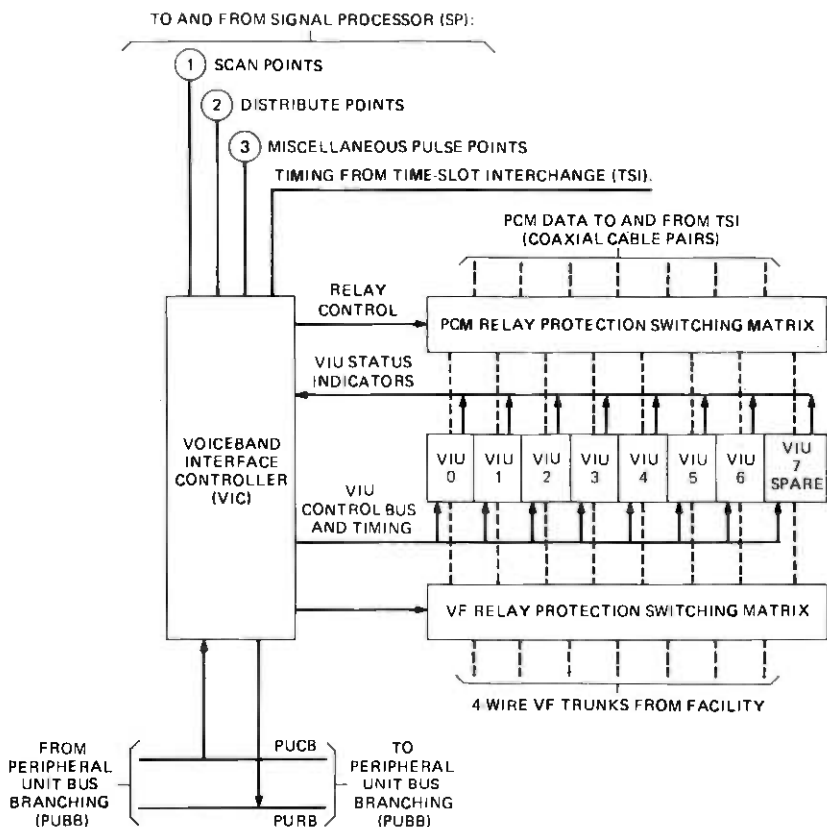


Fig. 10—Voiceband interface block diagram.

is illustrated in Fig. 12. This circuit pack performs the functions of filtering, sampling and PAM multiplexing for eight voice-frequency circuits. Filtering is provided by individual thin-film resistor-capacitor active filters. Sampling and multiplexing are performed on a bilevel ceramic (conductor paths on both surfaces interconnected by plated-through "vias") using beam-leaded silicon devices. Other analog functions use bilevel ceramics with thin-film resistors, applied chip capacitors and beam-leaded silicon devices.

## 4.2 Voiceband Interface Unit

### 4.2.1 VIU functions

A block diagram of the VIU is shown in Fig. 13. In the transmitting direction, the 120 trunks used for traffic and the eight trunks used for testing and maintenance purposes have low-pass filters at their inputs.

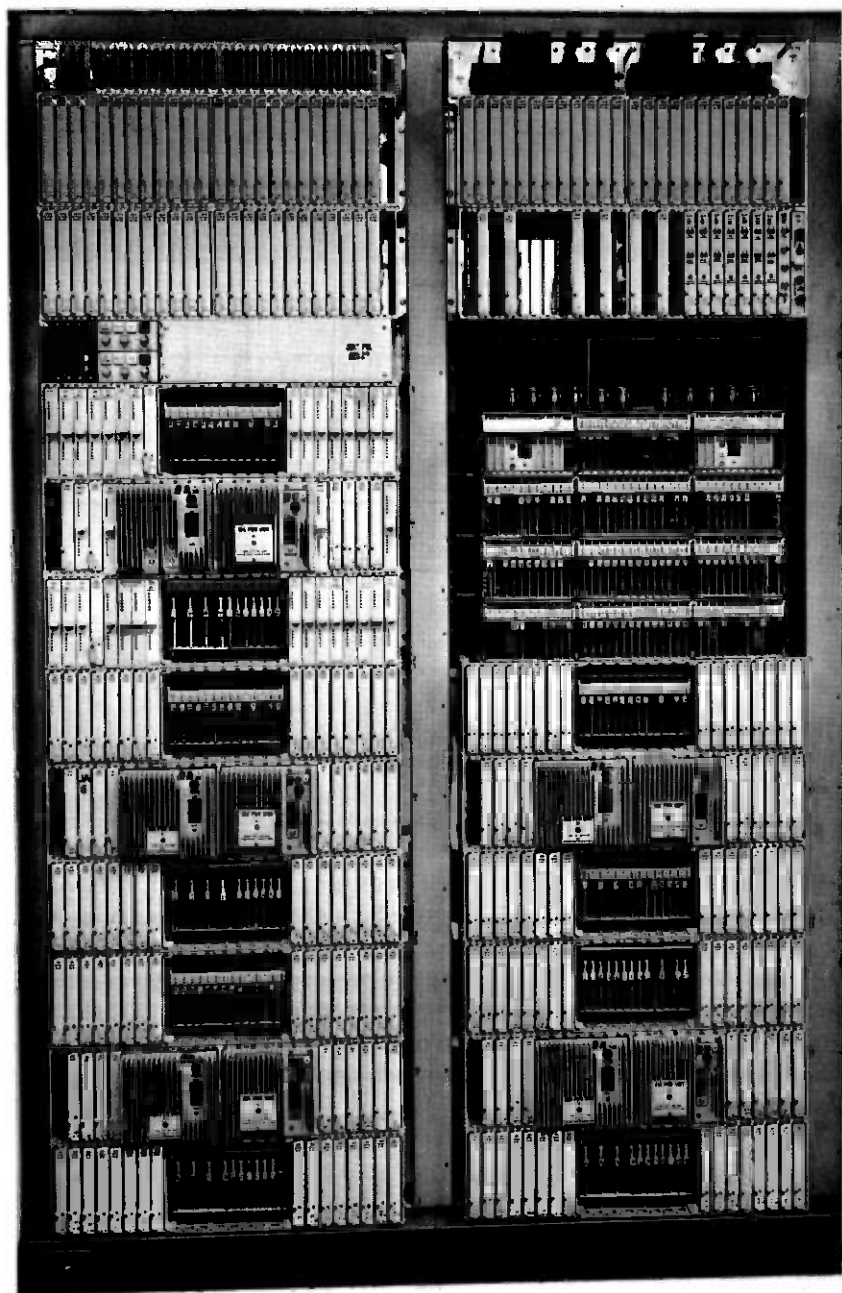


Fig. 11—Two bays of the voiceband interface equipment frame.



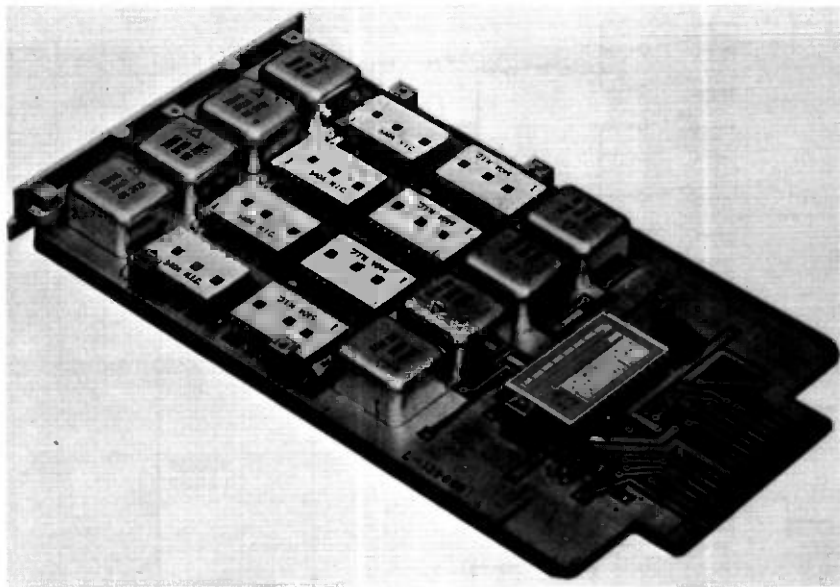


Fig. 12—VIU analog circuit pack.

These filters band-limit the input signals to less than 4 kHz to prevent foldover distortion. That type of distortion would otherwise be introduced by the 8-kHz sampling associated with PCM encoding.

The VIU uses two stages of multiplexing. In the first stage, all trunks are sampled at an 8-kHz rate and multiplexed onto four buses of 32 trunks each. Each of these buses contains samples for 30 service trunks and two maintenance trunks. At this point, the samples from the trunks are natural Pulse Amplitude Modulation (PAM) samples approximately  $3 \mu\text{s}$  in duration. In the second state of multiplexing, each pair of the 32-trunk buses is combined into a 64-trunk PAM bus. Each sample on the PAM bus is approximately  $2 \mu\text{s}$  in duration.

A coder performs A-to-D conversion for each 64-trunk PAM bus. The coders use a nonlinear 15-segment  $\mu$ -law encoding characteristic and code each PAM sample into an 8-bit PCM word. One bit is used as a polarity indicator, and the remaining seven are used to represent the amplitude of the sample. A coder output PCM (COP) circuit gates the parallel PCM words coming out of the two coders to the access circuit, one at a time. Each PCM code word exists in parallel form at the access for 976 ns, which is one PCM word time slot. The access circuit plays an important role in VIU maintenance, as described in Section 4.4.2.

Data from the VIU is sent, via the 16.384-megabaud coaxial line, to the TSI. Transmitters and receivers are used in both the VIU and the TSI to convert back and forth between parallel PCM data and the serial

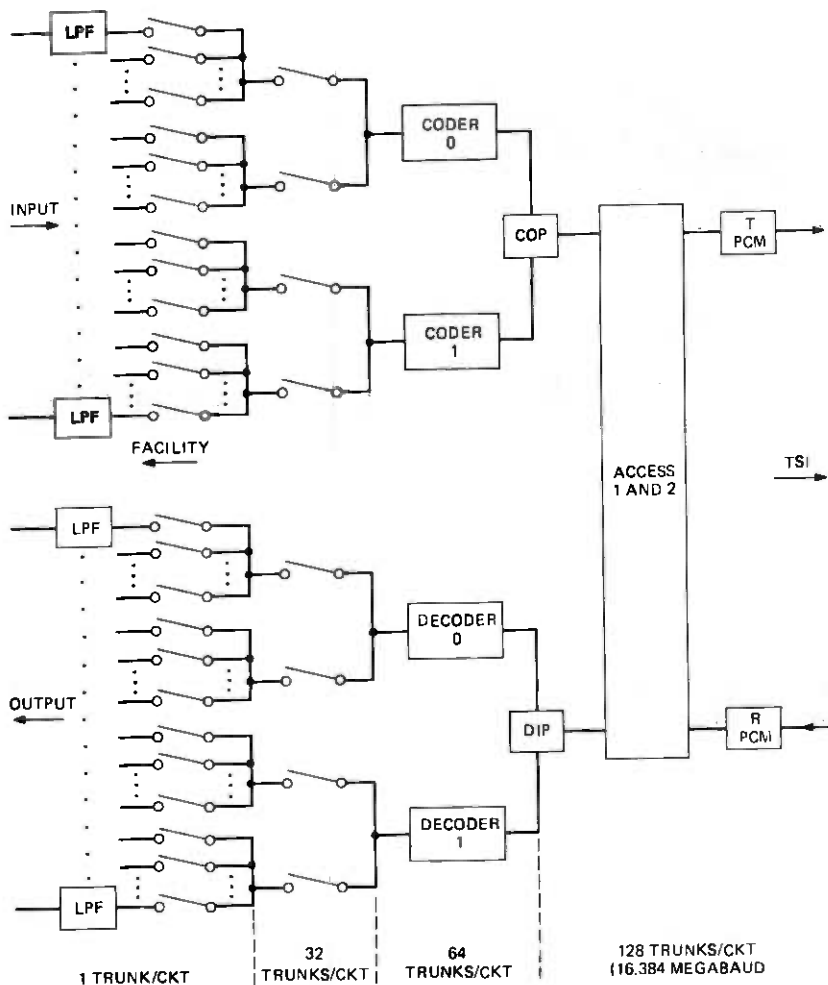


Fig. 13—Voiceband interface unit block diagram.

16.384-megabaud format and to derive local timing and framing information. The transmitter in the VIU converts the parallel PCM words from the access into a serial form containing timing and framing information, constructing the DS-120 signal. The transmission is over coaxial cable to the receiver in the TSI.

The receiver section of the VIU accepts the DS-120 signal coming from the TSI, extracts local data timing, regenerates the data pulses, and formats the incoming data back into parallel PCM data words. Buffer stores are used to realign incoming data words with timing supplied from the VIC.

PCM words pass through the access to the decoder input PCM (DIP)

circuit, which routes them alternately to one of two decoders. Each decoder is used for 64 trunks. The 64-trunk PAM buses are demultiplexed into four 32-trunk PAM buses and then into single trunks. Low-pass filters reconstruct the voiceband (up to 4 kHz) from the demultiplexed PAM signal.

#### **4.2.2 Transmission characteristics**

A-to-D and D-to-A conversions in the VIF use the Bell System standard 8-digit, 15-segment,  $\mu = 255$  coding characteristic. Transmission characteristics are compatible with and virtually identical to those of digital channel banks that use that coding characteristic.<sup>6</sup> Thus, an analog trunk terminating on a VIF can be switched to a digroup terminal, transmitted over a T1 line and ultimately terminate on a digital channel bank.

All analog terminations on VIF are four-wire with a nominal -3 transmission level (TL). This interface is not defined as a standard point; it is permitted to vary by a few tenths of a dB. Variation of this level, office cabling loss, etc., are compensated for by loss-adjustment pads in the interfacing terminal equipment to provide the appropriate standard levels in these terminal equipments.

### **4.3 Voiceband Interface Controller**

#### **4.3.1 Functions**

The primary functions of the VIC are to provide timing, control, and maintenance signals to the VIUs, to monitor the status or health of the units, and to control VIU protection switching. Most controller circuitry is fully duplicated for increased reliability and fault detection capability, and either controller can perform the required functions.

Timing is derived from duplicated master timing links provided by the TSI. This timing information is used to derive clock waveforms for use by the controller. The derived clock waveforms are distributed to the units on a clock bus from each controller. The units are configured to use the appropriate clock bus.

Various analog and digital maintenance signals are also derived in the VIC and distributed to the units. These signals are used to detect and isolate failures in a unit. The status of the units is monitored by sampling 22 indicators from each unit. When one or more of the status indicators report a failure in a unit, the VIC reports the failure to maintenance software residing in the 1A Processor, and the spare unit is protection-switched to provide service continuity.

#### **4.3.2 Controller-processor interfaces**

Processor orders are sent to the VIC through duplicated 16-bit SP pulse point buses. Either SP bus sends the orders to both controllers.

Responses from the controller are sent as 16-bit words plus parity over the PURB. The controllers can be configured such that either VIC can respond on either reply bus. In addition, the PUCB is used for failure reporting. The 1A Processor periodically interrogates the controllers, which respond if failures exist.

#### **4.4 Maintenance plan**

##### **4.4.1 Controller maintenance**

Controller hardware faults are detected autonomously by unique indicators in each controller half, such as parity, or by mismatches between duplicated controller functions. If one of these indicators records a failure, an interrupt is generated and fault-recovery software removes the suspect controller from service and schedules a diagnostic of the controller to isolate the source of the fault. However, since the VIC is a maintenance controller and serves no real-time, call-processing-related function, a large percentage of circuitry is exercised only during diagnostics. Thus, latent, non-service-affecting faults can exist that are stimulated and detected only by the diagnostic. Consequently, a diagnostic is run periodically even when there is no indication of failure.

##### **4.4.2 Unit maintenance**

Each controller monitors 22 status indicators for each VIU. These indicators include parity failures, power converter failures and out-of-limits signal or distortion levels in maintenance trunks. When an indicator records failure in a service VIU, the spare unit is protection-switched to provide service continuity, and the unit diagnostic is scheduled.

(i) *Signal and distortion monitoring.* Each 32-trunk PAM bus (Section 4.2.1) contains two trunks that do not carry traffic. One of these trunks for each bus is dedicated to full-time signal and distortion monitoring. A sinusoidal maintenance signal supplied by the VIC is applied to each of these trunks. The maintenance signal continuously cycles through four amplitudes at a low frequency rate. These maintenance trunks are sampled, multiplexed and coded in the same manner and by the same circuitry as service trunks. Internal timing of the VIU is arranged such that when any time slot, say  $N$ , appears at the access to be transmitted to the TSI, the same time slot received from the TSI also appears at the access. Thus, there is "time-slot alignment" of PCM data words at the access. This time-slot alignment permits looping the four maintenance trunks from COP to DIP through the access. These maintenance trunks are then decoded and demultiplexed to baseband.

The signal amplitude of each of these maintenance trunks is monitored and the distortion is measured. In the event the signal amplitude or

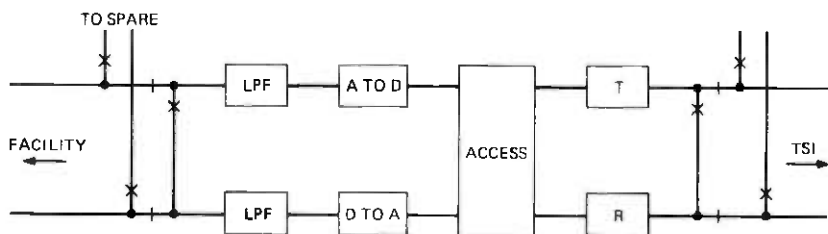


Fig. 14—Trunk looping in maintenance position.

distortion limits are not met, status indicators monitored by the VIC indicate a failure. These indicators detect failures of equipment common to 32 or more trunks.

(ii) *Time-slot exchange.* The unit diagnostic is run on the spare VIU only when it is in the nonservice position or on a service unit when it is protection-switched. When in this maintenance position, all 120 trunks are looped at voice frequency and PCM as shown in Fig. 14. Storage registers in the access can be used to perform a "time-slot exchange" as follows. Coded PCM words from one of the maintenance trunks are stored in a register in the access. This word can then be written in place of the data for any trunk, say  $N$ . The data are then decoded and demultiplexed in trunk  $N$ , looped back to the input of trunk  $N$ , and stored in another register in the access. They then are written into another maintenance time slot and decoded, demultiplexed, and tested by the maintenance signal monitor. This provides the ability to locate failures of per-trunk equipment such as filters and multiplexing gates.

## V. TOLL TERMINAL EQUIPMENT INTERCONNECTION

### 5.1 Frame interconnection characteristics

The voiceband interface frame establishes a well defined and standardized transmission interface between the No. 4 ESS switch and analog transmission facilities. Similarly, the Signal Processor 1 establishes a well defined and standardized signaling interface between the No. 4 ESS and analog transmission facilities. Together, the VIF and SP functionally eliminate the need for trunk circuits. Trunk circuits traditionally functioned to match the various types of transmission facilities, with their variety of signaling methods, with the service characteristics of the trunk and the signaling methods used within the switching system. In No. 4 ESS, all types of analog transmission facilities interface the VIF and SP1 in exactly the same way. This standardized interface and the modular design of both the VIF and SP1 grant the design freedom to distribute them as required to optimize floor-plan arrangements.

Digital transmission facility interfaces with the switching system are more drastically affected by the development of the Digroup Terminal.

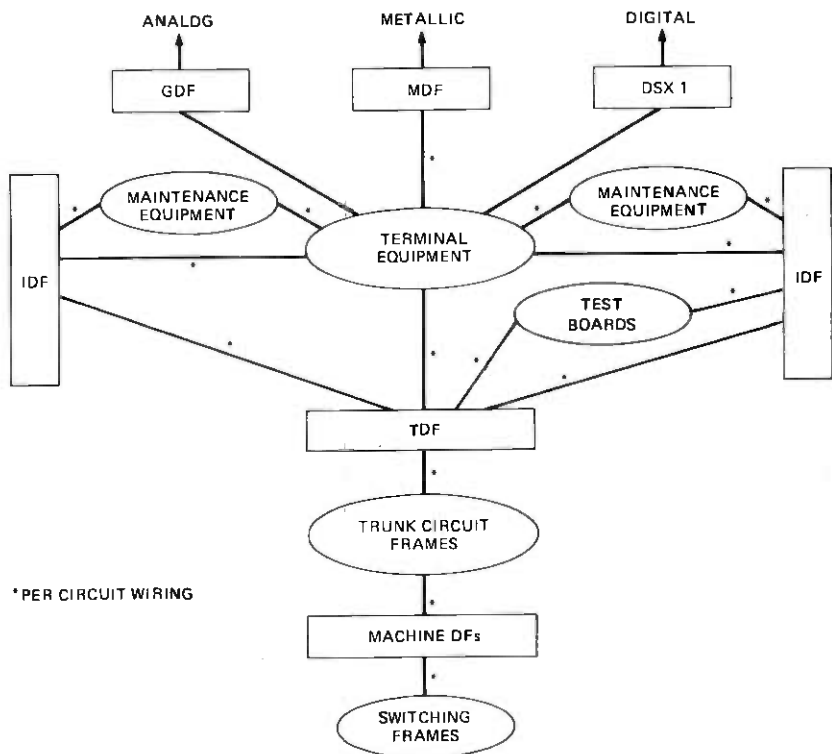


Fig. 15—Typical machine cabling.

Where digital facilities previously had to be converted to individual voice-frequency analog channels to interface trunk circuits, the digital signals can now be processed by the DT to enter the switching system directly as a digital bit stream. Signaling information, as previously described, is extracted by the DT and passed to the Signal Processor 2, also in digital form. This drastically simplifies the signaling interface and interconnection.

Perspective on how the characteristics of the No. 4 ESS with standardized DT, VIF and SP interfaces can be exploited to substantially improve office interconnection may be gained by considering a typical existing electromechanical toll switching system. As indicated in Figure 15, multiple Distributing Frames (DF) are used to cross-connect individual pieces of terminal equipment (channel banks, signaling units, attenuators, repeaters, analog echo suppressors, etc.) and maintenance equipment [toll testboards (TTB) with dedicated jacks, VF patching bays, etc.] to form the required variety of trunk types. The asterisks indicate that almost all cabling is on multilead, per-circuit (trunk) basis, equating to near astronomical numbers of wires for a system using the full capacity

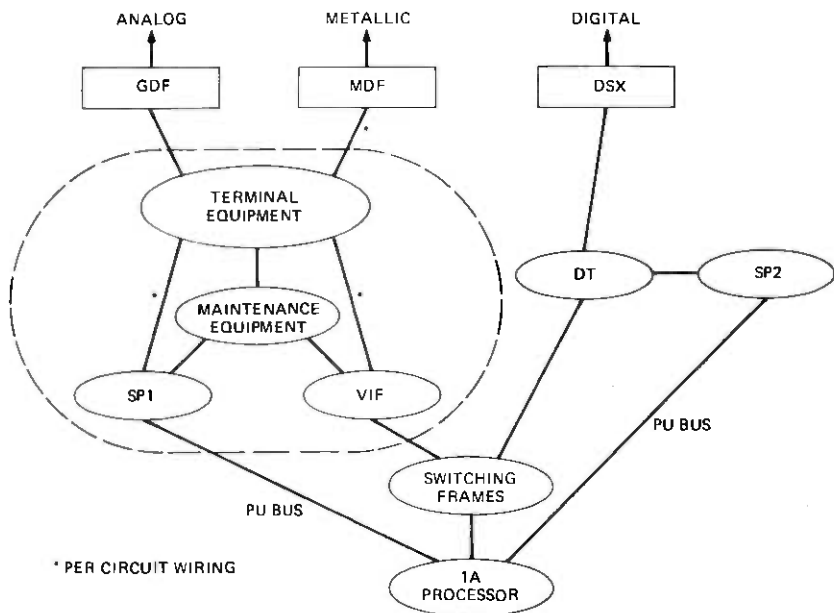


Fig. 16— No. 4 ESS cabling.

of No. 4 ESS. For simplicity in presentation, the terminal equipment is shown as a single entity. In reality it is scattered throughout the building, often on several floors, and several IDFs (Intermediate Distributing Frames) may be required to provide all the necessary terminations. When the terminal and maintenance equipment is properly interconnected by multiple DF cross-connects, the equipment connects via the trunk distributing frame (TDF) (sometimes called a machine IDF) to the trunk relay circuits. These are again cross-connected to the variety of switching frames composing the switching system. These DFs, in addition to permitting proper interconnection of the variety of trunk relay circuits with the variety of facilities and terminal equipment, permit traffic-load balancing on the switching frames.

In contrast, Fig. 16 illustrates interconnection within the No. 4 ESS. As will be discussed further, all distributing frames except the basic facility interfaces—Group Distributing Frame (GDF), Main Distributing Frame (MDF), and the Digital Signal Cross-connect frame (DSX-1)—have been eliminated. For digital trunks, all per-circuit wiring has been eliminated. All per-circuit wiring also has been eliminated between the switching frames (TSI, TMS, 1A Processor) and the analog terminal core indicated by the dashed oval. Although, individual per-circuit wiring could not be eliminated within this core, it has been simplified by the use of connectorization and reduced by a modular floor plan layout.

Much of the total wiring is performed at the factory using the facility terminal concept where the maintenance access equipment is built into the equipment frame together with all the circuit elements required for a working trunk. The basic cabling plan is diagrammed in Fig. 17, including interconnection of existing equipment where the No. 4 ESS replaces a No. 4A crossbar system.

The following paragraphs will describe how these new interfaces and new equipment were used to optimize interconnection of toll terminal equipment in the No. 4 ESS environment.

### **5.1.1 Analog facility terminals**

The concept of unitization (combining needed circuit elements in a single package) was well developed when the No. 4 ESS design was started. In fact, analog carrier unitized facility terminals which functionally provide the needed No. 4 ESS interface were developed for application in crossbar switching systems. The existing designs, however, did not fully exploit the interconnection characteristics of No. 4 ESS.

The standard interface and the nonblocking characteristics of the switch suggested elimination of the traditional distributing frame with its administrative and daily operational problems. Elimination of the distributing frame in turn suggested that connectorization be applied to the cables between the Unitized Facility Terminals (UFT) and the VIF and SP1. Connectorization could be applied readily to either end but would not be nearly as effective in reducing installation time and expense as would applying full connectorization. To achieve double-ended connectorization, a floor-plan arrangement would be needed that could be restricted and confined in terms of cable length and routing. Connector cables tend to be expensive if they cannot be produced in large quantities. Therefore it was essential that the connectorization plan require only a small number of distinct cable designs.

Once it was established that a suitable floor location and connectorization plan were feasible, the UFT was designed for connectorization in conjunction with the designs of the VIF and SP1.

Figure 18 shows a typical UFT design. All connectors are conveniently located at the top of the frame. Included in the frame are: the channel bank multiplex and demultiplex equipment to translate between individual VF channels and carrier channel groups, Single-Frequency (SF) signaling equipment to translate between line tone signals and standard dc unidirectional signals called E and M (signals toward the switch are carried on E leads and signals toward the facility are carried on M leads), attenuators to establish standard transmission levels, maintenance access equipment to permit testing at standardized transmission and signaling reference points, communications equipment to allow craftspeople to communicate with other craft and testers, patching equipment



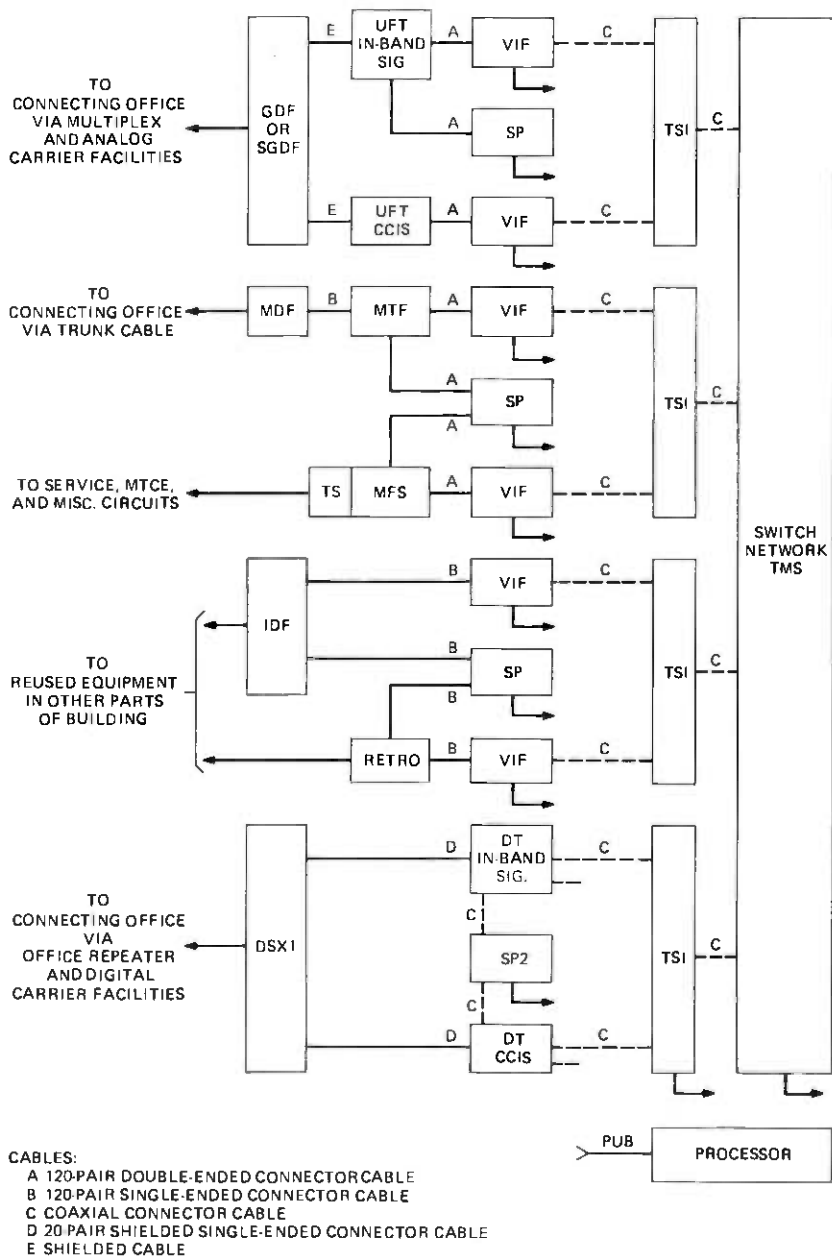


Fig. 17—Basic cabling plan.

to permit emergency restoration of failed facilities; and equipment common to all 96 circuits on the frame or the 480 circuits composing a standard frame set. Common equipment items are carrier frequency

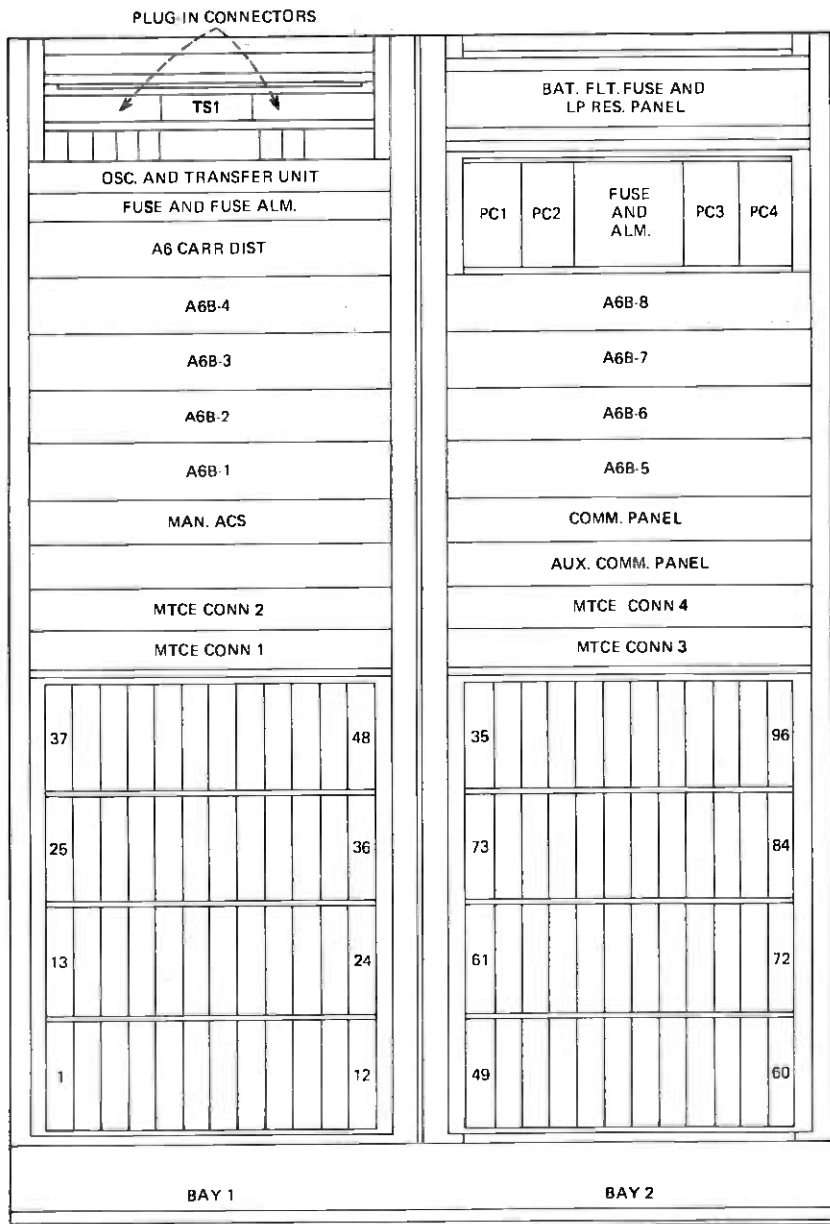


Fig. 18—Typical unitized facility terminal.

supply and distribution, SF tone supply, power, and fuse and alarm units.

A family of UFTs for analog carrier facilities has evolved to provide

needed features and characteristics economically. One member of the family will be used with trunks requiring in-band single-frequency signaling. This frame may also be used for trunks using Common Channel Interoffice Signaling (CCIS) by substituting a plug-in unit providing only an attenuator function for the signaling unit plug-in. It is substantially more economical, however, to use a second-type frame designed specifically for CCIS. The use of the first frame for temporary use on CCIS trunks will be useful as an expedient to transfer trunks from in-band signaling to CCIS as the CCIS network expands.

The frames are arranged to provide a number of features or capabilities on an optional basis. They may be arranged to interface the multiplex equipment at the group distributing frame as is current practice. Or, they may be arranged such that five channel groups (banks) are combined within the UFT to directly form a supergroup (60 VF channels) and interface at the supergroup distributing frame.

Each frame may be equipped with plug-in units to provide one-way or two-way carrier failure alarms if required. This feature provides a prompt notification to the No. 4 ESS of a carrier failure and subsequent restoration. For two-way operation, the alarm is used to notify the distant end of an incoming failure. It permits both ends of the trunk to be properly conditioned when a carrier system fails and again when it is restored.

Each frame may be equipped for local access in the equipment aisle, only or equipped for both local and remote access from a central 51A test position through the Switched Maintenance Access System (SMAS 3B).<sup>5</sup>

Recommended arrangements call for frames to be installed in full sets comprising 40 channel banks and 480 VF channels. Such a set fully utilizes the carrier frequency supply common to 40 banks, allows standard wiring and identification of banks used to form supergroups (some of which span more than one frame), and permits standard and orderly application of connectorized cabling.

### **5.1.2 Metallic Terminal Frame**

Short-distance toll connecting trunks, operator trunks, and a variety of other trunks utilize only cable pairs (sometimes enhanced by voice-frequency repeaters and signaling range-extension devices) as the transmission facility. Digital facilities are rapidly replacing these purely metallic (noncarrier) trunks throughout the Bell System. This process is expected to accelerate with the introduction of the DT because of the economy of the direct digital interface. Although the number of metallic trunks will be small, it is necessary to provide for their interface with No. 4 ESS.

Metallic trunks use dc signaling in a variety of forms. Signaling con-

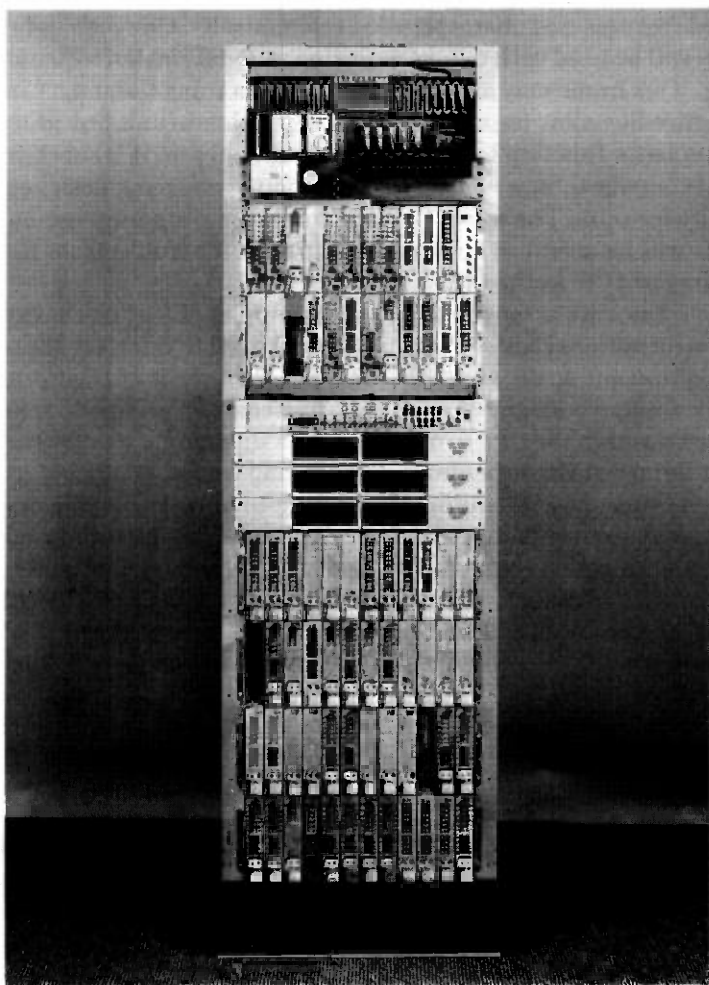


Fig. 19—Metallic terminal frame.

version must normally be performed to meet the standard SP signaling interface (four-wire E and M). Transmission gain or attenuation is normally required to meet the standard transmission level at the VIF interface. Since many of these trunks utilize two-wire facilities, two- to four-wire conversion is frequently required.

The Metallic Terminal Frame (MTF) has been designed to provide a single package capable of terminating the variety of metallic trunks likely to interface with No. 4 ESS. The frame is shown in Fig. 19. A family of plug-in metallic terminal units has been designed to accommodate the signaling, transmission, and service type variables (loop pulsing,

battery and ground pulsing, two-wire, four-wire, incoming, outgoing, operator, message, etc.).

Connectorization has been applied between the MTF and VTF and SP in exactly the same form as for the UFT. Additionally, connectors and single-ended connector cabling tie the MTF to the outside cable plant via the main distributing frame. Provisions similar to those provided in the UFT for local equipment aisle or optional remote testboard maintenance access have been incorporated in the frame design.

### **5.1.3 Digroup Terminal**

Interconnection of the DT to the No. 4 ESS differs markedly from the analog UFT and MTF since there are no requirements to interconnect on the level of individual VF channels. Single-ended connectorized cables connect the DT to the digital cross-connect frame (DSX-1) or directly to office repeaters. Only four such cables are required for the full 40-digroup (24 VF channels each) capacity of the DT.

### **5.1.4 Multifrequency signaling receivers and transmitters**

Address signaling in No. 4 ESS may be in the form of dial pulsing, multifrequency (MF) pulsing, or digital data on a common link in CCIS. Multifrequency pulsing is the dominant mode in today's toll network but will diminish with the deployment of the CCIS network.

MF receivers and transmitters are not dedicated to specific trunks but form a common pool such that any idle transmitter or receiver may be selected by the No. 4 ESS to transmit or receive MF pulses on any MF signaling trunk. Each receiver interfaces with a VIU to permit reception of MF signals from an incoming trunk through a switched connection. A multilead interface with the SP permits it to identify the received MF pulses for subsequent interpretation by the 1A Processor. Similarly, each transmitter interfaces with a VIU for transmission of MF signals via a switched connection to an outgoing trunk. The SP provides the address to be outputted to the MF transmitter.

The MF signaling frame (shown in Fig. 20) mounts up to 32 receivers and 32 transmitters and is connector-cabled to the VIU and SP. An optional terminal strip provides a convenient means to connect other service circuits into the No. 4 ESS. Terminal strips are also used to connect MF testing circuits, optionally mounted on the multifrequency signaling frame, and to provide loop arounds for periodic No. 4 ESS internal testing.

### **5.1.5 Maintenance equipment**

All UFT and MTF frames may be optionally equipped for 51A test position access by the SMAS 3B.

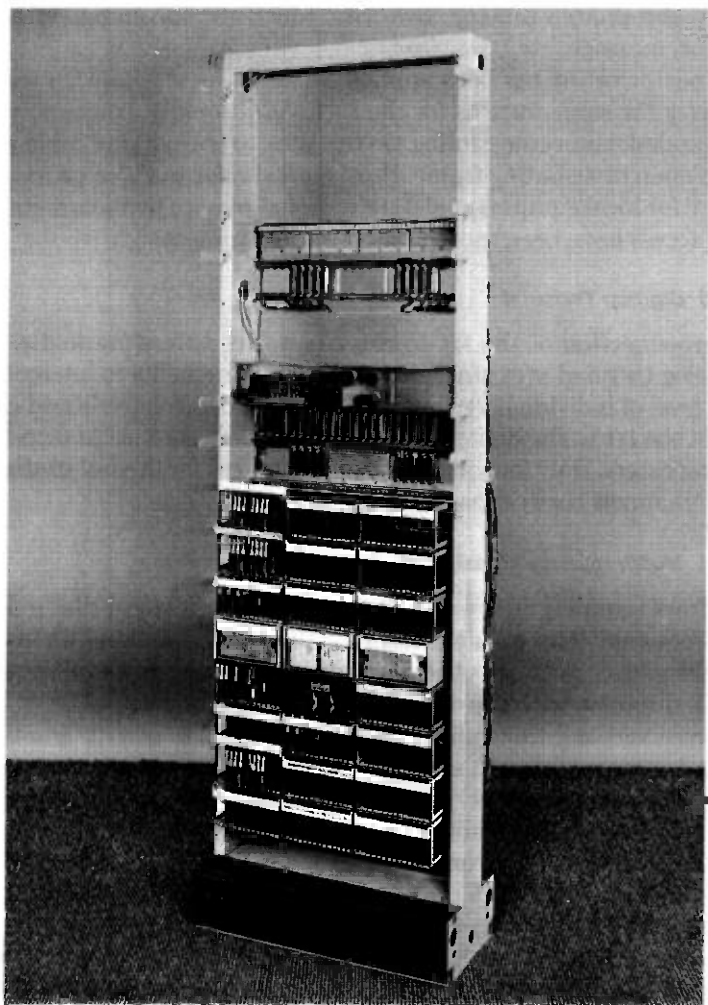
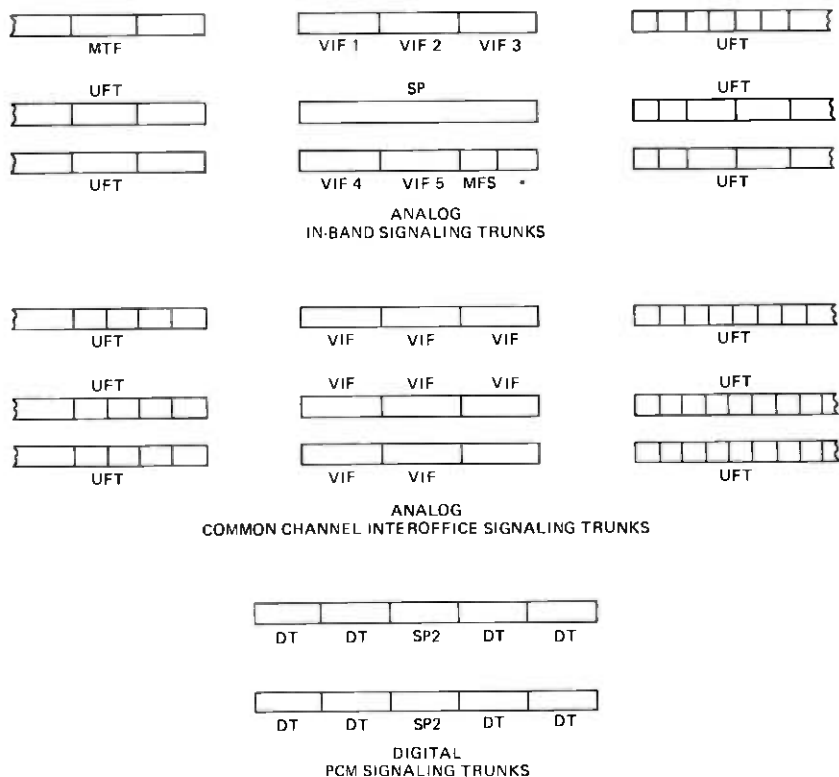


Fig. 20—MF signaling frame.

The optional SMAS allows one craftsperson at the 51A test position to gain access to both the switching interface (VIU) and transmission facility (carrier or cable) interface by a simple dial-up procedure or through CMS. Having both points readily available, the craftsperson can rapidly sectionalize trunk troubles within the building and achieve fast repair and restoral.

### **5.2 Floor plans**

While full connectorization demands effective control of the floor plan to avoid cable-rack congestion and cable routing problems, a completely



\* LOCATION FOR ECHO SUPPRESSOR TERMINAL OR MISCELLANEOUS FRAMES AS REQUIRED

Fig. 21—Basic core area layouts.

fixed floor plan would be too inflexible to meet network requirements. The modularity of the VIF, SP1, SP2, and DT made possible the creation of fixed core areas. These cores effectively confine and control cabling to near optimum but retain the flexibility to cope with wide variability in facility mix and growth patterns.

### 5.2.1 Analog basic core

The basic analog core illustrated in Fig. 21 combines five VIFs with an SP1 and, when required, an MF receiver and transmitter frame (MFS). SP1 provides signaling terminations for 4080 trunks and 2048 miscellaneous scan and distribute points. Each VIF mounts seven working VIUs, terminating 840 VF channels. Five VIFs mount a total of 35 active VIUs but one VIU is needed for the MFS and other service (nontrunk) circuits. The remaining 34 VIUs provide 4080 trunk appearances exactly matching the SP capacity.

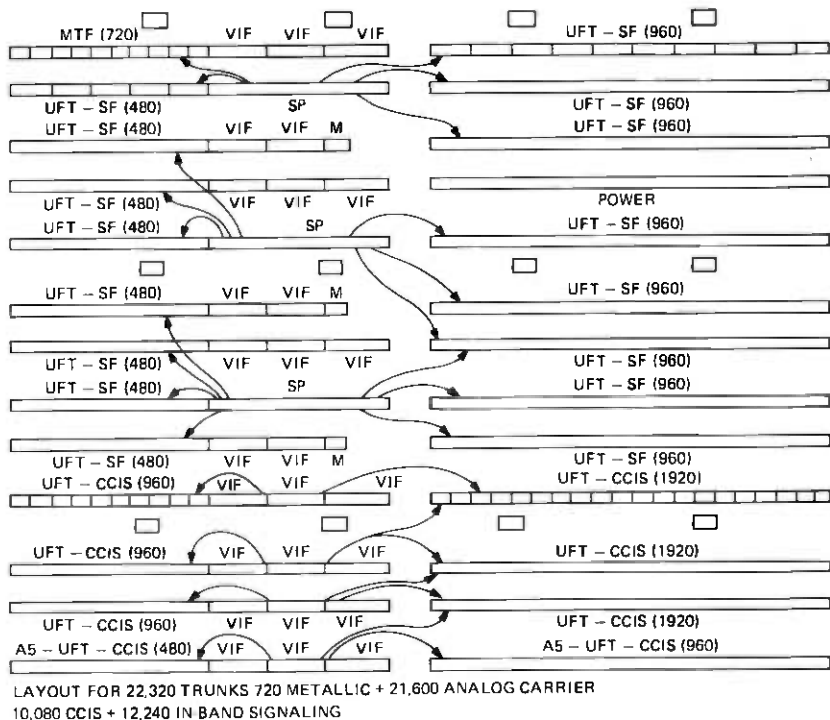


Fig. 22—Analog floor plan.

With UFT and/or MTF flanked on either side of the core, the bulk of the cables will either simply run longitudinally within the frame-mounted cable rack or make one transverse run cross-aisle to an adjacent lineup. Right- and left-hand feeds minimize cable crossover and pileup, which lead to congestion. Trial layouts and experience indicate that cable length should rarely exceed 80 feet.

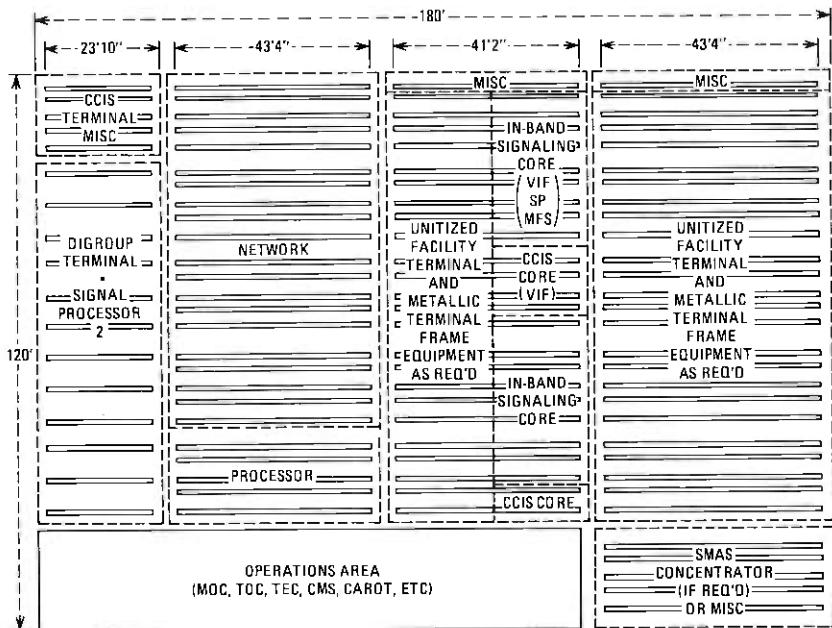
### 5.2.2 CCIS basic core

Signaling information for CCIS trunks is carried on a common data channel so the per-trunk SP signaling appearances are not required. The core therefore reduces as shown in Fig. 21. The resulting higher density compared with the in-band signaling core is compatible with the higher density achieved in the CCIS UFT. Retaining the same dimensions for the CCIS core allows both types of cores to be mixed as required to suit growth patterns.

### 5.2.3 Digital core

The basic digital core shown in Fig. 21 consists of four DT frames and one SP2 frame and serves 3840 MF or dial-pulse signaling trunks.





HYPOTHETICAL FLOOR PLAN LAYOUT APPROXIMATELY 82,000 TRUNKS  
 40,000 ANALOG (40% CCIS) 42,000 DIGITAL (25% CCIS)

Fig. 23—Overall floor plan.

In addition, one SP2 may also serve up to 11,520 CCIS trunks utilizing 12 additional DTs.

If the number of miscellaneous scan and distribute points required for the office (for such functions as office alarms, analog echo suppressor control, network management, etc.) exceeds the number provided by the SP in the analog cores, then a supplementary matrix frame may be added to the SP2. An office with a high percent of digital MF trunks may require supplementary matrix frames to terminate an adequate number of MFS frames.

### 5.3 Example floor-plan layouts

A hypothetical layout of analog core areas in No. 4 ESS is presented in Fig. 22. This compact arrangement serves 22,320 trunks (45 percent CCIS) in an area of approximately 6200 square feet (100 × 62). The directed lines indicate the cabling patterns for the SP-to-UFT connections. Similar patterns apply for the VIF-to-UFT connections.

In Fig. 23, the previous example is expanded to illustrate a complex as it might appear on one floor in a central office building. The trunk capacity is approximately 82,000 with 40,000 analog trunks (40 percent CCIS) and 42,000 digital trunks (25 percent CCIS). In the analog core area,

the layout of the previous example was taken to be an initial installation. This example shows how growth units can be added in an orderly and efficient manner. It may be noted that the CCIS and inband signaling cores become mixed in this growth process.

The DT area indicates that the frame layouts are spread to maintain a suitable heat load for conventional air conditioning. The lineups are located to align with network frame lineups to facilitate cabling passing between the two areas.

The trunk capacity of the area shown in this example, approximately 21,600 square feet (180 × 120), is influenced by the percentage of CCIS trunks. If there were no CCIS trunks, for example, the capacity would reduce from 82,000 to approximately 72,000.

If traditional means for providing toll terminal equipment were used, this overall floor-plan arrangement for an 82,000-trunk office would change substantially. The floor space requirement for toll terminal equipment would approximately double if 7-foot frames were used and increase more than 40 percent for 11-foot, 6-inch frames. Over one million distributing frame terminations would be required with the installation of over one-half million cross-connect wires. One physical distributing frame would probably not be usable because of its excessive length (over 300 feet for 7 foot frames and 185 feet for 11-foot, 6-inch frames). Multiple frames would require interconnection tie cables. The number of cables required would approximately double and the average length of cable would increase by a factor of about six. Administration and maintenance of such large distributing frames and of the equipment which would be spread over a much larger area could be expected to substantially increase work force requirements.

## VI. SUMMARY

In this paper we have addressed the interface between the transmission facilities and the No. 4 ESS. We have discussed the terminals, equipment arrangements, floor plans, etc., that collectively make up the interface between the facility terminations at the distributing or cross-connect frames and the switch itself. Seen from the switch, the transmission interface consists of serial PCM links using the DS-120 format, each carrying 120 two-way voice-frequency channels. We have described the two new terminals—DT and VIF—that interface digital and analog transmission facilities, respectively, to the DS-120 links.

The interface for digital facilities is particularly clean: the DT is the only frame required between the DSX-1 and the switch. There is no per-trunk circuitry; even signaling information is exchanged between the DT and SP2 on a 2-Mb/s multiplexed link.

We have seen that the interface for analog facilities has also been streamlined. Viewed from the facility, a standard interface is provided

by the four-wire VF transmission channel at the VIF and the four-wire E and M signaling port of SP1. Unitized facility terminals have been designed to provide in a single package all the required functions between the basic facility distributing frames and the VIF-SP1 interface. This analog arrangement has substantially reduced both interframe, per-trunk cabling and distributing frame terminations.

We have presented the terminal equipment arrangement for a typical system and seen that the combination of standardized interfaces, unitized terminals, modular floor plans and the nature of the switch itself results in a rather dramatic reduction in floor space, cabling, and distributing frames, when compared with traditional offices.

In closing, it is worth noting that the DT and VIF represent the first generation of what may be a line of terminals that combine classical transmission and switching functions and/or techniques. Both DT and VIF, in addition to their transmission roles in providing multiplexing, conversion, framing, etc., have access to the switching system processor. Both frames incorporate extensive maintenance hardware that functions with the processor access to achieve a high level of fault detection, diagnosability, and reconfigurability. Further applications of processor access would seem possible. Lastly, we note the disappearance of the single-voice channel as the switching/transmission interface. In particular, while single-channel access is always provided via the switch and maintenance systems, the DT itself provides no physical single-channel appearance.

## REFERENCES

1. J. Janik, et al., "No. 4 ESS: Peripheral Systems," B.S.T.J. (this issue).
2. R. E. Staehler, "1A Processor: Organization and Objectives," B.S.T.J., special issue on 1A Processor, 56, No. 2 (February 1977), pp. 119-134.
3. J. O. Becker, et al., "1A Processor: Technology and Physical Design," B.S.T.J., special issue on 1A Processor, 56, No. 2 (February 1977), pp. 207-236.
4. K. E. Fultz and D. B. Penick, "The T1 Carrier System," B.S.T.J., 44, No. 7 (September 1965), pp. 1405-1452.
5. M. T. Smith, Jr., et al., "No. 4 ESS: Data/Trunk Administration and Maintenance," B.S.T.J., this issue, pp. 1203-1238.
6. H. H. Henning and J. W. Pan, "D2 Channel Bank," B.S.T.J., 51, No. 8 (October 1972), pp. 1641-1657.



## **No. 4 ESS:**

# **System Power**

By H. J. LUER and R. OSTAPIAK

(Manuscript received July 26, 1976)

*The No. 4 ESS system is powered from a 140-volt battery plant. Modular dc-to-dc converters located in system frames are used to change the 140 volts to the many well regulated voltages needed by the ESS circuitry. The 24- and 48-volt power required by the system is provided by bulk dc-to-dc converters with 140 volts as an input.*

*The modular converters require no field adjustments, are pluggable, and contain many alarm and shutdown features, some of which are routinely tested by the system. The 24- and 48-volt bulk converters are available in 2.5- and 5-kilowatt sizes and, similar to the modular converters, contain many alarm and shutdown features for the protection of their 24- and 48-volt buses.*

*All converters are designed for low EMI (electromagnetic interference) emission and to operate in a room ambient of 0 to 50°C with in-frame ambients as high as 75°C. As a function of the output voltage, efficiencies range from 65 to 85 percent. Most converters regulate remotely with typical end-of-life regulation performance of better than  $\pm 2.5$  percent.*

## **I. INTRODUCTION**

### **1.1 Basic powering arrangement**

Although No. 4 ESS requires less power per trunk than its predecessor No. 4A crossbar, because of its larger trunk capacity, total power required for a typical No. 4 ESS office (over 500 kW) is larger than that for a typical 4A office. The most significant difference in the power required by the two systems is that most of the power used in the 4A equipment is taken directly from a 48-volt battery plant, whereas most of the power for No. 4 ESS has to be processed to provide the many very tightly regulated voltages needed by its electronics.

Since most of the power for No. 4 ESS has to be processed, 140 volts dc was chosen as the basic power arrangement for the system. This was done to take advantage of lower battery plant cost and reduction in size and complexity of power distribution cabling that a high-voltage battery system affords for large electronic complexes.

In the powering arrangement for No. 4 ESS, 140 volts is distributed to in-frame modular dc-to-dc converters which provide the many well regulated voltages needed by the system circuits. The 24- and 48-volt power requirements of the system are provided by centralized dc-to-dc converter plants which replace the normal battery plants at these voltage levels. The cost of the 24- and 48-volt converter plants is the only economic penalty that has to be paid for the use of the 140-volt powering arrangement. However, this penalty is more than offset by the savings realized in the cost of the battery plant and distribution cabling.

In the design of the required power processing equipment for the No. 4 ESS system, most of the problems encountered were with the power for the switching machine, including the IA Processor. As a result, the remaining part of this paper deals only with the power for the switching machine. Power for the remaining systems in the ESS office, all of which is obtained from the common 140-volt battery plant, is processed using identical or similar converters.

### ***1.2 Converters for the Switching Machine***

A total of fifteen in-frame modular dc-to-dc converter designs, ranging in output voltage level from  $-28$  to  $+28$  volts, and three bulk converter designs are used to supply all the voltages needed by the ESS circuits. To simplify the maintenance and minimize the cost of these converters, their designs were based on the following four standard basic power circuits: single-ended switching transistor circuit for output power levels of less than 30 watts, a ferroresonant circuit or a two-transistor pulse-width controlled circuit for output power levels between 30 and 350 watts, and a Silicon Controlled Rectifier (SCR) circuit for power levels above 1 kW.

### ***1.3 Use of In-Frame Converters***

To provide required voltages for the ESS circuits, in-frame modular converters are used instead of centralized voltage power plants because of system-required redundancy, required protection of the backplane wiring, and very tight voltage tolerances at the loads.

With in-frame converters, a loss of any one converter will result only in the loss of that frame or part of that frame. Since all critical frames in the system are duplicated, the use of an in-frame converter powering

arrangement ensures that a loss of any one converter will not seriously affect the service capability of the system.

All signal-carrying conductor paths in the backplane are limited to under 6 amperes of continuous current. To protect them from overcurrent damage in case of an accidental cross to a voltage bus, all converters supplying power to the backplane are either rated below 5 amperes or the minimum steady-state load connected to them is within 5 amperes of their current ratings. All converters contain high-current shutdown circuits that are set to operate just above their rated output current levels. Any type of fault at the load that conducts over 5 amperes will result in the shutdown of the converter supplying that current without causing damage to the backplane.

The voltages used to power the integrated circuits of the system are regulated to very tight limits. At low voltages and high currents, the distribution voltage-drops quickly become an appreciable percentage of the output voltage levels. Therefore, converters supplying critical loads regulate remotely and are limited to low output power levels to minimize the distribution voltage drops between the converters and the loads they power. The loads for each converter are concentrated in a very small area to make remote regulation effective.

## II. THREE-VOLT POWER FOR TTL LOGIC

To meet the required speed and accuracy of No. 4 ESS, the converters and the distribution system used to provide power to the 3-volt TTL (transistor-transistor logic) have been designed to meet 2.90 to 3.10 volt limits at the chips over a 20-year period for all line, load, temperature, and aging variations, including any voltage transients produced by a rapid change in the load current of up to  $\pm 10$  percent. To meet these very tight limits, a precise common reference voltage is provided to all the 3-volt converters in a frame. With this arrangement, variation in the reference voltage does not have to be included in the 2.90 to 3.10 volt limits since its effect on the 3 volts is identical for all the TTL chips in the frame.

To meet the 2.90 to 3.10 volt requirements at the logic chips, converters supplying the 3-volt power use remote sensing and come in 4- and 8-ampere sizes to minimize distribution voltage drops. To keep voltage transients caused by load switching down to reasonable levels, low-inductance cables are used for the distribution of the 3-volt power to the backplane. In addition, for each 4 amperes of load, a 4000  $\mu\text{F}$  cluster of capacitors is used to filter the ac load current variations induced by the logic. These capacitors are located on a printed wire board near the logic circuits. The board is connected to the backplane 3-volt bus by a connector. Finally, each plug-in card accommodating 3-volt logic circuits

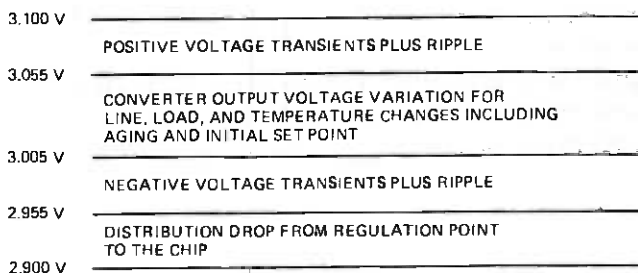


Fig. 1—Voltage allocation for the permitted 3-V voltage band.

contains 0.4  $\mu\text{F}$  ceramic capacitors to provide high-frequency filtering of the load current.

Figure 1 shows the allocation of the 200 mV permissible deviation of the 3 volts among various voltage-variation contributors for which the converters and the 3-volt distribution system were designed to guarantee the 2.90 to 3.10 volt limits.

### III. IN-FRAME CONVERTERS

#### 3.1 Single-ended switching transistor converter

The single-ended switching transistor power circuit is used for all converter designs under 30 watts because of its cost effectiveness at low power levels. The basic block diagram of the circuit is shown in Fig. 2.

The power transistor is switched ON and OFF at a fixed 20-kHz rate by the control circuit through an isolation transformer. With the transistor ON, input voltage is applied across the primary winding of the transformer which reverse-biases the rectifying diode and results in an increasing ramp of current through the primary inductance of the transformer. When the transistor is turned OFF, the energy stored in the magnetizing inductance of the coil is discharged through the secondary

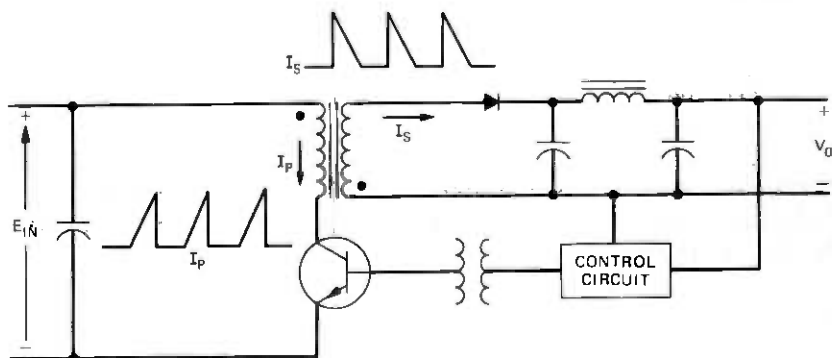


Fig. 2—Basic block diagram of a single-ended switching transistor converter.



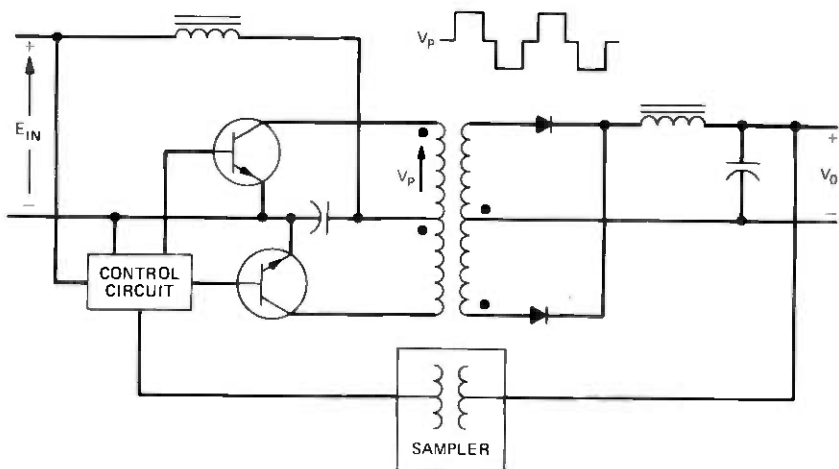


Fig. 3—Basic block diagram of a PWC converter.

winding into the output filter as a decreasing ramp of current. Output voltage regulation is achieved by varying the ON time of the transistor switch.

For ease of maintenance, for manufacturability, and to achieve long-term output-voltage stability, no adjustable components are provided in the converter. Required settings of the output voltage and alarm and shutdown trip levels are achieved by the use of very precise reference voltage sources and precisely trimmed thin-film resistor networks combined in a single hybrid integrated circuit.

To keep the radiated noise of the converter within the ESS EMI requirements, the following steps were taken in the design of the converters:

- (i) RC networks are used across all fast-switching power devices.
- (ii) Bypass capacitors are provided at the converter connector between all power leads and frame ground.
- (iii) All ac loops are minimized by careful physical layout of the converter circuits.
- (iv) Shielding is provided by the converter frame housing.

(Some or all of the above EMI suppression techniques are also used in the ferroresonant and PWC converter designs described in the next two subsections.)

### 3.2 Two-transistor pulse-width controlled converter (PWC)

A block diagram of the PWC converter circuit, which was designed for output power levels of 30 to about 350 watts, is shown in Fig. 3.

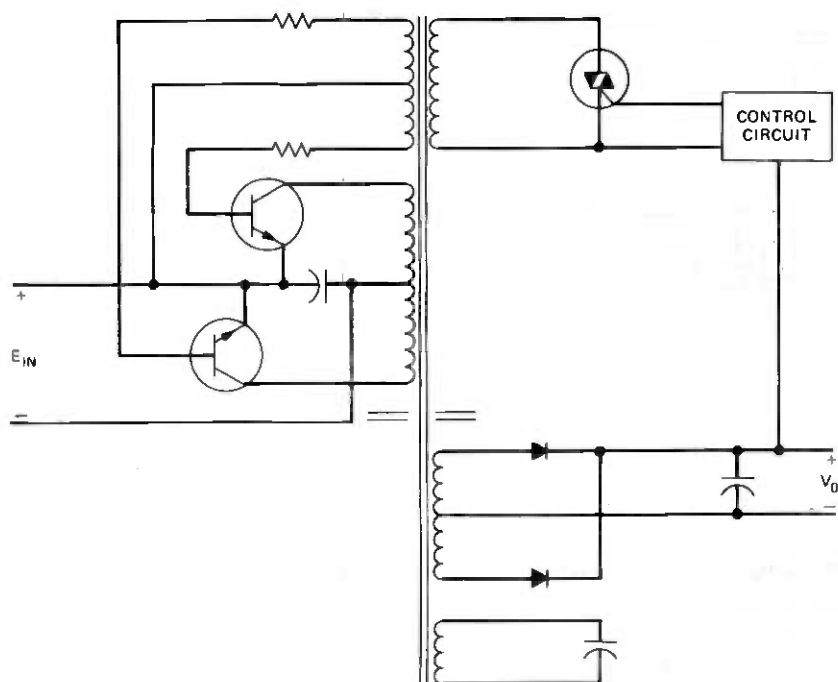


Fig. 4—Block diagram of a ferroresonant converter.

In the circuit, the power transistors are alternately switched ON and OFF at a fixed 20-kHz rate by the control circuit to produce a pulse-width controlled square wave voltage across the transformer primary. After this ac voltage is converted to an appropriate level by the transformer, it is rectified and filtered to produce the required dc output. Output voltage regulation is achieved by controlling the ON time of the power transistors.

Special features of the converter include high efficiency, fast response to load transients, and automatic symmetry correction of the power-transistor collector currents.

### 3.3 Ferroresonant converter

The ferroresonant converter was designed for output power levels of 50 to 200 watts. Its primary features are its simplicity and good EMI emission performance. A block diagram of a ferroresonant converter circuit is shown in Fig. 4.

The basic converter circuit is composed of a two-transistor self-oscillating power circuit and a transformer designed to have a high leakage inductance between the primary and secondary windings. An ac ca-

pacitor is connected across one of the secondary windings. Its capacitance is chosen to be at resonance with the transformer leakage inductance at the operating frequency of the converter, typically 150 Hz. This makes the secondary-winding voltages sensitive to the operating frequency of the power circuit—a feature which is used to regulate the output voltage of the converter. The operating frequency of the circuit is controlled by short-circuiting the control winding of the transformer, which is closely coupled to the base drive windings, at the desired termination time of each half cycle.

### **3.4 In-frame converter features**

All in-frame converters have been designed with the following features for the protection of the circuits they power and for proper interface with the ESS: (i) high-voltage shutdown, (ii) high-current shutdown, (iii) high and low out-of-range output-voltage alarms, (iv) system-controlled test of alarm circuits, and (v) automatic input current inrush control. In addition, all low-voltage converters are provided with an output voltage clamp (a high-wattage zener diode) for the protection of the loads in case of an accidental cross between high- and low-voltage buses.

In the event of a high-voltage or high-current shutdown and during an out-of-range alarm condition, alarm signals are transmitted to the ESS and a visual alarm indicator is activated on the face of the converter. During the alarm test, which can be initiated automatically by the ESS or manually by the use of a frame switch, the entire out-of-range alarm circuitry of the converter is checked for proper operation and, if everything is functioning properly, an all-pass signal is sent to the system.

In addition to having the above features, all in-frame converters have been designed for low EMI emission, to regulate remotely, and to operate in an aisle ambient temperature of 0 to 50°C with in-frame ambients as high as 75°C. The converters require no adjustments in the field and are pluggable for easy maintenance.

### **3.5 In-frame converter performance characteristics**

Typical performance characteristics of the three basic in-frame converter designs just described are shown in Table I. In this table, regulation pertains to static output-voltage regulation for all load, line, and temperature variations, transient voltage covers peak output voltage transient at the point of regulation for a 10 percent step change in the output load, efficiency is presented for various output-voltage levels, and EMI emission covers radiated noise levels of the converters referenced to  $15 \mu\text{V/m}$  at a distance of  $\lambda/2\pi$ , where  $\lambda$  is the wavelength in meters at the frequency of measurement, over the frequency range of 50 kHz to approximately 100 mHz.

Table I — Typical in-frame converter performance characteristics

|                             | Regulation | Transient Voltage | Efficiency           | EMI Emission  |
|-----------------------------|------------|-------------------|----------------------|---------------|
| Single Transistor Converter | 0.2%       | 1.0%              | 70% -3 V<br>75% -5 V | -20 to -40 dB |
| PWC Converter               | 1.0%       | 1.0%              | 75% -5 V<br>80% -9 V | -20 to -40 dB |
| Ferroresonant Converter     | 1.5%       | 5.0%              | 65% -5 V             | -30 to -50 dB |

#### IV. THE SCR CONVERTER

Converter designs above the 1kW range generally employ SCR power circuits because of the high power-handling capability of the SCR switches. A block diagram of the SCR circuit used for the 24- and 48-volt bulk converters is shown in Fig. 5.

SCRs Q1 and Q2 are alternately triggered on by the control circuit. When Q1 is turned on, the input voltage plus the initial voltage across capacitor C1 is applied across the primary winding of the transformer. With transformer winding polarities as shown, the rectifying diode (CR1) is initially reverse-biased, resulting in a sinusoidal current flow through Q1 because of the resonant action of C1 and the inductance of the primary winding. Sinusoidal current flow through Q1 continues until the voltage across C1 reverses and becomes equal to  $E_{in} + nV_0$ , at which point the rectifying diode becomes forward-biased. With the diode forward-biased, the constant voltage across the output filter capacitor clamps the C1 voltage to  $E_{in} + nV_0$ , forcing the current through C1, and hence Q1, to go to zero. The energy stored in the primary inductance of the coil is then transferred through the diode into the output filter as a ramp of current. Similar circuit operation to that just described occurs during the next half cycle when Q2 is fired. Output voltage regulation is achieved by controlling the firing frequency of the SCRs.

In an actual application, the SCR converters are used to form power plants consisting of an array of converters operating in parallel together with the necessary plant alarm circuits and distribution fuses. To achieve the needed reliability of the plants, the number of converters connected in parallel is always one more than required to power the maximum load of the plant. With this arrangement, if any one converter malfunctions, it is automatically disconnected from the output bus without affecting the output voltage of the plant.

For the protection of the plant output voltage and the loads, the following features were designed into the converter: (i) high-output-current limit, (ii) high-voltage shutdown, (iii) high-current shutdown, (iv) reverse-current shutdown, (v) automatic output-disconnect switch, and

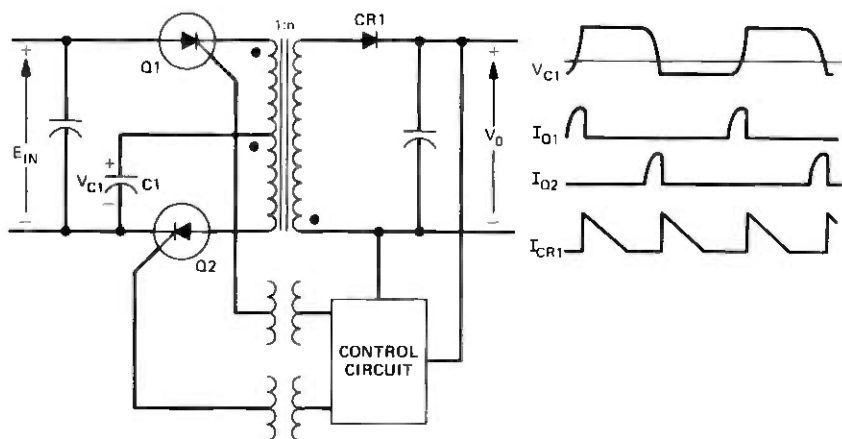


Fig. 5—Basic block diagram and waveforms of the SCR converter.

(vi) low-current alarm which is activated when the converter is grossly misadjusted or is incapable of delivering its rated output current. In addition, approximately 0.5 farads of capacitance are included in the output filter of the converter to help the plant in clearing faults at the load by operating distribution fuses without greatly affecting the output voltage of the plant.

Typical performance characteristics of the converter are as follows: (i) efficiency—83 percent for 24-volt output and 86 percent for 48-volt output, (ii) static output voltage regulation for load, line, and temperature variations—0.3 percent, (iii) peak output voltage transient for 10 percent step change in load—0.2 percent, and (iv) EMI radiated noise level—15 to 30 dB below  $15 \mu\text{V/m}$  at  $\lambda/2\pi$  over the frequency range of 50 kHz to approximately 100 MHz.

## V. PHYSICAL DESIGN CONSIDERATIONS

### 5.1 In-frame converters

Two major factors determined the overall physical features of the modular power supplies and their application in the switching frames: (i) isolation of 140 volts for personnel protection and (ii) thermal management.

While 140 volts dc is not considered to be a particularly hazardous voltage, it should be treated with respect. Therefore, it was deemed prudent to isolate and identify those areas in the frames where 140 volts is present. The modular converters are mounted in special shelves at the bottom of the switching frames immediately above the in-frame distribution fuses. The input power is fed down the frame upright, through the fuses, and then directly to the converter input connections at the rear

of the frame. All exposed 140-volt wiring is covered by an insulating panel mounted to brackets integral to the shelf, and appropriate warning labels are affixed.

The power shelves are themselves modular and can readily be adapted to accept various combinations of modular converters as needed by the individual frames. Two power shelves can be installed, one above the other in a frame, or one can be used if its capacity is sufficient. An air deflector is installed above the topmost shelf where necessary to divert hot air out the rear of the frame away from the switching circuits.

Because of the need to localize the converters, to provide EMI emission protection and 140-volt personnel protection, and because of the general packaging density of the converters themselves, there is little free air flow through the converters. All major heat sources are mounted on heat sinks which are affixed to the front of the converters with their heat-exchanging fins in the aisle. In this way, a predictable environment is assured and thermal performance was determined with considerable confidence. Figure 6 illustrates the shelves with a variety of power supplies installed.

An exception to this general physical arrangement is the 3-V, 2-A converter used to power bus drivers and bus receivers. Circuit considerations require that this converter be located close to the bus at the top of the bay, and be mounted in the same 80-type apparatus mountings as the bus. Since no 140-volt power is available at that location, this converter is powered from the 24-volt bulk converter output—the only low bulk voltage utilized in all bays.

## **5.2 Bulk converters**

Early in the development of the bulk converters, it became apparent that heat removal would be the major physical design problem. In order to be acceptable from a floor space point of view, the bulk converter frames would be required to dissipate far more heat than any other frame in the No. 4 ESS. Reliability considerations prohibited the use of forced convection since the best available blowers or fans have failure rates many times greater than that required for the entire converter.

The design has evolved through several stages. The 24-volt converter plant provides 5000 watts of output power per 2-foot, 2-inch bay with internal dissipation of 1200 watts. The two converters are mounted one above the other with an air-deflecting baffle in between. Heat removal is via large internally mounted heat sinks.

The later 48-volt converter plant, shown in Fig. 7, provides 10,000 watts of output power per 2-foot, 2-inch bay with internal dissipation of 1800 watts. This increase in dissipation was made possible by mounting the converters side-by-side so as to allow both converters ac-



Fig. 6—No. 4 ESS frames with power shelves. One 3-V, 8-A converter is shown partially withdrawn at bottom.

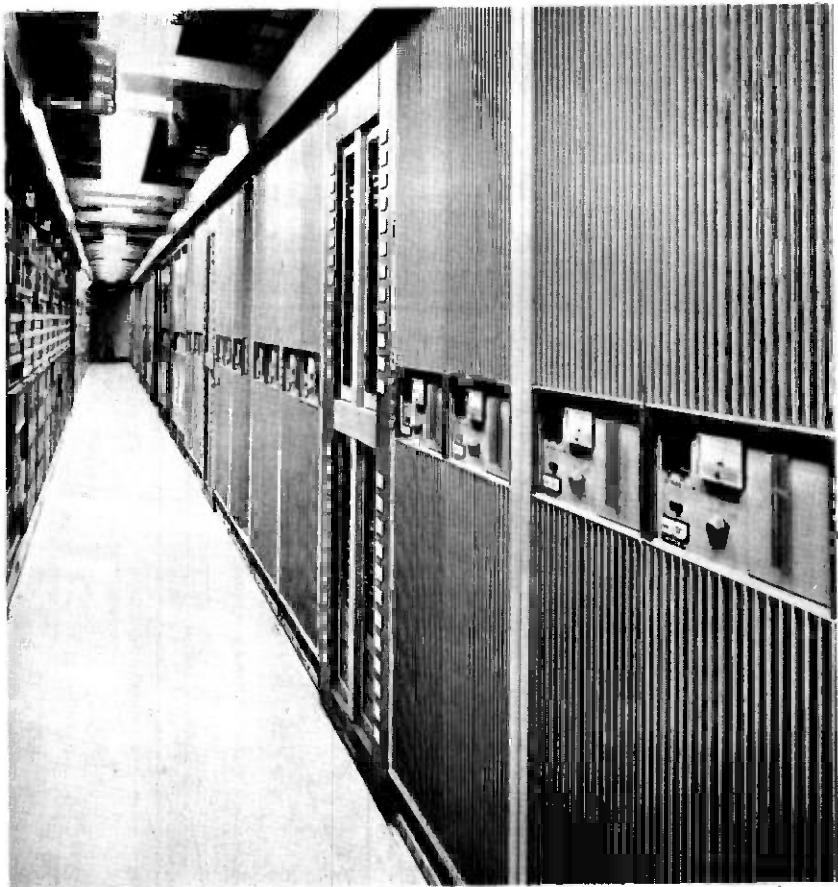


Fig. 7—625B converter plant lineup in Kansas City No. 4 ESS office.

cess to optimum cooling air, and by the provision of massive top-to-bottom heat sinks covering both front and rear surfaces. Air vents at top and bottom provide for additional cooling via two large internal heat sinks. All heat-producing elements are mounted directly to the heat sinks with electrical isolation provided by a heat-conductive epoxy lamination within the heat sink.

## VI. SUMMARY

Modern-day electronic systems require a variety of tightly controlled voltages for proper operation. As a result, power equipment has become an important part of all new system designs. Typically, it occupies up to 25 percent of total equipment volume and represents 10 to 20 percent of total system cost. Major problems encountered in the design of the



power arrangement for No. 4 ESS were the need for a large amount of power at very low voltages and high power density inside the frames. These requirements necessitated special converter and power distribution designs that could operate reliably at very high temperatures. To satisfy all of the voltage requirements of the switching machine, as well as those of the remaining systems in the No. 4 ESS office, a standard line of converters was developed to minimize the development effort and the cost of the equipment. In addition, since most of the power for the No. 4 ESS office had to be processed and could not be used directly from battery plants, 140 volts was chosen as the basic power arrangement for the office to reduce cost and simplify power-distribution cabling. Cost savings realized with the 140-volt versus a comparable 48-volt power arrangement for a typical No. 4 ESS office are approximately 15 percent of the total battery plant and distribution-cabling costs. This saving varies with the cost of copper.

100  
101  
102  
103  
104

## **No. 4 ESS:**

# **Software Organization and Basic Call Handling**

By T. J. CIESLAK, L. M. CROXALL, J. B. ROBERTS, M. W. SAAD,  
and J. M. SCANLON

(Manuscript received July 12, 1976)

*The No. 4 ESS has the majority of its control, administrative, and maintenance functions implemented by software. This software is organized into an operating system that provides scheduling and interrupt handling, operational programs that perform the required call-handling and administrative functions, and maintenance programs that provide diagnostic capability for trunks and switching hardware. This paper provides a general description of the design concepts and organization of the No. 4 ESS software, dealing in some detail with the operating system. Functional capabilities, design constraints, and organization of the call-handling software programs are also covered.*

## **I. INTRODUCTION**

The No. 4 Electronic Switching System (No. 4 ESS), a stored program switching system, has most of its required control, administrative, and maintenance functions provided via program. The initial generic program for No. 4 ESS is the largest initial program developed to date for Bell System Electronic Switching Systems. It consists of approximately 400,000 instructions and 400,000 words of diagnostic tests requiring in excess of 1.4 million 26-bit words of system storage.

In order to meet the overall design objectives for No. 4 ESS, four major objectives were set for the initial program:

- (i) Efficiency in real time.
- (ii) Simple man-machine interfaces.
- (iii) Defensive design.
- (iv) Ease of modification.

Meeting these objectives required a unified software/hardware impact on the system architecture and the introduction of a number of improved software design and implementation concepts.<sup>1</sup>

The generic program concept,<sup>2</sup> used in earlier ESSs, will be followed in the introduction of additional features to No. 4 ESS. Each new generic will contain feature additions and will be built on the previous generic's program base.

## II. SOFTWARE ORGANIZATION

The No. 4 ESS software consists of three basic categories of programs. They are:

- (i) Operating system programs, which provide task scheduling, interrupt handling, and man-machine input/output interfaces.
- (ii) Operational programs, which provide call-handling and administrative capabilities.
- (iii) Maintenance programs which provide trunk maintenance and frame diagnostics.

Components of both the operating and maintenance categories which are common to the other application of the 1A Processor (i.e., No. 1A ESS<sup>3</sup>) are designed as a separate package called 1A Processor common software.<sup>4</sup> These programs perform the functions of initialization and system restart, fault recovery, and diagnostics for 1A Processor hardware, and input/output for the various I/O devices. This common software represents approximately 150,000 instructions and 215,000 diagnostic tests.

All programs are executed from core (program store or in some cases call store), although their resident storage medium may be tape, disk, or core store. As indicated in Fig. 1, if the resident storage medium is not core store, programs are paged into core prior to execution. The basis for establishing the resident storage medium for a particular program is determined by response time and frequency of execution requirements. For example, call-handling programs are core resident, diagnostic tests are disk resident, and system update programs are tape resident. Core-resident programs are backed up on both disk and tape; disk-resident programs are backed up on tape; and tape-resident programs have no on-line backup.

The No. 4 ESS software is packaged into approximately 800 individual PIDENTs, or loadable modules, ranging in size from several hundred to several thousand instructions. Each PIDENT generally implements one of many tasks required to perform an entire system function. An example of a PIDENT is the program that performs network connections, one task required as part of the call-handling function.

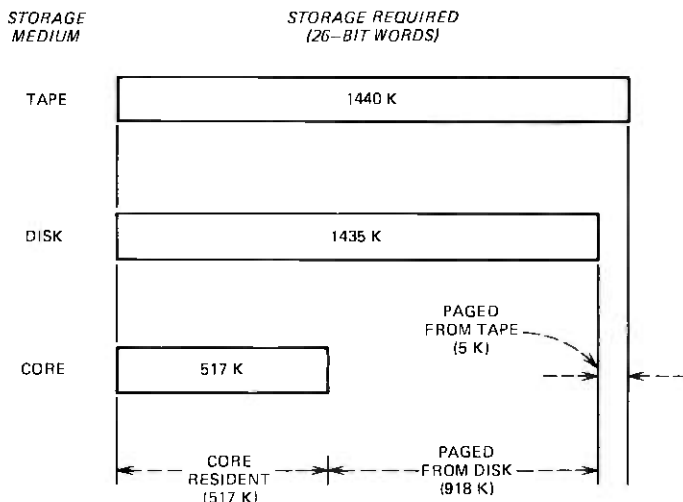


Fig. 1—Program storage media.

## 2.1 Operating system

The No. 4 ESS operating system can be conceptualized as a structure in three layers: executive control and audits, maintenance interrupts, and system integrity. Each layer is further subdivided into a number of activity levels that establish task priority within a layer. This structure is similar to that used in No. 1 ESS.<sup>5</sup> The principal difference is that timed interrupts are not used to perform high-priority operational functions. These functions are instead performed by using the interject approach. Figure 2 shows the relationship between the three layers of the operating system and, in decreasing order of priority, lists the 14 activity levels provided.

*Executive control.* The lowest layer of the operating system is the executive control, which performs job scheduling and sequencing in a normal, fault-free environment. Executive control schedules two levels of system activity: base level and interject level. As shown in Fig. 3, base level is a simple scheduling loop; the last task executed is succeeded in the scheduling order by the first. Base level tasks receive equal scheduling priority with each task served once per base. The time to complete one cycle of the base level schedule, called the "base-level cycle," varies with the instantaneous load on the system. It typically ranges from 11 milliseconds (ms) with no traffic load, to about 35 ms at a traffic load of 500,000 calls per hour. Examples of base-level tasks are the processing of new trunk service requests, administrative tasks (such as traffic measurements and network management), and frame diagnostic activity. Base-level programs are designed to divide their processing into a maximum of 3-ms segments, returning to executive control at the end

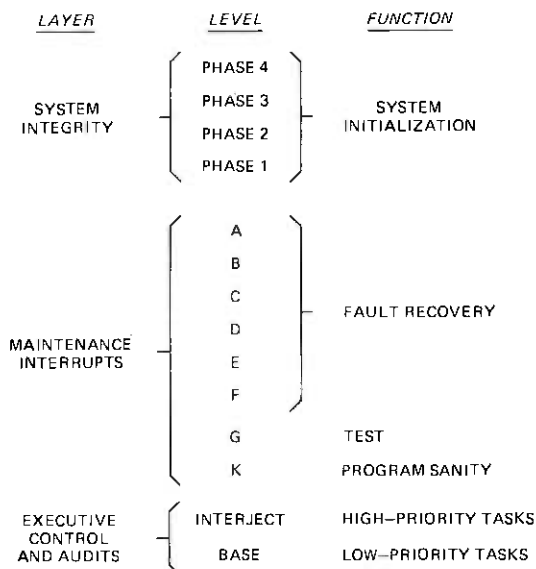


Fig. 2—Program activity levels.

of each segment. A task will be reentered on subsequent base-level cycles until completed. This allocation of 3 ms per task per base cycle is required to meet cross-office signaling delay requirements and prevents a single task from dominating system resources. This approach also helps limit the variance of the base-level cycle, which is important from the standpoint of designing stable real-time overload detection and control mechanisms.

Interject level is a scheduled programmed interruption of base-level activity, nominally every 10 ms, to do high-priority system tasks such as critical call-handling timing. As illustrated in Fig. 3, when each base-level task returns to executive control after completing 3 ms of processing, a check is made to determine whether 10 ms has elapsed since the last interject was serviced. If so, a transfer is made to the interject class of program tasks, which are linked together serially. At the conclusion of the final interject-level task, executive control returns to base-level processing. Since interject is a planned interruption of base activity, no extra overhead is required for saving and restoring processor registers. Also, the possibility of memory interwrite problems with other program activity levels is reduced when compared with a timed-interrupt approach.<sup>5</sup>

*Maintenance interrupts.* The second layer of the operating system provides detection and correction capabilities for hardware faults handled by maintenance interrupts. Maintenance interrupts (A through F) are triggered by the failure of some hardware fault detector (a parity

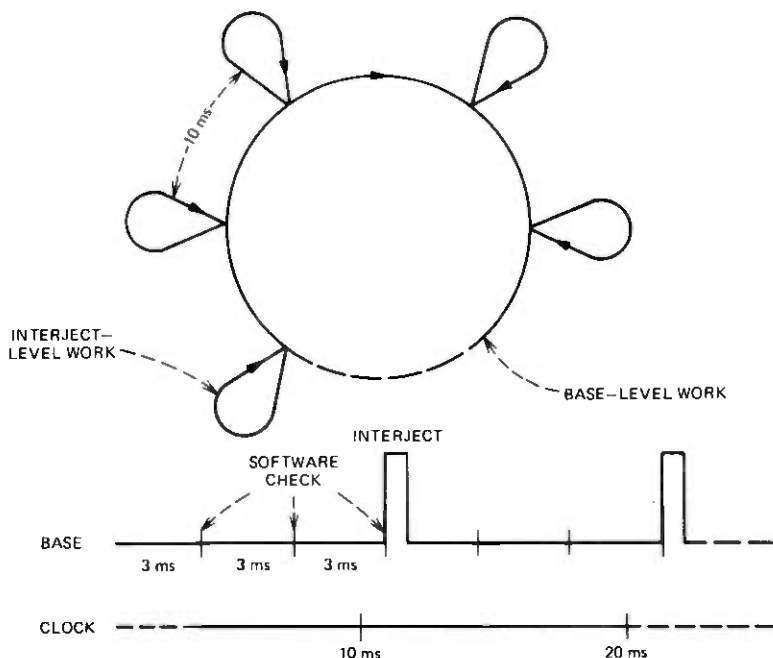


Fig. 3—Executive control.

check circuit, for example), and force the transfer of program control to the appropriate fault recovery program.<sup>6</sup> While the fault-recovery program is actively identifying and switching the faulty unit out of service, all other system operations are suspended. Fault-recovery programs are not segmented and will continue to completion unless a higher-level interrupt is triggered.

At the conclusion of the recovery action, normal system processing is restarted either at the point interrupted or at a "safe" starting point in the base-level cycle. Maintenance interrupts A through E are triggered in response to fault detectors resident in the 1A Processor. The F-level maintenance interrupt is reserved for units on the peripheral bus connected to the 1A Processor. These units include the network frames, signal processor, network clock, CCIS frame, and several 1A Processor units. G-level activity is restricted to such system troubleshooting tasks as obtaining data on complex field problems, and must be manually activated. K level provides an overall sanity check on the serving of 10-ms interject requests.

*System integrity.* The third and highest layer of the operating system, system integrity, provides a hierarchical structure of system initialization in four phases.<sup>6</sup> Although any phase may be manually selected and executed, automatic entry into the initialization sequence generally begins

at phase 1 and progresses in sequence, if required, up to phase 3. Phase 4 can be activated only by manual request. Each phase incorporates additional initialization measures and, as a consequence, further impacts calls. Phase 1 has no effect on call processing. Phases 2 and 3 maintain connections on stable calls but tear down calls which are actively being processed. Phase 4 initializes all calls in the system to an idle state.

## **2.2 Operational features**

The operational features of No. 4 ESS consist of call-handling capability for three signaling types: Multifrequency (MF), Dial Pulse (DP), and Common Channel Interoffice Signaling (CCIS); data administration software to modify the system translation data base; and surveillance software such as network management and traffic and plant measurements to monitor the efficiency of the No. 4 ESS system in switching calls.

Design of the call-handling software reflects heavy emphasis on efficiency, defensiveness, and ease of modification. The use of a single very large call register and a two-word trunk register, elimination of peripheral order time buffering, per-call event consistency checks on data, and extensive employment of high-level macros are a few of the techniques used to meet the call-handling design objectives. Section III of this paper provides further detail on the call-handling design.

The primary requirement on the design of the data administration software<sup>7</sup> was to provide a simple, reliable man-machine interface for altering the system data base. The use of a set of system-generated forms, displayed on a CRT, provides a highly reliable and easy-to-use "fill-in-the-form" approach to data administration and permits the use of straightforward English text in the process.

The design of the surveillance software for No. 4 ESS is characterized by interpretative-table-driven program designs that permit the easy alteration of the output formatting and data processing algorithms of the network management and traffic and plant measurement functions.<sup>8</sup> This technique allows new surveillance features to be provided by a process similar to updating the system data base and is not tied to new program issues.

## **2.3 Maintenance features**

The maintenance features of No. 4 ESS consist of trunk maintenance capabilities including manual and automatic trunk testing and frame maintenance provisions such as frame diagnostics and highly automatic trouble location procedures.

The trunk maintenance software system<sup>7</sup> controls the actions required for manual trunk tests via the 51A test position and automatic trunk tests



via code line calls. In addition, trunk maintenance software automatically collects a wealth of data on ineffective machine attempts in the office and performs threshold analysis on this data to identify marginal trunks not uncovered during routine trunk testing.

The diagnostic program design for No. 4 ESS is characterized by a highly structured design approach which consists of a table of diagnostic tests and their expected results, specified by macros for each frame type, and a general interpretative control program.<sup>6</sup> The tests are disk resident and are paged into core for execution. Unlike previous ESS designs, several diagnostics can be active at a given time and the maintenance-crafts force receives the identity of the packs to be replaced at the conclusion of the diagnostic instead of receiving trouble number requiring manual lookup.

### **III. CALL HANDLING**

#### **3.1 Introduction**

The call-handling programs control a call from origination to disconnection. These programs, either directly or indirectly, cause the performance of all the logical operations required to connect a call on an incoming trunk to the proper outgoing trunk, to supervise the connection, and to eventually disconnect the two trunks.

This section describes first the designs objectives for the call-handling programs and the capabilities of the No. 4 ESS. Next the hardware, data structures, and program structures that support call handling are described. The final subsection describes the operation of the call-handling program with a functional description.

##### **3.1.1 Design objectives**

The call-handling programs are designed to handle the three types of signaling provided by No. 4 ESS: MF, DP, and CCIS. Also, a means is provided for the collection of charging information on toll calls utilizing a Centralized Automatic Message Accounting (CAMA) feature. The call-handling design assumes that the No. 4 ESS will communicate only with other switching machines or operators but not directly with customer station equipment. Also, the initial No. 4 ESS call-handling design does not provide any special features required for gateway office operation or switching of international traffic.

In setting up the call-handling design for No. 4 ESS, a number of design objectives were established.

(i) A No. 4 ESS must meet specified traffic capacity objectives. The minimum required traffic capacities for No. 4 ESS are given in terms of switched attempts per hour, network CCS per hour, and trunk termi-

nations. *Attempt processing*: A switched attempt comprises recognition of an incoming call, establishment of a connection through the network, and control of the associated signaling functions. At a load corresponding to peak capacity, other normal machine functions must also be performed (e.g., maintenance, traffic administration, network management). False attempts (i.e., incoming trunk seizures with no digits pulsed), within the limits shown, should not reduce the switched attempt capacity. Minimum requirements are as follows: engineered switched attempt capacity—550,000/hr (based on 10 high day busy hour average 500,000, 10 percent peak day increase 50,000); false attempts—66,000/hr. *Switching network load*: Network load capacity is a function of the number of connections through the network and their holding times, averaged over an hour. It is measured in CCS/hr (or in erlangs) and must be related to a probability of blocking of new calls to be meaningful. Two load levels, corresponding to the design objective for a maximum engineered load and for a peak load, are most useful. Minimum basic requirements are as follows: engineered load (0.5 percent first trial matching loss), 1,000,000 CCS/hr; peak load (10 percent first trial matching loss), 1,700,000 CCS/hr. *Network terminations*: Network terminations are the connecting points for the incoming, outgoing, and two-way trunk circuits. The two-way trunk circuits are considered to have a single network-connecting point. The network also terminates service circuits, tone and announcement circuits, and test circuits. Minimum basic requirement: 107,000 terminations (includes service circuits).

(ii) It is of only slightly lower priority that No. 4 ESS meet certain specified performance objectives at its rated capacity. Table I gives the mean time for several (but not all) of these objectives.

(iii) The system performance under overload must be reasonable. Absolute throughput should not be degraded even if the offered load exceeds the system capacity. Although performance requirements are relaxed somewhat in overload, it is anticipated that the system capacity will be limited by the tightness of system performance requirements rather than real-time exhaustion.

(iv) It is important that the system have a high degree of reliability. Both hardware and software faults can affect reliability. Of concern here are the techniques that keep the call-processing system operative in the face of such errors.

(v) An important characteristic of a large call-processing system is simplicity. Simplicity is a difficult concept to define, since it takes different forms in different instances. For example, the utilization of subroutines can lead to simplicity since a subroutine to perform a function can be written and debugged once and then used by other programs. On the other hand, straight-line coding can lead to simplicity because one

Table I-Performance objectives

| Performance Measurement  | Mean Time |
|--|-----------|
| Seizure time [seizure on incoming trunk (ICT) to transmittal of beginning of wink]   | 60 ms     |
| Address time [receipt of last translatable digit to seizure of outgoing trunk (OGT)] | 100 ms    |
| Network path closure time (end of outpulsing to ICT-OGT connection)                  | 100 ms    |
| Cross-office answer time (receipt of answer on OGT to transmittal on ICT)            | 22 ms     |
| Immediate-start time (seizure on ICT to ready to receive first digit)                | 50 ms     |
| Response time (receipt of wink on OGT to transmittal of first address digit)         | 100 ms    |

does not have to keep track of transfers, data shifts, interfaces, etc. In fact, simplicity of a system may well come through the language in which it is written. One reason that simplicity is so important is that many other desirable properties follow from it: a simple system is easier to design, debug, make changes in, expand, etc.

(vi) Traditional objectives, such as efficient memory utilization and cost, have been considered, but the first concern has been given to the above considerations.

### 3.1.2 Signaling types

The address information (digits) that identifies the destination of a particular call and is transmitted between offices is classified according to the method of sending the information from one office to another. As mentioned previously, three types of signaling will be provided by the No. 4 ESS: Multifrequency (MF) pulsing, Dial Pulsing (DP), and Common Channel Interoffice Signaling (CCIS).

*Multifrequency (MF).* The No. 4 ESS is capable of sending MF digits to another office at either of two rates, one MF digit every 140 ms or one MF digit every 100 ms. These two pulsing rates are commonly called 7 and 10 pulses per second (pps), respectively. The No. 4 ESS also accepts MF pulses at either of the two rates. Also, No. 4 ESS can delay 0, 20, 80, or 220 ms between the receipt of the start-pulsing signal and the start of MF outpulsing.

*Dial Pulse (DP).* The No. 4 ESS accepts and transmits dial pulses at a nominal 10 pulses per second. No. 4 ESS will delay 280 ms between the seizure and start of DP outpulsing on DP immediate-start trunks. On non-immediate-start trunks, a 70-ms delay exists between the receipt of the start-pulsing signal and the start of DP outpulsing.

On DP immediate-start incoming trunks, the No. 4 ESS delays 90 ms between the seizure and start of digit collection to ensure that no digits have been missed. Any change of state during this 90-ms interval will be detected and cause the call to be handled as an ineffective attempt.

Table II — No. 4 ESS trunk types

| Category                                     | Trunk Types   |
|--|---|
| Incoming intertoll, incoming toll connecting | DDD access, CAMA, TSPS, secondary intertoll, intertoll  |
| Outgoing intertoll, outgoing toll connecting | Toll completing, intertoll, secondary intertoll, INWATS, rate and route operator, rate-quote operator |
| Two-way intertoll, two-way toll connecting   | MF-MF intertoll, MF-MF toll connecting, DP-DP intertoll, MF-DP intertoll, CCIS-CCIS intertoll         |
| Local tandem one-way                         | Tandem, tandem completing, intertandem completing, intertandem incoming, intertandem outgoing         |
| Local tandem two-way                         | MF-MF intertandem, MF-MF tandem connecting  |

*Common Channel Interoffice Signaling.* No. 4 ESS, as part of its original design, provides CCIS.<sup>9</sup> With CCIS all address and supervisory signals are transmitted over a CCIS signaling network to which all toll switching machines in the Bell System will eventually be connected. No. 4 ESS provides the full complement of CCIS features available to the intertoll network.

### 3.1.3 Trunk types

The No. 4 ESS handles several types of incoming and outgoing intertoll, toll connecting, and local tandem trunks. These trunk types are listed in Table II. The No. 4 ESS also interfaces with 3CL-type switchboards and is capable of operation with the Traffic Service Position System (TSPS).

*Service circuits.* No. 4 ESS has a variety of service circuits necessary to complete call-processing functions. All service circuits are treated as one-way outgoing trunks and, therefore, cannot originate service requests to the system. Service circuits provided in No. 4 ESS include: tones (e.g., reorder), announcements (e.g., vacant code announcement), MF receivers, MF transmitters, and CCIS continuity check transceivers.

*CAMA (Centralized Automatic Message Accounting).* The No. 4 ESS can be equipped with up to 72 CAMA position trunks. These trunks are used to connect incoming calls to a CAMA operator when it is necessary for the operator to obtain the calling-party directory number directly from the customer. This procedure is identified as Operator Number Identification (ONI) service.

The No. 4 ESS is compatible with those switching systems that have Automatic Number Identification (ANI) equipment. In this arrangement, the ANI equipment identifies the calling party and forwards the calling party identity via MF pulsing to the No. 4 ESS office where the charging information is accumulated. In the event of ANI failure or inconsistent

calling-party charge data, the No. 4 ESS uses ONI to identify the calling party.

### 3.1.4 Trunk characteristics

*On-hook-when-idle operation.* The No. 4 ESS software is designed to interface with on-hook-when-idle trunks. Any trunks requiring off-hook-when-idle operation for a connecting office will require a special trunk circuit at the No. 4 ESS office to convert off-hook-when-idle to on-hook-when-idle signaling.

*Ring-forward signals.* A ring forward is a timed on-hook followed by an off-hook signal on the ICT used by an operator to reestablish contact with another operator. Its duration must be greater than 60 ms but less than 200 ms to be a ring forward, with a nominal value of 100 ms, and it is allowed only on specified trunks.

There are two types of ring forward: an M-lead wink or a simplex ring forward, which is a 130-volt ac signal. Although No. 4 ESS can send either type, it can receive only the M-lead wink. Incoming trunks using the simplex version of the ring-forward signal have a special trunk circuit at the No. 4 ESS to convert it to an acceptable on-hook wink.

*Wink start, delay dial-start dial, and immediate-start operation.* No. 4 ESS will handle Wink-Start (WS) and Delay Dial-Start Dial (DDSD) signaling on MF incoming and outgoing trunks. Wink start, delay dial-start dial, and immediate-start signaling will be handled for DP incoming and outgoing trunks.

For both MF and DP incoming trunks with WS or DDSD signaling, No. 4 ESS will generate an off-hook wink with a 150-ms nominal value.

WS signals received by No. 4 ESS on MF or DP outgoing trunks must have a minimum off-hook interval of 100 ms and the on-hook must be received within 350 ms of the off-hook. In addition, the initial on- to off-hook transition must be received within 4, 5, or 10 seconds of the seizure, depending upon whether the trunk is intertoll, first-trial Toll Completing (TC), or second-trial TC.

MF DDSD signals received by No. 4 ESS must have a minimum off-hook interval of 100 ms, while DP DDSD signals must have a 60-ms minimum off-hook interval. For both cases, the delay dial signal must be received within 4 seconds of the seizure. The start dial signal must be received within 4, 5, or 10 seconds of the delay dial signal, depending upon whether the trunk is intertoll, first-trial TC, or second-trial TC.

*STOP/GO operation.* No. 4 ESS will handle STOP/GO signaling on DP outgoing trunks connected to a step-by-step office.

A STOP signal is an off-hook received on the outgoing trunk before the fourth DP digit has been outputted. This causes DP outputting to be suspended until the GO signal is received. The GO signal is an on-hook

that must be received within 4 seconds of the STOP. DP outpulsing will resume 210 ms after receipt of the GO signal.

*Echo suppressors.* No. 4 ESS will work with trunks equipped with no echo suppressor, half echo suppressor, or full echo suppressor. Software control of echo suppressors is provided via distribution points.

*Glare.* Glare exists when both connecting offices simultaneously seize the same two-way trunk for use as the outgoing trunk in a call. On MF and DP WS and DDS D trunks, glare will be detected by the failure to receive the WS or DDS D signal in the specified time. Glare cannot be detected on DP immediate-start trunks. No. 4 ESS can detect and resolve glare conditions on nonimmediate-start and CCIS two-way trunks.

When glare is detected, two courses of action are available to the No. 4 ESS offices: it can (i) release the trunk to the incoming call or (ii) continue sending its connect signal until the other office recognizes glare and returns a start dial signal. When glare is detected on a two-way operator trunk, No. 4 ESS will always take action (i) above. For all other two-way trunks, one of the two connecting offices will be designated as the control office. For this case, the control office will take action (ii) above while the noncontrol office will take action (i).

*CCIS continuity check.* CCIS does not use the voice-transmission path to pass signaling information; therefore, continuity of the voice path must be checked separately. A CCIS continuity check transceiver is connected to the outgoing trunk and a 2010-Hz tone is sent over the trunk. The connecting office loops the tone back to the transceiver where it is subsequently detected. A continuity signal is then forwarded to acknowledge a successful CCIS continuity check.

## **3.2 Call-handling system organization**

### **3.2.1 Hardware Interfaces**

The peripheral hardware (primarily the signal processor and time-division network) used in the No. 4 ESS had a significant effect on the call-processing design. In addition, since the periphery operates at a speed compatible with the processor, programs can communicate with the periphery in real time without the use of buffers. In previous ESS systems where the periphery was relatively slow compared with the processor, it was necessary for programs to time-buffer orders to the periphery. That is, a program would have to seize a buffer (possibly having to queue if all buffers were in use), load it with the necessary orders, activate the buffer, and then wait until all actions in the buffer were completed.

*Signal Processor (SP).* The SP is a No. 4 ESS peripheral unit that performs signaling and supervisory functions for up to 4096 E&M-type trunks. There can be up to 24 SPs in a No. 4 ESS office. The SP contains

6 bits of memory (called T bits) for each of its associated trunks. This memory is set by the call-handling programs and used by the SP to control its per-trunk functions. The SP scans each of its trunks once every 10 ms to determine if a change of state on its E lead has occurred. When a change of state does occur, the SP will make an entry into one of its output buffers if the trunk's T bits are set to a supervisory scanning state. The trunk's T bits can also be set to a state which will cause the SP to perform 30- to 40-ms hit-timing before reporting a state change.

The SP has four output buffers for reporting trunk status. The particular buffer used is a function of the type of report and the state of the trunk's T bits. One buffer is used for all high-priority reports (e.g., answers) and is interrogated by interject-level programs nominally every 10 ms. The other three buffers are interrogated at a slower, nonfixed rate by base-level programs. One buffer is used to hold only reports of new originations, another is used primarily for reports associated with digit functions, and the third is used primarily for reports associated with trunk abandons.

The SP also performs actions associated with digit reception and outpulsing for MF and DP trunks. Each SP can have up to 32 MF receivers connected to it. The SP periodically scans each of its MF receivers to determine if a digit has been received. The SP can accumulate from one to four digits before making an entry in its output buffer. As with MF receivers, each SP can have up to 32 MF transmitters connected to it. The SP can hold from one to four digits for each transmitter and can send the digits at either 7 or 10 pps.

DP digit receivers are not used in No. 4 ESS. Rather, DP digits are determined by detecting and counting changes of state on the trunk. The SP performs this function when such action is indicated by the trunk's T-bit state. The SP will make an output buffer entry after each DP digit is received. The SP will also detect and report abandons that occur during digit reception. The SP performs the reverse operations for DP digit outpulsing. The SP holds and transmits one digit at a time on the DP trunk. The SP also delays for the proper interdigital time before requesting another digit.

The SP can be used to report abandons on a trunk rather than simple changes of state from off-hook to on-hook. When indicated by a trunk's T-bit state, the SP will report an abandon when a change of state from off-hook to on-hook lasts more than 180 ms. In this state all other changes of state are ignored by the SP.

*Time-division network.* The No. 4 ESS switches all calls through a time-division switching network. This network switches data in Pulse Code Modulated (PCM) form with parity. Continuity of a path through the network is assured by the continued reception of good parity on the talking path. This parity bit also allows the detection of false crosses

within the network because crossed paths will cause mutilation of the parity bit. Therefore no special continuity check of the cross-office talking path by program is required. In addition, since the switched portion of the network carries PCM code and not voltage levels directly, there is no need for a cut-through relay to protect the network. The design of the time-division network allows all network operation to at electronic speeds occur compatible with the 1A Processor.

*CCIS terminal.* The CCIS signaling system requires a fully synchronized bidirectional data link passing data at 2400 bits per second. The CCIS terminal is a programmed controller which administers the routine tasks necessary to maintain synchronization of the data stream with the other office. The terminal recognizes incoming signaling messages, separates them by priority, and buffers them for the 1A Processor. Messages containing errors are intercepted by the terminal and a request for retransmission from the other office is made. The terminal also accepts messages from the 1A Processor and queues them for transmission on the data link by priority. It also maintains a history of transmitted messages so they may be retransmitted upon request from the other office. Thus the 1A Processor has a very simple software interface to the data link which takes very little real time to administer.

### **3.2.2 Data structures**

There are various data structures used by call-handling programs. These data structures include memory facilities, the call event reporting structure, link lists, queues, and the software timing structure. The two major memory facilities are the call register and trunk register.

*Call register.* A Call Register (CR) is a 64-word block of call store memory that is used for temporary storage of information during call setup. The CR is large enough to contain all information needed for processing a call. Thus, additional blocks of information need not be linked to the main register to obtain more storage space. Furthermore, any information that is derived and may be required later will be stored in the CR.

CRs are not dedicated on a per-trunk basis. Instead, there is an engineered number of CRs per office. Idle CRs are link-listed to minimize the time required both to find an idle CR and to restore a CR to idle.

*Trunk register.* Trunk Registers (TRs) are two-word blocks of call-store memory assigned on a per-trunk basis. TRs contain dynamic information about the current state of the trunk. All TRs are in a continuous block of call-store memory.

*Reporting structures.* When an internally or externally generated call stimulus (report) is detected, control is transferred to a processing routine that is identified by the type of report and the state of the call.



When a CR is not associated with a call, the state of the call is specified by the TR state code. In this case, the particular processing routine is normally determined by using the TR state code to index a table associated with a particular type of report.

When a CR is associated with a call (indicated by the TR state code), the state vector (SV) in the CR is used to specify the state of the call. The SV identifies the state of the call and is made up of eight components. Each of the eight components represents what is, in general, an independent function or part of the call. Hence, each component part of the total state vector can change independently without affecting the other state vector components. This structural arrangement simplifies the program design because it reduces the interaction of the various parts of the program and saves real time during the processing of a call.

When a report associated with one of the SV components occurs, the task program will combine the existing SV component value and the type of report and use the resultant value to determine the address in the program for processing the report. Normally, all reports can be processed independently and without checking the values of other SV component values.

*Link and engineered lists.* Two basic types of lists of software structures are provided in No. 4 ESS, two-way link lists and engineered lists. These lists are used to link together related software facilities, to provide timing functions, etc. For example, all idle call registers are placed on an idle-link list.

An entry on a two-way list has a linkage to the previous entry and another linkage to the succeeding entry on the list. Two-way lists are used in No. 4 ESS to minimize the overhead associated with entering and removing entries from the list. They can be of any length and are also easily searched for consistency, for example, by an audit routine.

Engineered lists, on the other hand, are used to provide special functions, for example, timing. These lists have a specific (engineered) number of entries.

*Queues.* A queue is a group of facilities of a given type which are all waiting for a second type of facility to become available for use. In the No. 4 ESS call-handling program, the only structures which can be on a queue for a facility are CRs and TRs. All queues consist of link-listed CRs and TRs. This provides a simple method of giving first-in, first-out service to the entries on a queue. CR and TR queues are two-way link lists.

The task program administering each queue is entered periodically to determine if there is an entry on the queue and, if so, if there is an idle facility available. If both conditions are met, then the first entry is removed from the queue and a transfer is made to the program needing the facility.

*Software timing.* Various types of timing are required during a call. For example, permanent signal-partial dial (PSPD) timing is required on all MF incoming trunks and disconnect timing is required when an on-hook report is received on the incoming trunk for a call in the talking state. In the first case, a CR is associated with the call while in the second case, only TRs are associated with the call. The No. 4 ESS call-handling programs use four types of timing: CR timing, TR timing, dedicated word timing and engineered list timing. CR and TR timing are performed by linking the particular structure (CR or TR) to a timing list. Both interject-level (accurate to within 10 ms) and base-level (accurate to within 1 base-level cycle length) timing can be performed with the CR and TR.

Dedicated word timing is another method of performing timing using the CR associated with a call. One word in the CR is reserved for timing and contains an active flag, index, and time-out time. The active flag, when set, indicates that timing is active in this CR. The index specifies which type of timing is being done, and the time-out time specifies when a time-out will occur.

When the dedicated word-timing task is scheduled, each CR in the office is interrogated for time-out if the active flag is set. This method of timing is used only where relatively long, inaccurate, and universal timing is required. An example of this is PSPD timing.

The fourth method of timing is performed with the use of an engineered list. This method of timing is used when a CR is not associated with a call and when the TR cannot be added to a timing link list. Two engineered lists are provided, one for interject- and one for base-level timing.

### **3.2.3 Call-handling program structure**

The call-handling programs are structured in a three-level hierarchy as shown in Fig. 4. The *task dispensers*, which are entered directly from executive control, interface with the signaling hardware (signal processors and CCIS terminals). The task dispensers pass call-handling stimuli to *task programs*. While performing call actions, the task programs may use one or more *call-handling subroutines* to execute repetitive or highly specialized actions. The task programs also interface with other operational programs (e.g., translations, trunk maintenance) as described in Section 3.2.4.

*Task dispensers.* The call-handling task dispensers are responsible for distributing call-related stimuli to the call-handling task programs. The stimuli can be either external (from the signaling hardware) or internal (timing or queuing reports). The task dispensers operate on both base and interject level, dispensing high- and low-priority reports re-

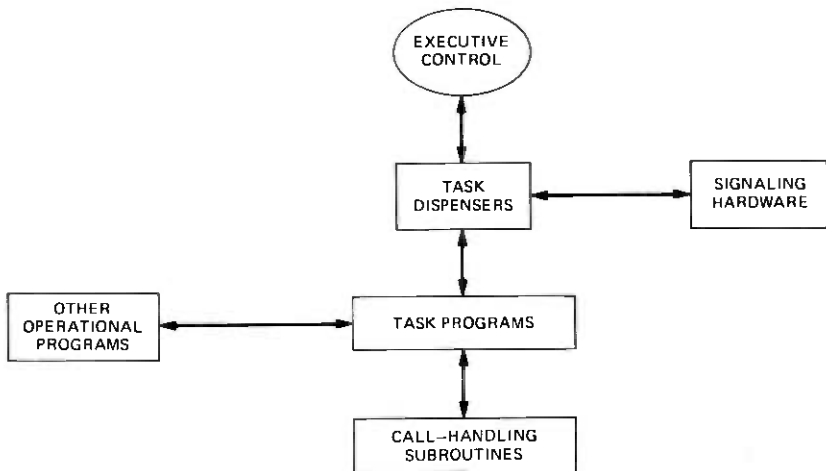


Fig. 4—Call-handling structure.

spectively. There are two basic task dispenser programs, the MF/DP task dispenser and the CCIS task dispenser.

The MF/DP task dispenser program interfaces directly with the signal processor. On each entry from executive control (interject or base), the signal processor buffers are examined for call-relevant reports (e.g., on-hooks, off-hooks, digits, stop dial). These are dispensed sequentially to the task programs for processing. The MF/DP task dispenser also dispenses internally generated time-out conditions to the task programs. Again, during each entry from executive control, the timing and queuing lists are examined for time-out conditions. If a time-out condition exists, the appropriate task program is entered to process the particular stimuli. The task dispenser remains in control until all relevant internal and external stimuli are processed or until an overload threshold is reached. The overload threshold provides a control on the amount of activity processed by the system during any base cycle.

The CCIS task dispenser interfaces with the CCIS terminals and, like the MF/DP dispenser, polls the terminal buffers for CCIS messages. If messages are present they are dispensed sequentially to the appropriate CCIS task program.

*Task programs.* Call-handling task programs, in general, are used to perform the specific actions that switch calls. In No. 4 ESS there are two types of task programs, MF/DP task programs, which handle MF and DP calls, and CCIS task programs, which handle CCIS calls. The task programs are entered from the task dispensers in response to a particular call stimulus. The task program will investigate the present state of the call by examining the TR and CR state codes. Depending on the present state of a call and the new call stimuli, the appropriate actions to advance

the call are executed. For example, if a call is in the "waiting-for answer" state and an off-hook stimulus is received by the task program on the outgoing trunk, the task program will verify the validity of the report, transmit an answer condition on the incoming trunk (e.g., off-hook), and advance the call state to "talking."

*Call-handling subroutines.* Certain repetitive or specialized call-handling functions in No. 4 ESS are designed as subroutines where they may be accessed by several task programs. Examples of subroutine actions are: the seizing and initializing of a call register, the connection of incoming trunk to outgoing trunk, the hunting of a service circuit and the pegging of a traffic counter.

### **3.2.4 Call-handling software interfaces**

As illustrated in Fig. 4, the call-handling programs interface with other operational programs during the processing of a call. These interfaces were established to allow independent software development of major operational functions such as audits, translations, and network management. Where these functions overlap during the processing of a call, clearly defined interfaces were established.

*Audits.* The call-handling programs make defensive checks to help ensure that data associated with a call has not been mutilated. When an error is found, an audit program is called. In general, when an audit program of this type is entered, the affected data structures (e.g., CR, TR) is isolated, information for a teletypewriter printout is generated, flags are set to demand additional audits to restore the structures on a deferred basis, and the affected call is terminated.

*Translations.* During the processing of a call, translation data is needed to complete the call. This data includes the trunk identification, trunk signaling characteristics, routing, and digit prefix and delete information. The call-handling programs call the appropriate translation retrieval routines to obtain the needed data.

*Trunk maintenance.* Trunk maintenance programs are called by the call-handling programs whenever a possible trunk-related hardware problem is encountered. For example, if not enough MF digits are received in the allotted time, the MF receiver used on the call is passed to trunk maintenance for testing on a deferred basis, since it may not be able to recognize digits. The incoming trunk is also passed to trunk maintenance for possible testing. The trunk maintenance programs are also entered to handle various test calls.

*Network management.* The call-handling programs interface with network management programs to modify the routing of calls based on network management actions. Network management routines also maintain a data base of call-completion statistics for later analysis.

*Overload Control.* The overload control program interfaces with the

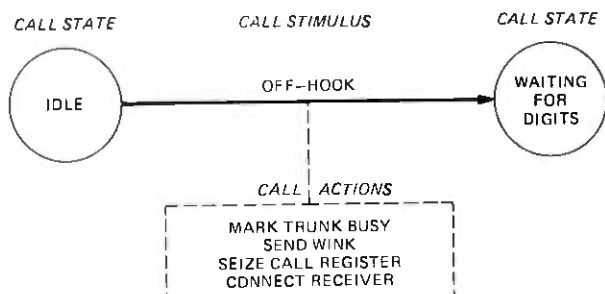


Fig. 5—Call-handling state control.

call-handling programs to control the number of originations processed in any one base-level cycle. In this way, strict control is maintained over the length of the base-level cycle.

### 3.3 Call-handling functional description

The handling of calls in No. 4 ESS follows a stimulus-response arrangement as illustrated in Fig. 5. Each call is defined in terms of a *call state* and each call stimulus (e.g., off-hook) advances the call state by performing specified call actions. In Fig. 5 the call state is advanced from idle to waiting-for-digits by the call actions of responding to the seizure, seizing a CR, and connecting the ICT to a receiver. Each stimulus is independently brought into the system by the task dispenser and passed to a task program for action. The call state is maintained in the trunk register and call register if attached.

The remainder of this section describes the No. 4 ESS call-handling programs in terms of their signaling capabilities and major functional modules. The handling of MF, DP, and CAMA calls is described in detail while the handling of CCIS is described in abbreviated form. The major call-handling subroutines of digit reception, final handling, and network actions are also described.

#### 3.3.1 MF and DP Signaling

No. 4 ESS recognizes originations on MF DDSD (Delay Dial—Start Dial), MF WS (Wink-Start), DP DDSD, and DP immediate-start incoming trunks via the signal processor. When the origination is on an MF trunk, a call register (CR) is seized and linked to the trunk register (TR) associated with the trunk and an MF receiver is connected to the incoming trunk (ICT). An MF origination will queue if there is not a CR and a receiver available. A DP origination does not have a CR associated with it until after some of the digits have been received in order to reduce the CR holding time. Overload can limit the number of MF originations served per base level and the number of DP originations served per in-

terject. An origination is put on queue if an overload threshold has been exceeded.

When an origination has been accepted, a wink must be sent to the preceding office if the trunk is either DDSD or WS. On MF DDSD trunks, the trunk circuit at the other end must receive the off-hook of the wink within a specified time or the call will be aborted. Therefore, if an origination on an MF DDSD trunk is queued for any reason, the off-hook is sent when the queuing starts.

When the "handshaking" has been completed, the digits of the called number are collected. When the ICT is DP DDSD, No. 4 ESS blinds itself for 40 ms after the start dial has been sent so that it does not interpret reflections from the start dial as dial pulses. When the ICT is DP immediate-start, the No. 4 ESS does early digit timing to ensure that the first part of a digit has not been missed because there is no delay before the digits are outpulsed to No. 4 ESS. If there is an SF (Signal Frequency) set involved, the timing is for 90 ms; if there is no SF set, only 60-ms timing (+30 ms in the signal processor) is performed. Any change of state during this timing is interpreted as an indication that part of a digit may have been lost.

For those trunks with an SF set, the SF set provides the hit-timing. When there is no SF set, the SP (Signal Processor) provides the hit timing.

The digits of the called number are used to obtain the routing information, which resides in the memory of the No. 4 ESS (see Section 3.3.4 for details of the digit reception function). An outgoing trunk is selected from the list of trunks supplied by the routing data and a path between the incoming and outgoing trunks is selected and reserved. If the outgoing trunk (OGT) is MF, the availability of a transmitter is checked, although it will not be connected until later.

A preliminary glare check on an OGT consists of performing a directed scan. If the trunk is off-hook, it is assumed to be in use by the other office and the No. 4 ESS backs off. On a one-way OGT, no glare is possible, so a glare detection is interpreted as a bad trunk. If glare is detected on a two-way operator trunk (which is really a one-way OGT), the No. 4 ESS backs off. The only true glare case can be encountered on the two-way OGT.

The appropriate handshake with the OGT is initiated. The OGTs handled by No. 4 ESS are MF DDSD, MF WS, DP DDSD, DP WS, DP immediate start, no-outpulsing, integrity check no-outpulsing [this is a WS no-outpulsing trunk for the five ACD (Automatic Call Distributor) feature], and two-way operator trunks. This last type requires that a miscellaneous point be operated to get access to the trunk.

When the handshake has been completed, the digits of the called number, after any necessary prefixing or deleting, are outpulsed. How-

ever, if the OGT is MF, the call first queues for an MF transmitter. When one has been obtained, it is connected to the OGT and outpulsing starts. If the connection fails, a new transmitter is obtained. A second failure aborts the call.

When outpulsing on a DP trunk, No. 4 ESS scans for a STOP (off-hook on the OGT) between digits. If a STOP is detected, outpulsing ceases and is not resumed until a GO signal is received.

When outpulses are sent on an MF OGT and the ICT is CCIS, the last digit to be outpulsed is held back until the COT (continuity) signals has been received. This means that there can be a delay between the tens and units digits.

Failures during MF outpulsing can be a result of a transmitter error, an off-hook on the OGT, or an on-hook on the ICT. This last case sends the call directly to final handling, and the OGT is idled as soon as the outpulsing of the current digit group has been completed. In the first two instances, the No. 4 ESS awaits the completion of the outpulsing of the current digit group before hunting a new trunk and trying again. In addition, if an off-hook occurs on a toll-completing trunk during the outpulsing, it will be considered an early answer; if this occurs on an intertoll trunk, however, it is considered to be an unexpected STOP and the call is aborted. A second transmitter error or a second unexpected STOP will abort the call.

When outpulsing has been completed, any transmitter is disconnected, the incoming trunk is connected to the outgoing trunk, and the CR is released.

If the ICT disconnects before an answer on the OGT is detected, the call is terminated and the connections are abandoned. If an answer is received, however, the call enters the talk state, in which state ring forward, clear back, reanswer, or disconnect can be received. Since No. 4 ESS employs calling-party hold, an on-hook on the OGT will be considered as a clear back. It will be passed on but will not be considered as a disconnect; only the ICT can disconnect. Once in the clear back state, an off-hook on the OGT is considered as a reanswer and is passed on.

Guard timing is the time given to the terminating office to idle a trunk. It prevents an attempt to use the trunk again before the terminating office equipment has had time to complete the release. It is always employed on an OGT and will also be used on an ICT if the M relay has been operated but digits have not yet been received. Guard timing is 1050 ms.

### **3.3.2 Common Channel Interoffice Signaling**

There are two functional components of the operational software required for CCIS. The first component, CCIS link security, maintains the integrity of CCIS data links to other offices. The second component, CCIS

call handling, handles all related signals associated with message circuits.

*CCIS link security.* The link security system administers the CCIS data links. These programs ensure that an optimum configuration is maintained for signaling facilities, that the error rate on the facilities is acceptable, and that alternate facilities are selected if a data link fails. Link security monitors and performs the synchronization protocol on the data link and provides a craft interface for facilitating repairs of the signaling link components.

*CCIS call handling.* The CCIS call-handling programs interface with signals received over the CCIS data links to switch calls between CCIS trunks and all other types of trunks. Initial CCIS incoming trunk actions handled by these programs include the analyzing of CCIS address messages, the connecting of a zero-loss loop via the time-division network to facilitate the interoffice continuity check performed on the CCIS incoming trunk, and the initiation of routing and outgoing trunk selection. Outgoing CCIS trunk actions include the control of the interoffice continuity check with a continuity check transceiver and the administration of backward failure signals. Once a call has reached the waiting-for-answer state, the CCIS programs will administer answer, ring forward, and disconnect signals.

An important factor in the CCIS program design results from the error-correcting characteristic of the signaling link. Since signals transmitted on the data link may contain errors and be retransmitted, it is possible to receive CCIS call signals out of sequence. Although this happens infrequently, the CCIS call-handling programs must process these out-of-sequence signals to successfully handle affected calls.

### **3.3.3 CAMA**

The No. 4 ESS can provide CAMA (Centralized Automatic Message Accounting) service for all customers in class 5 offices which home on it and which do not have LAMA (Local Automatic Message Accounting), for multiparty line customers served by a LAMA office (since LAMA service is limited to one- and two-party lines).

All the data for a particular CAMA call are buffered in a dedicated Accounting Block (AB) and stored in a single entry on a nine-track 800-bpi AMA tape. The data consist of the calling number, the called number, the answer and disconnect times accurate to 0.1 second, plus any other information needed to correctly bill a call. The calling number may be reported to the No. 4 ESS via either ANI or ONI. In the ANI case, equipment in the local office where the call originates outpulses the calling number to the CAMA office. If the call is ONI or if there has been an ANI failure, the No. 4 ESS attaches an operator to the call to query the customer for the calling number. The operator then keys the calling number into the No. 4 ESS.



In addition to collecting and recording toll billing data, a No. 4 ESS office can be equipped to collect and record data on calls for which CAMA records are not usually made but which do come over CAMA trunks.

A No. 4 ESS CAMA office can handle a maximum of 8160 CAMA incoming trunks. No. 4 ESS can handle a maximum of eight NPAs (Numbering Plan Areas). It can interface with a maximum of 72 CAMA operator positions consisting of a keying trunk and a talking trunk. With these two trunks, the operator is able to key in the calling number over the keying trunk while receiving it from the calling subscriber over the talking trunk.

Signaling between No. 4 ESS and the CAMA positions can be on either a loop or an E&M signaling basis. If E&M signaling is used, a special trunk circuit allows the CAMA position to be a TSPS operator position. The CAMA operator may be at a regular CAMA cordless position, a cord switchboard modified for CAMA operation, or a TSPS No. 1 100B position. These positions can be in the same building with the No. 4 ESS, or they can be located remotely. To the toll office, however, they always appear to be at a remote location.

Incoming CAMA calls are processed by No. 4 ESS on a first-come, first-served basis. When a CAMA call requires ONI, the most idle CAMA position trunk is chosen to permit an even distribution of calls among CAMA position trunks. If no CAMA position trunk is available for a CAMA ONI call, the call is queued and the customer receives audible ring until a position becomes available.

A hardware system clock is used to accurately maintain the software time-of-day clock.

### **3.3.4 Digit reception**

The digit reception module is responsible for performing all actions unique to the collection and analysis of digits received on an incoming call, including the calling number for a CAMA call. This module centralizes all call-handling actions relative to the accessing of translation and routing data, including interfacing with code restricting network management routines. The actions performed by this module are described below in terms of the digit and routing capability of No. 4 ESS.

No. 4 ESS will accept 3 through 11 digits (excluding the KP and ST digits) domestically via MF, DP, or CCIS signaling. It can translate on 3 through 9 digits to obtain the routing data. When outpulsing, it is possible to delete any number of digits and/or prefix up to 6 digits. The digit translation tables and the routing data blocks can be in core and/or disk storage.

When the incoming signaling is via MF, all digits are collected before a digit translation is requested, since the ST digit acts as a positive indication of end of dialing. When digits arrive via CCIS, they are all con-

tained in one message, so when the digit translation is requested, all the digits have been received. When the incoming signaling is DP, however, there is no positive end-of-dialing indication. Five-second critical interdigital timing is performed after any digit which may have been the last, and 15-second noncritical interdigital timing is performed after any digit which is not expected to be the last. To avoid delaying each DP call by at least 5 seconds in determining end of dialing, the digit translation is overlapped with the DP digit collection. Normally, the routing information will be obtained before all digits have been received. Part of this routing information specifies the expected number of digits and, at least in the case of receiving the maximum number of expected digits, this information can be used to avoid the necessity of timing after the last digit.

Normal calls, maintenance calls, and internally generated test calls all use the same digit reception software to collect, analyze, and translate digits, although some calls, such as test calls, do not require the translation.

Any failure encountered in the digit reception process, such as an illegal digit (i.e., two consecutive KP digits) or a digit error (an MF code which is not 2-out-of-6), result in the call being sent to final handling for termination.

### **3.3.5 Final handling**

The final handling module is called to idle facilities and update counters associated with calls that are not completed by the No. 4 ESS; i.e., Ineffective Attempts (IAs). An IA is any attempt recognized by the switching machine as a bid for service but which does not subsequently result in the call being completed in the desired manner. This category consists of the calls which do not reach the waiting-for-answer state for any reason. Once a call reaches the waiting-for-answer state it will be idled by normal call-handling actions.

When an IA is detected, there are facilities associated with that call which must be restored to an idle state so that other calls can use them. A "facility" is defined as a hardware facility, or a software structure: e.g., service circuit, trunk, call register, or timing-list entry. Final handling also connects the incoming trunk to a recorded announcement or tone as appropriate. It will time the announcement connection and will disconnect it if the incoming trunk does not disconnect within a reasonable amount of time. No. 4 ESS provides all necessary tones and announcements for toll service.

### **3.3.6 Network actions**

The network-actions subroutine is the interface between the call-handling programs and the No. 4 ESS time-division network hardware,

specifically the time-slot interchange (TSI) and the time-multiplexed switch (TMS). To provide this interface function, the network-action routines must hunt network paths, perform the path setup, maintain the residual path information necessary for releasing the path on demand, and must also administer the use of the network links to prevent crosses from occurring. Since the network-action routines are the only ones designed to access the time-division network, these routines must set up all the types of paths required by the call-handling programs. The types of routines provided include a normal 2-way path connect, a path reservation (used while one or both trunks are involved in another connection), monitor connections and a broadcast feature.

*Path hunt strategy.* The path hunt strategy chosen to select idle network paths must satisfy two criteria:

(i) It must not degrade the inherent blocking characteristics of the time-division network.

(ii) It must be real-time efficient; i.e., the run time of the hunt must be kept to a minimum.

In order to satisfy the first criterion, extensive simulations were made of the final strategy, proving that the network met its blocking objective with a substantial margin. To satisfy the second requirement, great care was exercised in the design of the data structures and the resultant program necessary to execute the network path-hunt algorithms. Once a path is selected between two endpoints, the network program causes that path to be set up in the network. In order for this function to be performed efficiently by the program, the network hardware was designed to accept data to set up paths in a form compatible with the internal data formats used by the network programs.

*Network link administration.* A map of the busy/idle status of each link on each time slot is maintained in the software. The purpose of this map is to ensure that only idle facilities are used in setting up new network paths. The map is composed of three parts. The first part, the time-slot map, keeps track of each of the 128 time slots to which a trunk has access. The second part, the A-link map, keeps status on the links joining the TSI to the TMS. Finally, the B-link map has the status of the links joining the two stages of switching within the TMS.

The two criteria which governed the design of these status maps were speed of access for the path hunt program and modularity to allow graceful growth of the map as the office grew.

The network map structure enables the network programs to keep track of link usage. It is also necessary to maintain information concerning active paths in the network, so they may be disconnected properly. Since this is trunk-related information, it is stored in the trunk register of one of the connected trunks. The trunk register also has a

pointer word which identifies the other trunk which is connected via the network path. The two trunk identities and the path information contain enough information to identify all links used by the network path being held.

*Monitor connections.* The ability to monitor transmission for any trunk within No. 4 ESS is a requirement for trunk maintenance. It is possible to use the one-way transmission capability of the time-division network to connect a trunk circuit, which is already busy, to a special trunk maintenance bridging circuit. This allows measurement of transmission factors while the trunk is in use.

*Broadcast of announcements.* Connections to customers needing announcement or tone treatment are made by a special connect routine which allows the same announcement to be broadcast to many customers simultaneously. This is provided by dedicating a special TSI to this function, making a semipermanent network connection from the announcement source to the special TSI, and using the unique properties of the time-division network to make the announcement or tone available to as many as 1024 network ports.

#### IV. SUMMARY

This paper has described the operational software structure of No. 4 ESS with particular emphasis on the design of the call-handling software. The development of this software package was a large undertaking utilizing the skills of over 100 software designers. While some of the software philosophy used in No. 4 ESS was built on previous ESS systems, much of the design was totally new, particularly in the areas of handling and switching toll traffic through a digital switch and the accompanying administrative features. We acknowledge the effort of those designers, too numerous to mention, who contributed to the successful No. 4 ESS development.

#### REFERENCES

1. J. M. Scanlon and L. S. Tuomenoksa, "No. 4 ESS Software," Proc. Intl. Switching Symp., October 1976.
2. M. J. McPheters, S. Bloom, S. H. Tsiang, "Software Quality Control," Proc. 1973 Software Reliability Symp., April 1973.
3. J. S. Nowak, "No. 1A ESS—A New High Capacity Switching System," Intl. Switching Symp., October, 1976.
4. B.S.T.J. special issue on 1A Processor, 56, No. 2 (February 1977).
5. J. A. Harr, E. S. Hoover, and R. B. Smith, "Organization of the No. 1 ESS Stored Program," B.S.T.J., 43, No. 5 (September 1964), pp. 1923-1959.
6. M. N. Meyers, W. A. Routt, and K. W. Yoder, "Maintenance Software," B.S.T.J., this issue, pp. 1139-1168.
7. J. A. Giunta, S. F. Heath, J. T. Raleigh, and M. T. Smith, "Data and Trunk Administration and Maintenance," B.S.T.J., this issue, pp. 1203-1238.
8. T. V. Greene, D. G. Haenschke, and B. H. Hornbach, "Network Management and Traffic Administration," B.S.T.J., this issue, pp. 1169-1202.
9. B.S.T.J., special issue on CCIS (to be published).

## **No. 4 ESS:**

# **Maintenance Software**

By M. N. MEYERS, W. A. ROUTT, and K. W. YODER

(Manuscript received July 21, 1976)

*To ensure quality service, a toll switching system must be able to meet very stringent dependability and maintainability requirements. To meet these requirements a large package of maintenance software has been developed. This software consists of four functional areas. The first area deals with the detection and recovery from software malfunctions. These malfunctions include failing defensive program checks, scheduling irregularities, and mutilated data. The second area is concerned with the recovery from hardware faults. Hardware fault recovery is stimulated by a failing hardware check and is completed when the system has been reconfigured around the faulty unit. The third area is concerned with the diagnostic programs that aid the craftsperson in the identification and repair of the faulty unit. The fourth area provides for overall coordinated system recovery from multiple or very severe hardware and software malfunctions.*

## **I. INTRODUCTION**

To ensure quality service, a toll switching system must be able to meet very stringent dependability and maintainability requirements. Dependability requirements are defined in terms of service continuity and accuracy. Maintainability requirements provide a measure of how quickly hardware or software malfunctions must be corrected. These requirements of dependability and maintainability considerably influence the design of the hardware and software subsystems composing No. 4 ESS.

Continuity of service is provided by both hardware and software redundancy. Hardware redundancy takes the form of providing mechanisms to switch to a spare unit whenever a hardware fault is detected.

The spare unit then performs the function of the faulty unit during the repair process. Redundant software is provided by mechanisms that regenerate program and data structures found to be in error.

Accuracy of service is guaranteed by extensive checking mechanisms in both hardware and software. Typical hardware checks include parity and matching circuits. Software checks include the auditing of data structures and program sanity. The failure of a hardware or software check provides the stimulus for switching to the spare unit or for regenerating a data structure.

Hardware and software maintainability is provided through tools that aid in the rapid repair of system faults. Hardware repair aids include extensive on-line diagnostic tests that isolate a hardware fault to a small number of replaceable circuit packs. Software repair aids include output messages that contain the necessary data to aid in the isolation of the software fault to a particular program or data structure.

The above basic plan is based on prior experience with electronic switching systems.<sup>1,2</sup> However, new concepts and significant extensions of the prior art have been incorporated throughout the design.

### **1.1 Maintenance software**

To meet the stated dependability and maintainability requirements, a large package of maintenance software has been generated. This software has been functionally divided into four areas. The first area is concerned with the detection and recovery from software malfunctions. These malfunctions include failing program checks and illegal data structures. The second area provides for the recovery from hardware faults. Hardware fault recovery is stimulated by a failing hardware check and is completed when the system has been reconfigured around the faulty unit. The third area is concerned with the diagnostic programs that aid the craftsperson in the identification and repair of the faulty module. The fourth area provides for overall coordinated system recovery from multiple hardware and software malfunctions.

## **II. SOFTWARE ERROR RECOVERY**

No. 4 ESS depends upon the data contained in its memories to control the actions of the system. Also, in an operational mode, No. 4 ESS can write into any of its memories and consequently the system is subject to memory mutilation. Therefore, it is necessary to make the system as error-tolerant as possible and also as error-free as the architecture will permit. In order to be error-tolerant, the system must operate in the presence of memory mutilation. In order to be as error-free as possible, there must be restrictions placed upon the software system which can be learned from the analysis of previous systems' error characteristics.

Once the error characteristics are defined, one can strive toward error prevention. Knowing the system can never prevent all errors from occurring, one attempts to achieve a system that is tolerant of as many errors as possible. Then, for the types of errors the system is not tolerant of, error-detection schemes must provide rapid detection. Once errors are detected they should be handled efficiently to minimize real-time usage and to assure the integrity of No. 4 ESS.

## **2.1 Software-error characteristics**

In order to achieve error tolerance and error prevention in No. 4 ESS, software errors must first be defined and characterized. An error can be defined as any data that cause the system to operate abnormally.

### **2.1.1 Causes of errors**

Errors can be introduced into the system in many different ways. In No. 4 ESS, it is the intent of the software-error recovery strategy to eliminate or reduce the occurrence of as many causes of software errors as possible. There are two causes of errors, however—hardware faults and craftsperson errors (other than those at the man-machine interface)—that are not considered within the scope of this section on software-error recovery.

Often programmers have to be aware of and understand many complex and nonstandard program interfaces. The lack of this understanding often results in programs that cause errors when communicating with other programs.

A programmer who is not fully aware of or does not understand the system rules can produce programs that violate one or more of these rules. This can result in data mutilation. For example, a programmer who does not know that the system's shadow registers are destroyed during certain subroutine calls can write a program that leaves pertinent data in the shadow registers over one of these subroutine calls. The information held by these shadow registers will therefore be destroyed upon the subroutine's return.

Logic or coding errors can cause data mutilation. It is possible for these types of errors to go undetected by the program debugging processes, especially if the errors reside in an infrequently entered path of the program. The following two examples draw a distinction between a logic error and a coding error. A logic error would be using  $2 * (\text{base} + \text{index})$  to derive an address when  $\text{base} + 2 * \text{index}$  should have been used. A coding error occurs when the programmer codes an SWK instruction when CWK was intended.

Man-machine interface procedures that are complex and nonstandard often cause errors. For example, if a problem exists that requires per-

sonnel on duty to follow a poorly defined or overly complex procedure at the Master Control Console (MCC), confusion will often result. This will frequently lead to erroneous action at the MCC, thus compounding the problem.

### **2.1.2 System effects from errors**

System effects from errors can appear in several different ways.

The most severe effect of an error is the loss of processing viability, where the processor does not have the ability to perform any software functions—that is, an error which causes a loss of program sequencing such that the system is driven into a system initialization phase. (This kind of error is discussed further in Section V.)

Even though loss of processing viability results in the loss of call-handling capability, it is possible to retain viability (do other work and cycle) and yet have no calls completed through the system.

A facility can be considered as any equipment or software item required for proper system operation. Specific facilities are required to perform a given function. An error that causes denial of a facility will affect the system by restricting the associated function. The error effects that are characteristic of this category do not include a denial of the total facilities that constitute a function.

Loss of a function is closely related to the previous one, denial of facilities. Facilities are required to perform a function. Therefore, an error which causes the total loss of one type of facility implies the loss of the associated function. The loss of a function can also be caused by a scheduling error that does not allow the function to ever be entered.

The capacity of an office can be significantly reduced as the result of system errors. For example, if a link word is "garbaged" part way down a link list of idle call registers, which are required on all calls, then the office has a reduced number of call registers to work with. Therefore, the load-handling capacity of the office has been reduced.

Loss of a single call is the result of mutilation of information pertinent to the setup of a single call. For instance, destroying digits in a call register will cause the aborting or mishandling of the associated call. If, however, similar data associated with many calls is consistently mutilated, then the system effect will clearly be more severe.

It is also possible for errors to have no effect on the system. One example is mutilation of a word in an unassigned trunk register.

### **2.2 Impact of software-error prevention and tolerance**

To ensure the integrity of the memory in the No. 4 ESS, the first step is to prevent the occurrence of as many errors as possible. But errors will still occur. Therefore, to further ensure the integrity of the memory, the system should be as error-tolerant as possible.



### **2.2.1 Error prevention**

After the causes of errors were considered, three general methods of error prevention became apparent. These were standardization, simplification, and improved documentation.

These general methods of error prevention were applied to potential causes of errors in No. 4 ESS. Increased standardization was directed toward program interfaces and man-machine interface procedures. Simplification was applied to program interfaces, man-machine interface procedures, and main call flow under overload. And last, documentation improvement was applied to the areas of system rules and man-machine interface procedures.

### **2.2.2 Error tolerance**

After error prevention techniques were applied to No. 4 ESS, attempts were made to improve the error tolerance of No. 4 ESS, since errors will still occur. Error tolerance implies that the system is able to operate in the presence of memory mutilation.

In an attempt to achieve a high degree of error tolerance, two major mechanisms are used in No. 4 ESS. These are defensive coding and defenses for memory.

Defensive coding mechanisms are used when writing programs in an attempt to remain operational in the presence of errors. When this is not possible, the goal is to operate so that an error will have a minimal effect on the system. In order to operate properly in the presence of errors, defensive coding in programs attempts to detect any error in the data that the program is using. In order to operate with a minimal effect on the system, while in the presence of undetected errors, defensive coding in programs attempts to restrict program accessing of data. It attempts to restrict access to noncritical areas where memory mutilation will have a minor effect on the system.

Specific types of defensive coding that were used towards both of the above-mentioned objectives are:

- (i) Checking state codes
- (ii) Range checks
- (iii) Positive decisions (no decisions by default)
- (iv) Symbolic addressing
- (v) Interpreting all possible stimuli
- (vi) Linking by index (rather than link by absolute address)

In No. 4 ESS there is certain information aside from the actual programs stored in writable memory that is critical to the proper operation of the system. Critical memory can be considered as any memory in which an error, if it occurs, could have a drastic effect on the operation

of the system. An example of critical memory is the basic office parameters (e.g., starting addresses of software items, numbers of active hardware unit types, etc.). If an error occurred in these parameters the operation of the system would be severely affected. Therefore, it is essential to protect this critical memory.

The one main defense used for critical memory is a physically protected area. A special instruction is required to write into the protected area. Therefore, the physically protected area of memory is guarded against wild writes. Also, physical protection becomes an even more powerful defense when the frequency of writing in the protected area is kept very low.

No. 4 ESS, having writable program stores, requires some form of protection for them also. Therefore, no operational programs are allowed (as a standard procedure) to write into program store except for the paging routine and specific recovery routines. In addition, the program stores are all physically protected as described above.

It should be noted that duplication of memory is not considered a defense since it protects only against hardware faults and not against memory mutilation resulting from erroneous software operations.

Another defense for memory is to provide a software protection scheme when operation programs are writing disk. The disk system contains backup copies for critical memory and program stores in addition to storing noncritical disk-only data. Therefore, some protection is required when operational programs are writing into noncritical disk areas. The software protection scheme checks identification tags on write requests against a list of valid writable areas for that given identification tag.

The third form of defense for memory is defensive memory layouts in the unprotected area. Even though the most critical office information is stored in the protected area of memory, defensive memory layouts still need to be employed in the unprotected area of memory. There is still important office information stored in this area which, if mutilated, could have a serious effect on the operation of the system. Since there is a high frequency of writing in the unprotected area, there is a high probability of memory mutilation occurring there. The two major methods used in unprotected memory are to disallow any common scratch areas, and to prevent overlapping of private scratch areas by requiring all private scratch to be allocated by COMPOOL (common pool of data) and defined on COMPOOL.

### **2.3 Software structure**

Once all error-prevention and error-tolerance measures were taken, a software structure was designed to detect, analyze, and correct the

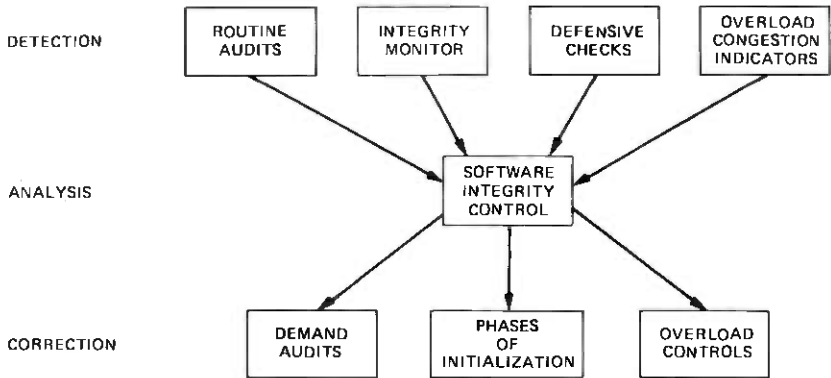


Fig. 1—Software integrity control function.

software errors that will still occur. The software structure that exists for software-error recovery has three major components. These are:

- (i) The software integrity control program
- (ii) The audit system
- (iii) The software-integrity monitor system

### 2.3.1 Software integrity control

The Software Integrity Control (SICO) program serves as the centralized control for the integrity function. It has all software errors reported to it, makes decisions about the appropriate actions needed, and then activates the appropriate corrective action (see Fig. 1).

The detection of software errors is done via the audit system, the integrity monitor system, and defensive checks implanted throughout the entire No. 4 ESS software. The audit system detects primarily data errors, the integrity monitor detects primarily scheduling and cycling irregularities, and the defensive checks detect primarily mutilated data. All of these errors, when detected, are reported to the SICO program for analysis and corrective action. After analysis, SICO can decide whether to request an audit to correct the error, and if so, which audit. SICO can also make a decision, based on an audit history, to escalate the request to a more severe corrective action such as a phase of software initialization (Section V).

In addition, SICO also receives reports on internal machine congestion from the overload program. This permits SICO to check via other error reports whether this was a congestion falsely indicated by software errors. If so, SICO will request an audit to correct the error, otherwise SICO will allow the overload program to activate the appropriate overload controls.

### **2.3.2 Audit system**

The audit system detects and corrects software data errors. It is basically composed of a control structure and several audit routines, each one tailored to a specific data structure or a group of similar data structures.

The audit control structure schedules the routine audits in addition to running demand audits. The routine audits are run according to the direct search method of error detection. The frequency for running each routine audit is dependent upon the system's sensitivity to errors in that particular data structure or group of data structures. Therefore, the more critical audits are run at a higher frequency than the less critical audits. Also, the audit control structure interleaves the running of routine audits so that the longer duration audits do not lock out the shorter-duration and usually more critical audits. The demand audits, requested either manually or automatically (via the SICO program), are run on a higher priority than routine audits and are not interleaved.

The audit routines detect errors using three basic techniques. These are:

- (i) Direct comparison (e.g., comparing data with a duplicate copy in core or on disk).
- (ii) Comparison by association (e.g., verifying that the proper registers are linked together).
- (iii) Format comparison (e.g., verifying that the data in a particular register appears to be reasonably correct).

The individual audits are written for a specific data structure or a group of similar data structures. The general types of data structures audited are common usage registers, timing structures, queues, and general lists.

### **2.3.3 Integrity monitor system**

The integrity monitor system is primarily concerned with detecting scheduling and cycling irregularities as well as losses of major system functions. It is composed of the general time monitors, the software integrity monitors, and the test call program.

The first of the time monitors is the Program Sanity Timer (PST). The PST is a hardware timer in the central control with a time-out interval of 640 milliseconds (ms). Within this interval, an enable signal must be sent after 320 ms have elapsed. A reset signal must be sent after the enable and before time-out. The PST is administered by the system integrity monitor program on interject. The fact that the system does not time out verifies that the system software has a certain amount of competence or sanity. If the PST times out, a B-level interrupt is generated and a phase of software initialization is run.

The second time monitor is the K-level interrupt. A K-level interrupt occurs whenever the 10-ms clock attempts to set the software interject request flag and the flag is already set. This situation occurs when interject has not been served for 10 to 20 ms. When a K-level interrupt occurs, a failure count is incremented and compared with a threshold. If it exceeds the threshold, a phase of initialization is run via the SICO program. If the threshold is not exceeded, the interrupt returns to normal processing.

The Software Integrity Monitor on Base level (SIMB) performs detailed checks to verify the validity of the base cycle. This includes checking base-level program entry counts, comparing the base-level cycle length just completed with the previous cycle, and comparing the last base-level cycle length with a minimum allowed value. SIMB also routinely performs functional checks to verify that major system functions such as the disk system and the input/output system are available.

The Software Integrity Monitor (SIM) on interject checks the validity of the scheduling on interject and ensures entry to other integrity routines lower in the scheduling structure. SIM also, as previously mentioned, administers the PST. Both SIMB and SIM use failure counters and appropriate threshold values when deciding whether to report a failure to the SICO program.

The test call program provides a gross check upon the system's operation by preventing special calls to the system and observing the progress of these calls. If a call should fail to progress as anticipated, then checks are made which attempt to isolate the cause of the trouble. The test call program consists of four sections: a generator, a call monitor, a progress monitor, and failure processing. The test call program presents calls of all three pulsing types: MF, DP, and CCIS.

### **III. HARDWARE ERROR RECOVERY**

The nature of No. 4 ESS peripheral hardware and the software structures used to control it are the two major points related to hardware-error recovery. The hardware is highly autonomous. Various means for providing redundancy and error detection are used. The separate hardware units are tied into an overall interrupt structure.

The software controls the interconnection of communication and control paths between the peripheral units and the central processor, and between peripheral units themselves.

#### **3.1 Hardware architecture**

The periphery of No. 4 ESS can be broken down into three areas: switching, network, and transmission/switching interface, Fig. 2. The last two are often inseparable when error detection and recovery are

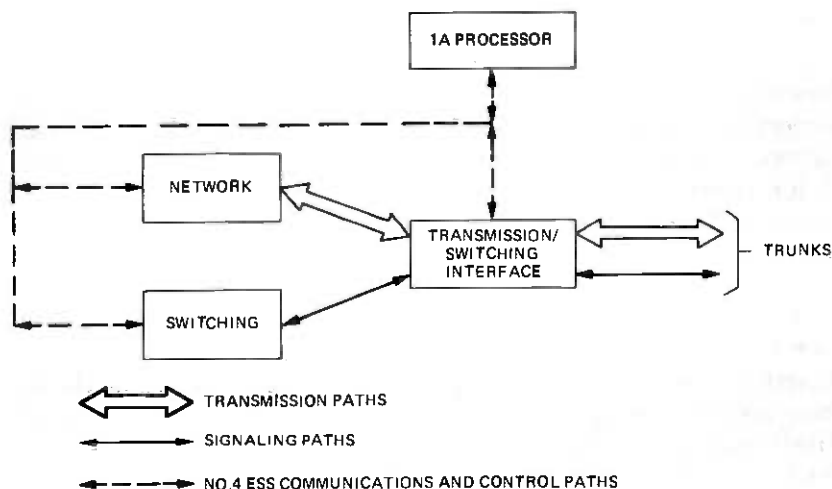


Fig. 2—No. 4 ESS architecture.

considered. However, each of the three areas has a high degree of autonomy and its redundancy and error-detection scheme is unique.

### 3.1.1 Autonomous nature of hardware

The signaling units (signal processors and CCIS terminal) are autonomous processors. The SP is a wired-logic machine that scans for supervisory changes, and collects/transmits dial pulse and MF signaling information. The CCIS terminal is a programmable processor that performs analogous tasks in its environment.

Each has an independent clock and except for inquiries from the central processor, they are independent of the central processor.

Network and transmission/switching interface frames are linked together by a common network clock and by common transmission paths. These units work together to autonomously set up paths through the system and to convert between various transmission formats (i.e., analog to digital, digital to digital, digital to analog).

The central processor's only contact with these units in the operational environment is to provide path setup information. The time-shared paths are set up and removed independent of further intervention by the central processor.

### 3.1.2 Redundancy

The redundancy imposed on a peripheral unit is related to the number of trunks affected by a failure, the probability of a fault in the unit, and the practicality of a particular redundancy plan.

Table I — System outage for a total unit failure

| Failed unit                    | Number of trunks or percent of capacity lost | Unit redundancy scheme   |
|--------------------------------|--|--|
| Peripheral unit bus            | All trunks, 100 percent                      | Full duplication plus dispersion   |
| Network clock                  | All trunks, 100 percent                      | Two fully duplicated halves  |
| Time-multiplex switch          | 25 percent to 50 percent                     | Full duplication   |
| Time-slot interchange          | 1680 trunks                                  | Full duplication   |
| Signal processor               | 4096 trunks                                  | Full duplication   |
| CCIS terminal access circuit   | 16,000 trunks                                | Full duplication   |
| CCIS terminal                  | 10,000 trunks                                | Load sharing between two terminals   |
| Voiceband interface controller | 840 trunks                                   | Full duplication   |
| Voiceband interface unit       | 120 trunks                                   | Protection-switched space for 7 units  |
| Digroup terminal controller    | 960 trunks                                   | Duplication (some maintenance features are not fully duplicated)             |
| Digroup terminal unit          | 120 trunks                                   | Protection-switched space for 8 units  |
| System clock                   | 0 trunk                                      | Full duplication (a software timer is available to back up the system clock) |

The need for more or less hardware to perform a chosen function, the economies of scale, and packaging constraints can have more influence on redundancy than the number of trunks affected by an error in the unit. However, in the case of No. 4 ESS, the number of trunks affected by a failure played the major role in determining a redundancy plan.

The loss of the network clock means the outage of the entire No. 4 ESS as a switching machine. The network clock forms the foundation for the entire network and transmission/switching interface. It is a dual-duplex arrangement. There are two pairs of clock chains. Either chain in a pair can fully replace its mate, but the members of one pair cannot take the place of the members of the other pair. A pair provides clocking to one half of all duplicated units in the network and the transmission/switching interface equipment. The other pair provides clock to the other halves.

A total hardware failure of a unit other than the peripheral bus or network clock affects less than 100 percent of the No. 4 ESS capabilities either in capacity or trunk access. The number of trunks that can be denied access or the reduction in capacity that can occur when a unit fails is summarized in Table I.

### 3.1.3 Error detection

Error detectors provide a means of identifying when errors occur and, if possible, pinpointing the cause of the error. In the No. 4 ESS pe-

riphery many types of error detection are used: parity checks, code checks,  $1/n$  enable checks, matching, etc.

These error checks can be classified as unique and nonunique. Unique error detectors indicate the presence of an error and locate the error to a reconfigurable block of the system. A reconfigurable block is half of a duplicated unit, one of  $n$  units with a protection-switched backup, etc., that can be placed in or out of service. Sometimes many of these are in series. An example might be the central processor (CP), peripheral unit bus (PUB), and a peripheral unit.

Figure 3 has six configurable blocks or three pairs of interchangeable blocks. A unique error detector would be one that uniquely identified one of these blocks as faulty. If Unit 0 had internal memory with a built-in parity check P, a parity failure would be unique.

Nonunique error detectors identify error conditions but give little information as to which configurable block is at fault. An example is a matcher "m" between an output register in each half of the unit. If a mismatch occurs, no clue exists as to which half of the unit is at fault.

Furthermore, if data input to the unit has any effect on outputs and was not error-checked on the buses, the CPs could be at fault and the error would not be detected until the mismatch occurs.

No. 4 ESS employs both types of error detectors. Most communication paths and links in the periphery employ unique error detectors or those approaching uniqueness.

Unique error detectors proved to be too expensive in most areas where logical and arithmetic operations are performed. In these cases matching was employed between duplicate halves for error detection and resulted in nonunique error detection. When an error is detected in the periphery, the main program in the central processor is notified via an interrupt structure.

### **3.1.4 Interrupt structure**

Errors detected in the periphery are reported to the central processor via hierarchical interrupt levels. There are three levels based on the urgency of correcting the effects of the error.

The F-level interrupt is the highest level for the peripheral system and causes the central processor to give immediate attention to the error condition.<sup>3</sup> It breaks into the basic task being executed and the task may be aborted. An F-level interrupt has two subclassifications, a Peripheral Unit Failure (PUF) and an Autonomous Peripheral Unit Failure (APUF). PUF can be triggered only by error-check hardware when the central processor is actively addressing a peripheral unit via the peripheral bus. APUF can be triggered by error-check hardware when the central processor is not addressing the periphery or when the error is independent of central processor access.



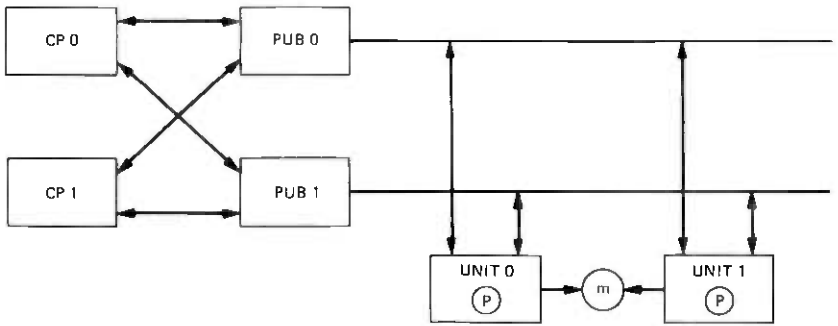


Fig. 3—Example of reconfigurable blocks.

An Autonomous Peripheral Unit Trouble (APUT) is the second level the central processor will recognize. It will be recognized within 3 ms of error detection or as soon as the main program completes its present basic task and arrives at a safe point. Interwrite problems might occur if it were immediately recognized by breaking into the execution of the current task. Where time allows, the autonomous peripheral unit trouble interrupt is used instead of the F level so the current task can be successfully completed.

The Autonomous Peripheral Unit Base-level APUB interrupt is the third and final level. Treatment of an error is deferred even longer, up to 100 ms. The treatment becomes a base-level task and the actual interrupt does not affect the performance of the main program.

Each hardware-error detector causes one of these interrupt levels to be entered, which in turn leads to a software structure that will locate the source of error and isolate it from the active system.

### 3.2 Software architecture

The software structure which deals with errors consists of functions that are unique to a particular peripheral unit type (concentrated in per-unit type software packages) and functions which control a large subset of the peripheral unit types.

#### 3.2.1 Concentrated Unit Structures

Although most of the peripheral units are connected to the central processor over a common peripheral bus, internally they differ markedly. Even the bus interfaces, although functionally equal for operational access from the central processor, are different when examined in detail. Thus the routines needed to deal with a particular unit type are concentrated in one software package. This allows for better maintainability of these functions because the individual programmer need only

be aware of the detailed workings of one or two units. The types of routines concentrated on a unit basis fall into three broad categories, unit fault recovery, unit bootstrap, and unit configuration.

Unit fault recovery isolates a fault to a configurable portion of a unit type once the source has been traced to a unit. It assumes the central processor and peripheral bus have been exonerated by preceding recovery action. The unit fault recovery filters out the most likely source of error according to a priority structure based on the architecture of the unit. In the case of a nonunique error it will run tests to determine the configurable portion of the unit that is at fault. Once the portion at fault is identified, fault recovery selects a course of action. It then confers with a centralized error analysis to have the decision accepted or changed with regard to previous errors from the same unit or units interacting with it. Upon return from error analysis, unit fault recovery carries out the action agreed upon by setting up intraunit functions itself and by going to a centralized peripheral configuration program to set up interunit functions. Once the action is complete, a report of the error and its resolution is made and all collected data is archived via the 1A common error analysis program.<sup>3</sup> Control is then passed to a system restart program.

Unit bootstrap routines perform initialization of a unit. They assume the unit can be in any state and will bring it to a state suitable to begin call processing. An access test is performed on the unit to ensure that it has basic sanity and that the risk of introducing a formerly out-of-service unit into the overall system is minimal. All of these types of routines—bootstrap and access test—are tied together by a centralized peripheral hardware recovery.

Unit configuration routines provide both inter- and intra-unit routing of communication and control paths. These routines are linked into a common peripheral configuration package by a centralized configuration control program.

### **3.2.2 Centralized Control Structures**

The centralized control structures in peripheral maintenance can be divided into five categories:

- Hardware recovery
- Peripheral configuration
- Peripheral error filtering
- Error analysis
- System restart

Hardware recovery is called by system recovery (Section V) to select and certify a working combination of peripheral equipment. Each of several levels of hardware recovery is progressively more severe. The first

level tries not to disturb any hardware in service at the time it is entered. Successive levels simplex the equipment and interchange redundant portions of equipment previously left out of service.

At the most severe level, a minimal set of peripheral equipment is reinitialized and configured. The object is to eliminate more and more possible sources of system upheaval that may have lead to system recovery action. To perform its task, hardware recovery calls on unit bootstrap and access test routines as well as unit configuration routines via the peripheral configuration.

The peripheral configuration program acts as a clearing house for all interunit configuration changes of communication and control paths. It calls upon unit routines to perform specific tasks, and it stitches these tasks together to assure access to and from the various areas of the system is not lost to call processing. An example is the removal of a peripheral bus which interconnects the periphery with the processor. Before the bus can be removed, all other units interfacing with the bus must be examined and possibly reconfigured to ensure all in-service units have functional interunit address and control paths with the remainder of the system.

System peripheral-error filtering resides in a program that is entered for each type of error detected from the periphery or at the processor-to-periphery interface. It determines the peripheral unit implicated by the error indicator and then isolates the cause of error to the processor, the peripheral bus, the implicated unit's bus interface, or the implicated unit. It thus must deal with all units in at least a superficial manner. If it determines the error is within the unit, it will transfer control to a unit fault recovery routine for further error resolution.

Error analysis adds the element of past history to fault recovery. It acts to resolve interframe errors that are not associated with the peripheral bus. It maintains a record of all errors and their resolution for a period of time. A decision made by a fault recovery routine is passed to error analysis for examination before it is acted upon. Error analysis can concur or alter a decision, according to past history as examined via a sequence table. Sequence tables are a collection of decision schemes which make different decisions on successive occurrences of an error. A sequence table is selected by fault recovery for each type of error. If there is not past history active in the error analysis data base associated with the error and unit under investigation, the first decision scheme of the sequence table is used to carry out the analysis. If there is past history present, the next decision scheme of the sequence table recorded in the last record of past history is used.

These decision schemes can draw on several factors such as:

The environment of the configurable portion of the system in error (i.e., duplex, simplex etc.).

The number of times the error has occurred over an interval of time.

The type of error (unique, nonunique).

The characteristic of the error (transient, hard failure, illegal system action).

System restart provides a point where all error treatment is terminated and the system is gracefully returned to call processing. Cleanup of the system software resources disturbed by the error is done at this point. Records of the error and error treatment are transferred to an archival data base where they can be retrieved from and analyzed off-line. An attempt is made to restart call processing at a point where as little system perturbation as possible will occur. This is often a difficult task. Errors can occur in a variety of places and the disturbing effects of an error are difficult to predict.

### **3.3 Fault recovery strategy**

Fault Recovery (FR) must change a system partially or wholly incapacitated by an error, believed caused by malfunctioning hardware, into a completely viable system that has shaken off the effects of the error.

FR avoids call disturbances perceptible to the customer. This requires that FR operate within timing constraints dictated by call processing. Only in exceptional cases does FR take excessive time and cause perturbations in call processing.

Errors must be detected as close to their source as possible. The further they propagate, the harder it is to find the source, the harder it is to discriminate the types of error, and the harder it is to clean up the deleterious effects of the error. Within economical constraints, error detectors in No. 4 ESS were placed in the system to provide detection as soon as possible.

Each error indicator and each error source leading to an error indicator has been given a position in an error priority structure. Priority is given to errors occurring during processor access of the periphery, over autonomous errors (see Section 3.1.4). Priority is given to unique error indicators, which allow for the fastest and most concise resolution, over nonunique error indicators.

Once an error indicator is chosen, the error must be classified. Classifications of errors include: software, transient, hard fault, and error analysis resolvable. Software errors occur when nonexistent hardware is accessed. Transient errors are those which are not repeatable via retries or other techniques. They may be caused by marginal hardware failures, systems noise, etc. Faults cause errors which will be repeated until the fault is removed. Error-analysis resolvable errors are those that cannot

be more precisely classified because of the nature of the error or inadequacies of FR.

The best of several techniques is chosen to identify the portion of the system that caused the error and to classify an error. The simplest technique is to assume the portion of the system containing a unique error indicator is at fault. Other techniques include retries of the sequences of events leading to the error, testing of the hardware involved using test data derived from the data present in the vicinity of the error, and testing of the hardware involved using fixed predetermined test data. Unique error resolution is much preferred over fixed-data testing. It is less time-consuming and more reliable as an identifier of the type and source of the error. Testing with predetermined test data can be likened to a minidiagnostic that runs unsegmented in real time. FR requires resolution to a configurable portion of the system and not to a replaceable module.

Once the type of error is ascertained and the reconfigurable portion of the system with the fault is identified, FR selects a course of action and a sequence table. This information is passed to error analysis. Error analysis will agree with the action or provide an alternate course of action based on the present error and consideration of its relation to past errors. FR will carry out the action finally agreed upon. All data collected during the treatment of the error is archived and the system is returned to normal processing.

### **3.4 Example**

The following is an example of what might happen if an interrupt occurred on an access of a duplicated signal processor's trunk status memory.<sup>4</sup> The central processor would be interrupted by an F level and control would be given first to the routine for peripheral error filtering. This routine would identify the source as failure of the central processor to obtain an All-Seems-Well (ASW) signal from the peripheral unit. Furthermore, it would identify the unit as a particular SP from peripheral address information saved by the processor at the time of interrupt. The SP's routing and error-source registers would be read and saved if the error condition permitted. Then a series of access tests on the SP in question would be executed to verify that the central processor and the peripheral bus were not at fault. If they were certified as good and examination of the SP's internal error-source indicators did not implicate the bus, further processing of the error would be turned over to the SP's unit fault recovery routine (SPFR).

SPFR would examine the SP's error source indicators and for our example find a number of mismatch indicators (internal sequencer mismatch, memory address and data mismatch). The sequencer mismatch is the highest priority of these mismatches and is treated by retrying the

failing order independently on each half of the duplicated SP. Let us assume the retries are inconclusive and SPFR has taken all the time it can. SPFR then classifies the error as error-analysis resolvable, picks a half to be removed from service and diagnosed, chooses a sequence table, and consults with error analysis.

For our example, error analysis already has a similar error for this particular SP on record and notes SPFR chose the same half for removal the last time. Error analysis then changes the decision to remove the other half and SPFR carries out the decision. All data gathered about the error as well as the action taken is archived and the system is returned to normal processing. The SP diagnostic subsequently finds a faulty pack in the logic that causes the two halves to execute peripheral orders in synchronization and the repair is made. Some time later, after the SP has run in full duplex for a period of time, the error analysis past history for this event and the events leading to it are automatically removed by the system from the current error analysis files.

### **3.5 Expected results**

The expected results of hardware recovery are:

- (i) One interrupt to recover from hard unique errors or software-type errors.
- (ii) One to two interrupts for nonunique hard errors contained within adjacent units.
- (iii) Two or more interrupts for error-analysis-resolvable errors, transient errors, and errors with effects propagated over several units along the interunit communication and control paths.

For errors that deviate from these expected results in No. 4 ESS, new or modified sequence tables will be added. Sequence table structures were designed with change in mind. To avoid changing the FR control structures or routines that are inseparable from the hardware they interact with, the decision criteria for treating a particular error source indicator and/or type of fault over time are changed by modifying the error-analysis sequence tables. These tables are a series of macro expansions which can be easily changed with a high degree of confidence that the change is correct and will not cause unexpected side effects on FR recovery from the target errors.

In general, our expectations have been met. Our experience has led to some changes in the original sequence tables to treat high-frequency transients of short durations and to treat errors whose effects propagate further in the periphery than had first been expected. The sequence table structure has been useful in implementing these changes in the decision criteria.

## **IV. DIAGNOSTICS**

### **4.1 Overview**

#### **4.1.1 Objectives**

The basic object of a diagnostic program is to detect and locate hardware faults. The diagnostic program accomplishes this objective by:

- (i) Applying inputs to the unit under test.
- (ii) Comparing the outputs with the expected outputs in order to detect the fault.
- (iii) Using the pattern of failing tests to locate the fault.

Typically, a diagnostic program is designed to detect greater than 90 percent of the faults, and for these faults resolve the problem to an average of less than five circuit packs.

Since the repair process is a deferred task involving manual action, the execution time of the diagnostic is not a prime consideration. However, the diagnostic program contains many thousands of tests, and it is desirable to minimize the memory required to store these tests. Thus in No. 4 ESS the diagnostic program is designed to minimize program size at the expense of some execution time.

Unlike most other programs within the system, the diagnostic program listings are used by the craftsperson to manually analyze failure data. The diagnostic programs are therefore designed to be easily read and understood. For ease of use by craft, the diagnostic programs are grouped according to unit type. Within each diagnostic, the tests are subdivided into groups with each group testing well-defined blocks of circuitry.

Since diagnostic programs are often affected by hardware changes, the diagnostic is designed to be easily modified via future generic updates.

#### **4.1.2 Diagnostic execution**

A diagnostic program execution can be stimulated from any of several sources. These sources include:

- (i) Fault recovery following a maintenance interrupt
- (ii) System recovery following system reinitialization
- (iii) The craftsperson during the repair process
- (iv) Routine exercise to perform periodic testing of the frame

Once initiated, the diagnostic program executes concurrently on an interleaved time basis with normal call processing in a noninterfering manner. During diagnostic execution the test results are printed on a teletypewriter. At the conclusion of the diagnostic, a summary message

is printed. This message indicates one of the following: all tests passed (ATP), some tests failed (STF), conditional all tests passed (CATP), or no tests run (NTR). The CATP response is printed whenever it is necessary for the diagnostic to skip tests because of the unavailability of a system resource. Examples of system resources needed by diagnostics include buses, mate units, and pulse points.

If there were any test failures, a list of suspected faulty circuit packs may also be printed at the teletypewriter. Unlike previous ESS, the translation between failure pattern and suspect circuit packs is performed on-line. In all cases, the suspect packs are ordered with the most probable packs printed first. The repair procedure requires sequentially replacing each circuit pack on the list until a diagnostic ATP or CATP condition is reached. The diagnostic program is manually initiated between each circuit pack replacement to check if the fault has been corrected.

## **4.2 Implementation**

### **4.2.1 Test design language**

In order to facilitate the writing of the diagnostic program, a special-purpose language, denoted DIAL, was developed. The DIAL language is oriented to the special requirements of diagnostics in the ESS environment. Statements in DIAL can be divided into two classes: testing statements and general purpose statements.

An example of a testing statement is:

```
STM1 TMSOP OPER (READ), OPAD (4TGOP),  
                                MASK (4TGM), EXPR (4TGE)
```

In this case, "STM1" is the statement label, "TMSOP" identifies the type of unit being tested (TMS), "OPER (READ)" identifies this as a read from a unit, "OPAD (4TGOP)" is the input to the unit that will elicit the reply, "MASK (4TGM)" masks the reply from the unit to certain specified bits, and "EXPR (4TGE)" is the expected reply from the unit. For a write to a unit without a corresponding read, the mask and expected result field are defaulted. Similar statements exist for each peripheral unit. In addition many statements exist in common for all peripheral units. Commonality of testing statements is enhanced since most peripheral units have the PU bus as their input/output medium.

The general-purpose statements are similar to most other high-level languages. Statements exist to move data in memory, perform arithmetic and logical functions, call subroutines, etc. The language is procedure-oriented in that the total program is subdivided into a set of "phases" and subroutines called by these phases. Each phase has only one entry



and one normal exit point. Each phase tests a functional block of circuitry and the phases are executed in order.

The DIAL general-purpose statements have several unique aspects. The DOLOOP statement allows for the shifting or rotating of specified data patterns each time through the loop. This feature facilitates the generation of test patterns for regular logic. Another unique feature is that program branches are allowed only in the forward direction. This feature facilitates program reading and debugging; however, the main impetus for this restriction is that unique test numbers must be assigned at compile time to each test. If tests are skipped during diagnostic execution, the test number can be advanced correspondingly.

Assembly language coding of diagnostic tests is not allowed and DIAL is generally independent of the 1A Processor. The testing statement parameters are specified in terms of the unit inputs and outputs, not in terms of the 1A Processor. This independence of the host computer facilitates the writing of compilers for other machines. Compilers have been written for No. 1 ESS and a host of various minicomputer systems. These minicomputer systems execute the diagnostic programs in other environments such as in factory frame testing.

A compiler also has been developed to translate the diagnostic program into LAMP logic simulator inputs.<sup>5</sup> This tool made it possible to execute the diagnostic on a software model of the unit under test before the unit was physically available. Many hardware and software design errors were thus detected early in the development.

#### ***4.2.2 Test generation and evaluation***

The number of tests that must be generated required the development of several aids. One of these aids is DIAL<sub>1</sub>, which allows the tests that follow a repetitive pattern to be easily coded using the DOLOOP and subroutine features of the language. Another aid was the development of a set of programs which would map existing circuit pack tests into the frame diagnostic tests. Automatic test generation programs were also used to generate some of the tests.

The tests are evaluated by both physical fault insertion and the LAMP simulator.<sup>5</sup> Lamp provides a means to verify whether the test will pass on a fault-free machine. This feature is used to debug tests before the actual frame is available. Another feature of LAMP is the ability to measure and identify the number of faults that would be undetected by the diagnostic. This feature provided a measure of diagnostic effectiveness and indicated areas for diagnostic enhancement.

#### ***4.2.3 Diagnostic structure***

For No. 4 ESS, the DIAL compiler was written to generate a compact representation of the program in a "data table" or interpretative format.

An on-line control program interprets the data table at execution time to effect the execution of the diagnostic.

An interpretative program has another advantage in that other controlling operations can be implemented easily. One example is the automatic segmenting of the diagnostic program. To allow for concurrent execution with call processing the diagnostic execution is broken into time segments of approximately 3 ms. With an interpretative control, this segmenting is done at execution time with such variables as unit response time being automatically accounted for. For those rare cases where the segment boundaries must be explicitly specified, facilities exist in the language to define the segment boundaries at compile time.

Interpretative control also provides for manual interactive control of the diagnostic execution. Features in the interactive control subsystem allow the craftsperson to pause at selected points within the diagnostic execution, loop the diagnostic execution over specified addresses, etc. The input commands, received from the craftsperson by the control program, cause the diagnostic execution to conform to these commands. Automatic segmenting provides advantages to interactive diagnostic use, since the craftsperson can pause or loop the diagnostic virtually anywhere without taking the segment boundaries into account.

The diagnostic control program implementation is similar to that for 1A Processor Units. This commonality provides savings in design effort and forces uniformity of man-machine interfaces. Many features common to 1A Processor and peripheral diagnostics have proved to be very valuable. One common feature of 1A Processor and peripheral diagnostics is the ability to execute several diagnostics concurrently. This feature is especially valuable for peripheral diagnostics because of the large number of peripheral units. Interfering peripheral diagnostics are automatically prevented from executing concurrently.

#### **4.2.4 Diagnostic output**

Diagnostic output includes the raw data output of the failing tests and a list of suspect circuit packs. The raw data output includes the following information for each failing test: the unique test number, the test failure pattern, the actual response from the unit on the PU reply bus, the input to the unit on the PU address and write buses, and the location in the diagnostic of the failing test. The purpose of this raw data output is to present to the craftsperson an easily understood description of the test(s) that failed. This information would be used in the manual analysis of data whenever the suspect pack printout failed to locate the problem. Figure 4 shows a sample raw data teletypewriter output.

The listing of suspect circuit packs gives those packs that are most likely to contain the fault. Included in the output is the physical location of the suspect and the circuit pack code. Other special information

DGN:TMSP 0, CONTR 1 PH 5 STF (MISMATCHES=3)

| TEST | MISMATCH | SUPPLEMENTARY DATA                  |
|------|----------|-------------------------------------|
| 17   | 01000000 | 01000000 12024540 00774000 00000266 |

| test number | mismatch | response  | input to unit | location of  |
|-------------|----------|-----------|---------------|--------------|
|             |          | from unit |               | test failure |

Fig. 4—Sample raw data output.

concerning the pack, such as warning flags, may also be printed. Figure 5 shows a sample suspect circuit pack list.

#### 4.2.5 Trouble location methods

Several distinct methods have been developed to map the diagnostic failure pattern into the list of suspect circuit packs. One method is an on-line pattern analysis of the failure data. This method is especially applicable to cases where the logic is regular. For example, the address of the failing memory word plus the failing bit will normally uniquely identify the faulty circuit pack.

Another method of fault location is to match the failure pattern with a predetermined set of failure patterns. These predetermined failure patterns are gathered by a combination of physical fault insertion and fault simulation. The algorithm uses seven key parameters derived from the failure pattern in an attempt to attain as close a match as possible. This method is the same as used by most processor units.<sup>3</sup>

However, peripheral units generally use a method based on the circuit

```
M 36 ANALY:TLPPFILE TSIF 0, CONT 0 SUSPECTED FAULTY EQUIPMENT
TLPPFILE 57 ENTRY TIME 01/11/76 23:33:13
EQPT LOC CODE NOTE WT PS SYM SD HELPER ID
046-19 FA0557 9 11 8 4A024
042-25 FA0543 9 13 1 4A024
046-17 PA0540 5 9 2 4A024
042-21 FA0632 5 10 8 4A024
042-13 FA0540 5 3 3 4A024
042-19 FA0551 5 14 2 4A024
046-18 PA0540 5 9 1 4A024
```

Fig. 5—Sample suspect circuit pack list.

topology of the unit and the diagnostic failure pattern. For peripheral diagnostics, the failure pattern contains the individual bits that failed plus the "ADDRESS" of the point within the unit read for this test. This point (address and bit position) is known as a monitor point. These monitor points are typically flip-flops, internal registers, dc leads, parity bits, etc. The diagnostic failure pattern thus maps into a list of failing monitor points.

The circuit topology of the unit is contained in what is known as a "connectivity data base." This data base is a listing for each monitor point of the circuit packs associated with these monitor points.

The generation of the connectivity data base involves the following operations. First, all monitor points within the unit are identified. Second, a list of associated circuit packs for each monitor point is generated. This list of associated circuit packs is made up of the following two components:

(i) All circuit packs containing logic paths to this monitor point from external inputs or from other monitor points.

(ii) All circuit packs containing logic paths which transmit the state of this monitor point to external outputs.

Off-line, a list of circuit packs associated with each monitor point within the unit has been generated and stored on tape. This tape is accessed by the ESS resident diagnostic-results processing programs.

First, the on-line fault location procedure summarizes the monitor point occurrences in the failure pattern. Second, the union of the associated circuit pack lists for each failing monitor point is generated. Third, the circuit packs are ordered according to various criteria. Examples of possible criteria include number of occurrences, reliability data, number of gates in the logic path on this circuit pack, etc.

A trouble-locating method based on the circuit topology has several advantages. First, the method is independent of the diagnostic program. One can add tests to the diagnostic without affecting the trouble-locating method. With methods that rely on previously generated fault signatures, test enhancement is difficult since it must not affect the existing fault-signature data base. Second, the connectivity data base can be automatically generated from existing files containing the circuit description. This attribute is important if the trouble location procedure must respond to hardware changes. Other methods restrict such changes or force the regeneration of the data base.

The decision on which trouble-locating method to use is based on several considerations. In general, the decision revolves around the following points. Pattern analysis can be used for regular logic. Methods based on predetermined fault signatures are used for units that must have high trouble-location accuracy and resolution. Methods based on

circuit topology are used for the remaining units. It is also possible for a diagnostic to use a combination of methods based on different considerations within the diagnostic—for example, pattern analysis for the regular logic and connectivity for the irregular logic.

#### **4.2.6 Routine testing**

Since up to 25 percent of the hardware circuitry is involved only in maintenance operations, it is important to routinely exercise this circuitry. The usual hardware checks will detect only faults in the operational circuitry. The problem is determining the frequency of routine testing. Infrequent testing increases the possibility of multiple faults. Frequent routine testing decreases reliability by increasing the simplex operation time. For peripheral units, formulas were developed to calculate the optimal frequency of routine testing.

#### **4.3 Results**

Significant results were achieved in several areas of diagnostic program design. First, the high-level diagnostic language increased diagnostic programming productivity, standardized the diagnostic design effort, and enabled the diagnostic programs to be compiled into a form for use in several diverse applications. Second, the common structure encompassing both 1A Processor and peripheral diagnostics decreased the program integration effort, provided for a uniform man-machine interface, and led to commonality of design. Third, the various support programs such as LAMP decreased design effort and provided a way to evaluate tests independent of physically inserting faults into the machine. Finally, the use of the connectivity trouble-location methods enabled the generation of quality trouble-location algorithms with significantly less effort than has been applied with other methods.

### **V. SYSTEM RECOVERY**

Memory mutilation in either program store or call store can cause abnormal system operation. Such situations can be detected by monitoring various system characteristics and auditing certain data structures, as discussed in Section 2.3. Once it is decided that severe mutilation has occurred and remedial actions such as demand audits will not correct it, system recovery actions are taken to reconstruct a sane program and data base and to obtain a viable hardware configuration. System recovery basically consists of hardware reconfiguration and software initialization and it can be invoked either automatically under program control or manually.

## 5.1 Automatic system recovery

Automatic system recovery takes place when the program detects memory mutilation and determines that a severe recovery action is needed. Automatic system recovery is needed when remedial actions such as demand audits are not able to correct the problem causing abnormal system operation. These system recovery actions are termed phases of initialization and these phases increase in the severity of their corrective actions.

### 5.1.1 Justification

*5.1.1.1 Need for system recovery.* System recovery in the form of a phase of initialization is needed whenever one of several severe problems has occurred, affecting normal operation. Some of these are:

- (i) Mutilation of writable program store
- (ii) Loss of a vital system function
- (iii) Loss of a major facility
- (iv) Escalation of remedial actions
- (v) System start-up

System start-up is not a problem as such, but does require a phase of initialization and occurs whenever the system has been "down" for any length of time or whenever a complete new issue of the program is being loaded.

*5.1.1.2 Phase triggers.* There are specific triggers built into the No. 4 ESS that will cause a phase of initialization to occur. These phase triggers were chosen in an attempt to clear problems as quickly as possible which were determined to be severe enough in nature that the taking of further remedial actions (or any action in some cases) would only delay their correction. The basic phase triggers are:

- (i) Problems in the software integrity control program (SICO)
- (ii) Excessive lower phases (phases 1 and 2)
- (iii) Program sanity timer time-out
- (iv) Excessive maintenance interrupts
- (v) Excessive audit requests
- (vi) Excessive K-level interrupts
- (vii) Nonservice of interject programs
- (viii) Nonservice or mutilation of the software clock
- (ix) Invalid entry to the interject monitor
- (x) Duplex failure of a unit

These triggers each request a specific phase of initialization and these requests can be escalated based upon the recent occurrence of other phases of initialization.

### **5.1.2 Recovery sequence**

The sequence of system recovery consists of hardware reconfiguration and recovery as well as software recovery or initialization.

**5.1.2.1 Design philosophy.** The phases of initialization were designed such that each phase increases in severity. The lowest-level phase was designed to be the least severe and fastest running, and was intended to clear a majority of the problems. Most problems can be cleared by a short, direct phase and do not require a complete system initialization. All phases are designed with the philosophy of initializing memory as opposed to auditing and correcting memory. Initialization is generally faster and more effective at clearing severe problems or memory mutilation than detecting and correcting errors.

The phases are numbered 1 through 4, phase 1 being the least severe. It is short in duration and it assumes all hardware is good and all permanent memory is good. It initializes specific areas of transient memory. The phase 1 also saves all calls.

Phase 2 is next in the escalation order and it assumes all processor hardware is good and all permanent memory is good. It basically reconfigures the peripheral hardware and initializes all of transient memory. It saves stable calls.

Phase 3 assumes nothing is good. It first establishes a processor hardware core and then a complete processor hardware configuration. Next permanent memory is verified and/or reinitialized. The peripheral hardware is configured and then transient memory is initialized. Phase 3 saves stable calls. Phases 1 through 3 can be activated automatically or manually.

Phase 4 is the highest-level phase and it can only be activated manually. It is the same as the phase 3 except that it tears down all calls and as a result totally reinitializes the entire system.

**5.1.2.2 Software control structure.** The software integrity control (SICO) program controls the execution of system recovery. SICO runs the appropriate hardware and software recovery routines based upon the phase being run and then passes control to the software initialization program (SINT), which performs the initialization of memory and the saving of calls. SICO can also escalate any automatically generated phase request according to upon the trigger and the recent phase history.

Once SINT has completed the software initialization, SICO will once again be given control to prepare the system for restarting after the phase. The duration of the phases are basically dependent upon the phase which is run and the size of the office. Therefore, phases can last from 1 second up to 1 minute or slightly longer. As a result of this outage, certain actions must be taken by SICO before restarting the system, such as clearing out the buffers in the signal processors and instituting overload controls to control the anticipated traffic buildup. SICO also runs

certain specialized restart routines and formats the printout of the phase results.

*5.1.2.3 System hardware recovery.* As discussed in the previous sections, the processor hardware recovery is accomplished first and is followed by: permanent memory verification and initialization, peripheral hardware recovery, and transient memory initialization (including saving of calls). The processor recovery establishes a hardware core and then basic-sanity tests are run on this hardware core. Once these sanity tests are passed, the entire processor hardware complex is established. Permanent memory is then verified by a hash summing procedure and any failing blocks of memory are reinitialized using the backup copy on the disk file system.<sup>3</sup>

Peripheral hardware recovery is then run in four levels. Successive levels increase in severity. The level run depends upon the phase being requested, the triggers, and the recent phase history.

*5.1.2.4 System software recovery.* The system software recovery is done after the processor hardware has been configured, the permanent memory verified and initialized, and the peripheral hardware has been reconfigured. The SINT program performs this initialization of the transient memory which includes the saving of stable calls on phases 2 and 3. SINT is organized into a single control module and several specific initialization modules. The control module will select the appropriate initialization modules to be run depending upon the phase being requested and will then execute them in a predetermined order. Each initialization module performs a fairly self-contained initialization function such as zeroing all scratch areas or initializing the network maps in call store. When SINT completes the software initialization, it passes control back to SICO to prepare the system for restart.

## **5.2 Manual system recovery**

Manual intervention may be necessary to regain system sanity or overcome system deficiencies. Before such action can be taken, one must be able to recognize the need for manual intervention. There are several types of indicators.

The master control console<sup>6</sup> has a number of visual displays that indicate system performance. If these alarm repeatedly, manual action should be taken. An excess of audits or interrupts can indicate system failure that requires manual intervention if the system does not initiate effective corrective action within a reasonable time (e.g., phase of initialization).

Once a need for manual action is ascertained, the correct manual action must be chosen from a number available. All combinations of processor configurations can be forced and tried if the processor is insane.



System initialization phases 1 through 4 can be activated manually with options that:

- (i) Inhibit error indicators and force the system to ignore errors
- (ii) Initialize short-term transient call store
- (iii) Initialize long-term transient disk storage
- (iv) Remove system data-base changes just activated

If the system is sane enough to perform input-output requests, a host of commands are available for manual intervention. All configurable hardware units in the system can be forced in and out of service, have their individual internal-error indicators inhibited and uninhibited, and have their internal memory and control points examined and modified via commands from a teletypewriter. Almost all actions that are performed automatically by the system can also be initiated by manual action such as demanding audits, requesting phases of initialization, or requesting diagnostics. Many additional actions that cannot be performed automatically can be executed by the craft people.

## VI. SUMMARY

In order to meet the very stringent dependability and maintainability requirements, four very large software packages were developed in the area of maintenance software. The software-error recovery package detects and recovers from software malfunctions. The hardware-error recovery package recovers from hardware faults through fault detection and reconfiguration. Diagnostic programs are used to detect and locate hardware faults in a given faulty unit. The fourth package, system recovery, provides for overall coordinated system recovery from multiple or very severe hardware and software malfunctions. All four software areas were developed concurrently, each having to meet its own stringent area requirements as well as having to interface with the other maintenance software areas. The end result was a unified software package that covers the detection of and recovery from hardware, software, and system malfunctions both automatically and manually.

## REFERENCES

1. R. W. Downing, J. S. Nowak, and L. S. Tuomenoksa, "No. 1 ESS Maintenance Plan," *B.S.T.J.*, 43, No. 5, Part 1 (September 1964), pp. 1988-2018.
2. E. J. Aitcheson and R. F. Cook, "No. 1 ESS ADF Maintenance Plan," *B.S.T.J.*, 49, No. 10 (December 1970), pp. 2831-2856.
3. P. W. Bowman, M. R. Dubman, F. M. Goetz, W. R. Hudgins, D. E. Lutz, and E. H. Stredde, "IA Processor: Maintenance Software," *B.S.T.J.*, 56, No. 2 (February 1977), pp. 255-288.
4. J. Janik, Jr., J. H. Huttenhoff, M. F. Slana, and F. H. Tendick, "Peripheral Hardware," *B.S.T.J.*, this issue, pp. 1029-1065.
5. "LAMP: Logic Analyzer for Maintenance Planning", *B.S.T.J.*, 53, No. 8 (October 1974), pp. 1431-1555.
6. A. H. Budlong, B. G. DeLugish, S. M. Neville, J. L. Quinn, and F. W. Wendland, "IA Processor: Control System," *B.S.T.J.*, 56, No. 2 (February 1977), pp 135-180.



## **No. 4 ESS:**

# **Network Management and Traffic Administration**

By T. V. GREENE, D. G. HAENSCHKE, B. H. HORNBAUGH,  
and C. E. JOHNSON

(Manuscript received July 30, 1976)

*The No. 4 Electronic Switching System has been planned to provide operational ease in the efficient management of traffic-sensitive equipment for the purpose of maintaining a high quality of service. The network management function has the goal of optimizing the completion of calls during periods of traffic stress. In No. 4 ESS, innovative real-time control and surveillance features are provided to meet this goal. Traffic administration involves the activities of personnel in managing the traffic-sensitive equipment in an efficient, timely, and economical fashion. These activities depend upon the collection of data which reflects the operating characteristics of the No. 4 ESS, and the reporting of information in a form tailored to the user functions.*

## **I. INTRODUCTION**

High-quality service requires the proper planning for the quantities of central office and trunking equipment and the effective utilization of this equipment in the operational environment. The network management and traffic administration functions in the No. 4 ESS are planned with these objectives in mind. The network management function is directed at maintaining a high level of service during unusual traffic situations. The traffic administration function has two aspects: network engineering and machine administration. Basically, network engineering encompasses the planning function associated with the dimensioning of equipment, and machine administration consists of the ongoing activities aimed at meeting service objectives. A fundamental distinction between these activities is their differing time frames:

(i) The network management activities deal in real time with the unexpected or unusual situation.

(ii) Machine administration activities deal with maintaining service objectives on a daily and monthly basis.

(iii) Network engineering activities are concerned with future planning for customer service.

### **1.1 Network management**

It is a property of modern telephone networks with common control switching and alternate routing arrangements that they are highly efficient and economical under engineered load conditions but deteriorate under overloads. Once this decline sets in, it is difficult to recover even if the load level is reduced to normal.<sup>1</sup>

The reasons for this decline in efficiency are excessive alternate routing<sup>2</sup> and regenerative switching (queuing) delays. The latter has the more direct impact on performance and its prevention is a prime objective of the No. 4 ESS network management system. Regenerative switching delays, if left uncontrolled, can quickly spread throughout the network, causing the type of decline in carried load shown in Fig. 1. This figure is from Burke who first analyzed this phenomenon in detail.<sup>3</sup> The mechanism at work here is waste usage of common control equipment which, for instance, can be caused by transmitters in one office waiting for receivers in another. A similar throughput decline occurs also within a single heavily overloaded No. 4 ESS because of loss in internal efficiency due to increasing real-time overhead.

The network management system must provide the real-time control and surveillance capabilities that are needed to realize the economical advantages of alternate routing and modern switching systems, without compromising service during stress. In the mid-1960s the dynamic overload control system<sup>3-5</sup> and manual trunk group controls were introduced, which have proven to be reasonably effective during peak-day overloads such as Christmas and Mother's Day. However, these controls are not code selective and, therefore, not very effective during focused overloads which are characterized by a surge of traffic from all parts of the network to a small number of offices or destination codes. In No. 4 ESS, a major innovation in surveillance capabilities is the automatic determination of destination codes which have a low probability of completion and are said to be "hard to reach." This information reveals equipment failure and traffic congestion at points far removed from the No. 4 ESS or its trunking field.

The integration of hard-to-reach code data with automatic controls accomplishes the objective of making this system responsive to a wide range of trunk facility and machine overloads. This enhanced responsiveness of the automatic controls should minimize the need for manual control interactions. Human reaction times are often too slow to prevent

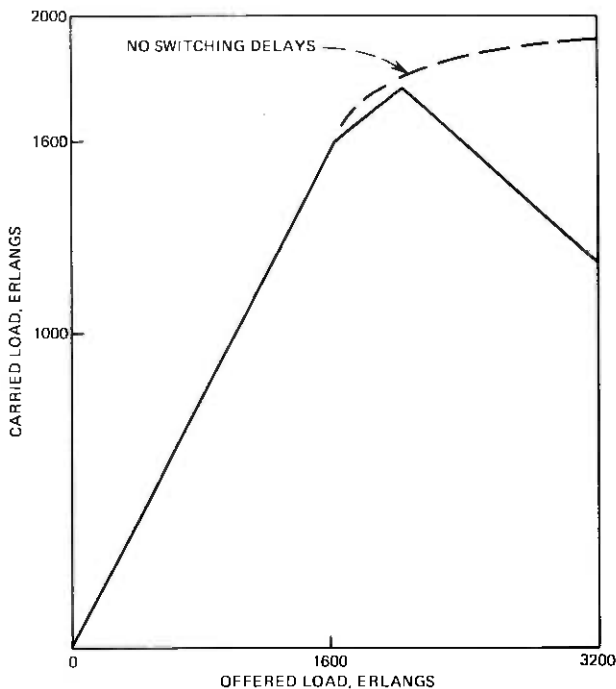


Fig. 1—Network performance under overload.

the earlier-mentioned decline in network efficiency during unexpected overloads.<sup>6</sup> The system has been planned so that network managers can concentrate their efforts on the general supervision of real-time network performance, analysis of network weak spots before they become service-affecting, preplanning of control strategies, fine tuning of the automatic system, and intervening with manual control actions only in problems requiring human judgment. For the surveillance of real-time network performance, a display system is provided which furnishes the network manager with preprocessed data. In the past, network management problems had to be determined from traffic registers, trunk-group busy lamps, and a few machine-status lamps. In No. 4 ESS, potential problems are automatically reported and detailed problem investigation is accomplished using CRT pages programmed for efficient data analysis.

### 1.2 Traffic administration

The No. 4 ESS, serving as a switching node in the telecommunication network, has to have its central office equipment and trunking to other offices properly engineered and administered to meet its service objectives.

The quantity of provided equipment is determined by the size and character of anticipated traffic loads. In planning the growth of the telephone network, the network engineer must determine how much equipment will be needed so that it can be ordered and installed in time to provide an objective level of service throughout the engineering periods ranging from 6 months to 2 years in length.

In the No. 4 ESS, as in other systems, the network engineer is responsible for both central office and trunk equipment. Central-office engineering involves the establishment of configurations for new offices and office additions, and the specification of quantities of switching subsystems to be installed. Trunk engineering involves the determination of which new trunk subgroups\* will be established, the number of trunks each should contain and the changes to be made in the size of existing trunk subgroups.

Once the equipment is installed and working, the machine administrator is concerned with potential and real service-affecting problems occurring within shorter intervals of time than those of interest to the network engineer. However, the machine administrator is not expected to react as quickly as the network manager to service problems, since the network manager is primarily responsible for maximizing the flow of traffic through the network during unexpected events of a transient nature (such as earthquakes and storms).

In preventing and solving service-affecting problems, the machine administrator is responsible for the processing, collection, analysis, and distribution of traffic data. The performance and loading of central-office equipment and trunk equipment should be periodically analyzed and trended to assure sufficient capacity for handling existing and future traffic loads. The machine administrator is also responsible for maintaining the objective service level during office transitions when equipment is repaired, added, or replaced.

The basis for satisfying the information needs of these engineering and administrative functions is a comprehensive set of measurements. The measurement data which is maintained for the previous hour reflects both service as seen by the customer as well as performance of the switching equipment. Subsets of this data are extracted and maintained for extended intervals to tailor several information bases in a cost-effective fashion.

Through sorting and processing the collected measurement data, the information needed by network engineers and machine administrators

---

\* The term "trunk subgroup" is No. 4 ESS nomenclature for the set of trunks in a given route with certain common features such as directionality, pulsing type, and transmission delay characteristics, i.e., satellite versus terrestrial; each trunk subgroup is an entity in itself for traffic measurements and network management control.

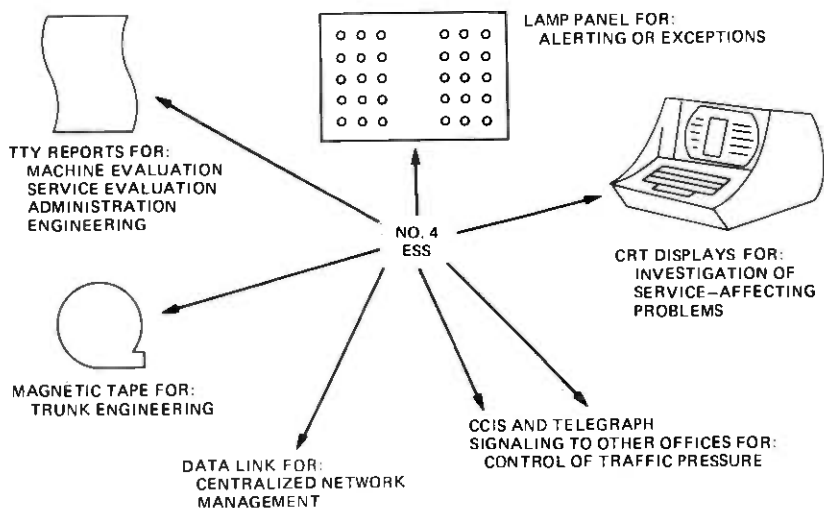


Fig. 2—System interface.

is provided in the form of hard-copy reports which can be flexibly scheduled and directed to work centers.

The type of the report can vary from an overview report obtained by extensive processing of data to a detailed printout of the individual measurements. This structuring of reports, from overview to detail, is intended to minimize the volume of paper generated. In general, overview reports should satisfy the information needs of office personnel and will be supplemented with detailed reporting only when exceptions are indicated by the overview report.

Along with hard-copy reports, a magnetic tape can be scheduled for downstream processing. This tape contains the statistics for the trunk subgroups in the No. 4 ESS which, along with similar information from other toll offices, are analyzed by a downstream process to produce forecasts of message-trunk requirements.

## II. SYSTEM INTERFACES

The network management and traffic administration features provide interfaces with office personnel and other machines (Fig. 2).

The main personnel-machine interface is realized with Model 40 teletypewriters which are serviced through the I/O channels of the 1A Processor. The reports for general administration and engineering the office can be scheduled to be output on any of the standard I/O channels. Printed on fanfold paper, each page of report is contained on a separate sheet for assembly into a report folder.

The real-time surveillance system for network management provides

for both local and remote network management centers. The Model 40 teletypewriter terminals for providing CRT displays can be remoted through a standard arrangement using 202 data sets. The lamp-driving circuitry for the network-management display panel can interface with an E2A telemetry system for the remoting of the lamp display. The panel consists of Nixies to provide numeric information and lamps for indicating YES/NO type of exceptions which are color coded to denote problem severity. Since the panel is an exception-reporting medium, it has been designed to have a darkened appearance if the system is operating normally.

Besides direct communication with office personnel, three machine-to-machine interfaces are provided. For network management purposes, communication interfaces are provided to pass control information between switching offices and to provide data to centralized network-management systems. For traffic administration, a magnetic tape containing trunk utilization statistics is generated for processing by a downstream system.

The network management control plan calls for the transmittal of machine congestion signals between interfacing offices. There are three levels of dynamic overload control, or DOC, signals that can be transmitted to control the traffic on a trunk subgroup between the offices. To control multifrequency and dial-pulse traffic, signals are transmitted over telegraph channels or data channels between the offices. For integrity purposes, the office receiving a DOC signal will send an acknowledgment signal to the sending office. In this way, impairments to the signaling channels can be detected either through failure to receive an acknowledgment or the receipt of a false acknowledgment. For trunk subgroups that are served by Common Channel Interoffice Signaling<sup>7</sup> (CCIS), the DOC information is passed between offices using the CCIS channel. DOC acknowledgment signals are not used in the CCIS case since the CCIS signaling channels have built-in self-checking features.

A data-link interface is provided for centralized surveillance and control. In particular, standard arrangements will be available for transmitting network management data to a remote centralized network management system called EADAS/NM (Engineering and Administrative Data Acquisition System/Network Management). EADAS/NM is presently being introduced into the Bell System to provide network management surveillance and manual control of an entire geographical area, such as a state, not just a single office. The No. 4 ESS will supply EADAS/NM with a subset of its measurement and control status data for processing and display through EADAS/NM.

The final machine-machine interface provides trunk engineering data to a downstream processing system. The No. 4 ESS generates a standard label tape which contains hourly statistics for all the trunk subgroups



in the office. An identification map for the trunk subgroups in the office is provided daily on this tape to furnish a means for checking the consistency of the trunk assignment data contained in the data bases of the No. 4 ESS and the downstream processing system.

### III. NETWORK MANAGEMENT CONTROL

The goal governing the application of network management controls is to give the best possible service during overloads by utilizing all available facilities as efficiently as possible. These control actions, whether manually or automatically instituted, modify the routing of traffic. Such actions, however, are not a substitute for proper engineering and equipment provisions.

#### 3.1 Hard-to-reach code analysis

A Hard-To-Reach (HTR) code is a 3- or 6-digit destination code to which successfully outpulsed calls have a very small chance of completing. If the probability of completing through the distant network is very low and the outgoing trunk groups or connected switching offices are congested, the waste usage of these overloaded network resources for traffic to HTR points should be prevented.

To this end, the No. 4 ESS automatically monitors the completion rate on all calls to each numbering plan area (NPA) code. It also monitors call completion to each central office (NXX) code in its home NPA and in as many as six other NPAs which can be selected by the network manager in real time. Every 5 minutes, the No. 4 ESS counts the number of "Network Attempts" (NA) that are successfully forwarded to the next office, and of those, the number of "Ineffective Network Attempts" (INA) which abandon without receiving answer supervision. The INA count represents the number of calls that fail in the distant network for any reason, including line busy and line doesn't answer conditions. The ratio of abandonments, INA, to network attempts, NA, is called the INA rate. Normally, the INA rate will run about 20 to 40 percent depending on the incidence of line busy and doesn't answer conditions. In focused overloads, the INA rate may be in excess of 90 percent.

The No. 4 ESS places an NPA or NXX code on the HTR list if for a statistically sufficiently large NA:

$$\frac{\text{INA}}{\text{NA}} > T_1$$

where the on-list threshold level,  $T_1$ , is typically 70 percent. Once a code is declared hard to reach, it will be automatically restored to normal if:

$$\frac{\text{INA}}{\text{NA}} < T_2 < T_1,$$

where the off-list threshold level,  $T_2$ , is typically 60 percent. The threshold value,  $T_2$ , is chosen to be less than  $T_1$  to avoid oscillation when the control of HTR codes results in a reduction in the INA rate but the basic hard-to-reach cause remains in the network.

By definition, HTR codes are those codes with a very low probability of completing after they have been forwarded to the next office. They do not reflect completion problems the No. 4 ESS may encounter prior to outpulsing, such as no-circuit conditions and time-outs. These failures prior to outpulsing are called "Ineffective Machine Attempts" (IMAs). The No. 4 ESS gathers IMA data for the same codes for which it gathers INA data. The IMA counts are available to the display system and provide the network manager with valuable per-code switching machine performance data. However, a No. 4 ESS will not use the IMA counts for entering codes on its own HTR list. This is done because a high IMA count for a given code is not an indication that calls to this code will have a small chance of completing once they succeed in seizing outgoing trunks.

The No. 4 ESS's ability to determine HTR codes anywhere in the network can benefit not only the No. 4 ESS but also other switching offices that are able to administer an HTR code list for use with controls but which are unable to make an HTR code determination on their own, for example, No. 4A toll crossbar offices arranged for CCIS.<sup>7</sup> For these offices, the No. 4 ESS can serve as a network information center by informing them of HTR code problems.

### **3.2 Automatic controls**

#### **3.2.1 Protective automatic controls**

Protective network management controls are employed during overloads to prevent the decline in carried load due to switching delays shown in Fig. 1, and to achieve best possible use of available trunk facilities. Two new types of protective controls are employed automatically by the No. 4 ESS: selective dynamic overload control (SDOC), which responds to machine congestion, and selective trunk reservation, which responds to trunk congestion.

The stimulus to SDOC comes from a connected switching office that is sensing machine congestion. Two levels of machine congestion, a "low" and a "high" level, can be sensed with present nonselective DOC as well as by the No. 4 ESS overload-control program. As shown in Fig. 3, the No. 4 ESS will typically respond to the "low" congestion signal by restricting HTR traffic that is headed for the congested machine. Only when a "high" congestion signal is received will other traffic be restricted, typically all alternate-routed calls. The No. 4 ESS's ability to respond to "low" congestion signals with control only if an HTR problem exists,

| SELECTIVE<br>DYNAMIC OVERLOAD<br>CONTROL  |                                      | SELECTIVE<br>TRUNK RESERVATION    |
|---|--------------------------------------|-----------------------------------|
| CONGESTION LEVEL OF<br>REQUESTING MACHINE | TRAFFIC<br>CONTROLLED                | TRUNKS REMAINING<br>IDLE IN GROUP |
| NONE                                      | NONE                                 | $n > n_1$                         |
| LOW                                       | HTR ONLY                             | $n_1 \geq n > n_2$                |
| HIGH                                      | HTR<br>+<br>OTHER<br>ALTERNATE ROUTE | $n_2 \geq n$                      |

Fig. 3—Protective automatic controls of the No. 4 ESS.

coupled with its ability to transmit and receive such signals via CCIS without the extra cost of separate signaling facilities, allows a much wider deployment of SDOC than was possible with nonselective DOC.

The No. 4 ESS overload-control program continuously looks for shortages in real time and common equipment, and will initiate the "low" and "high" congestion signals when shortages are detected. It also governs the allocation of its own resources during overloads to prevent loss in internal efficiency. For instance, it will postpone nonessential tasks and, as a last resort, reduce the accepted traffic load if a sufficient load reduction does not occur through the use of DOC controls. The No. 4 ESS is also arranged to transmit a unique signal when it is no longer able to process new calls because of a major software or hardware failure resulting in a system outage. This signal is broadcast to connected offices to protect them from wasting resources while attempting to forward calls to the failed No. 4 ESS.

The other type of protective automatic control, called selective trunk reservation (STR), responds to trunk congestion as measured within the No. 4 ESS by the number of trunks remaining idle in the associated trunk subgroup. The response to trunk congestion, also shown in Fig. 3, is quite similar to that with SDOC. When the number of idle trunks,  $n$ , is less than  $n_1$ , access will be denied to the few remaining idle trunks only for HTR traffic that is headed for the congested trunk subgroup. If this HTR definition does not apply, no controls are taken. When trunk congestion builds up and less than  $n_2$  idle trunks remain,  $n_2 < n_1$ , trunk access can also be denied to other alternate-routed traffic. However, this second control step is usually applied only to the last-choice trunk subgroups for call routing, called finals, with the objective of protecting service of direct-routed (nonalternate-routed) traffic during heavy alternate routing. The No. 4 ESS will be able to meet this objective without the

trunk penalty associated with the present practice of splitting the final trunk group into two groups, one dedicated to direct-routed traffic, the other to alternate-routed traffic.

Calls denied access to an outgoing trunk subgroup by SDOC or STR can be canceled and routed to an announcement, or can be skipped over that trunk subgroup to an alternate route with idle capacity. SDOC and STR are more powerful than nonselective DOC because they are able to respond automatically to a much wider range of overload problems. They can be applied without the hierarchical restrictions inherent in the older controls. They are compatible with existing controls and need not be deployed extensively to be of benefit.

These claims of increased effectiveness of SDOC and STR have been verified by computer simulations under conditions of peak-day and focused overloads. One set of results, derived from a simulated 24-switching-office network<sup>4</sup> is shown in Fig. 4. The results apply to a focused overload during which the load offered to a given office increased 8-fold compared with normal levels. This type of overload is equivalent to that experienced by offices during the 1971 earthquake in southern California. The figure shows the number of successful messages in progress versus time, measured from the beginning of the overload. Without controls, we see the transient decline in carried load, which is similar to the decline shown in Fig. 1 for the static case. Present automatic controls improve call completion but are unable to maximize network utilization. STR, without SDOC, is able to keep network completions high for about an hour. Eventually, however, the buildup of ineffective short-holding-time attempts reduces trunk subgroup occupancies below the point where STR will trigger. Nevertheless, STR alone can keep the network operating efficiently for a sufficiently long time to allow the network manager to intervene with additional manual control. This is particularly significant since STR, unlike SDOC, is a completely autonomous protective control which requires no signaling channels from distant offices.

STR combined with SDOC provides close to optimum call-carrying capacity<sup>8</sup> and furnishes a marked improvement over present nonselective automatic controls. The actual improvement of SDOC and STR over present controls amounted to over 60 percent in the scenario presented in Fig. 4. While these improvements are greatest for focused overloads, simulation results indicate that similar benefits are obtained for peak-day overloads, such as those encountered on Christmas Day.

Further analysis of the simulation results indicate that selective blocking of HTR traffic with SDOC and STR increases network efficiency sufficiently to also benefit HTR traffic and, therefore, improve service into overloaded offices. An even greater service improvement results for customers calling out of the overloaded offices. This means that calls

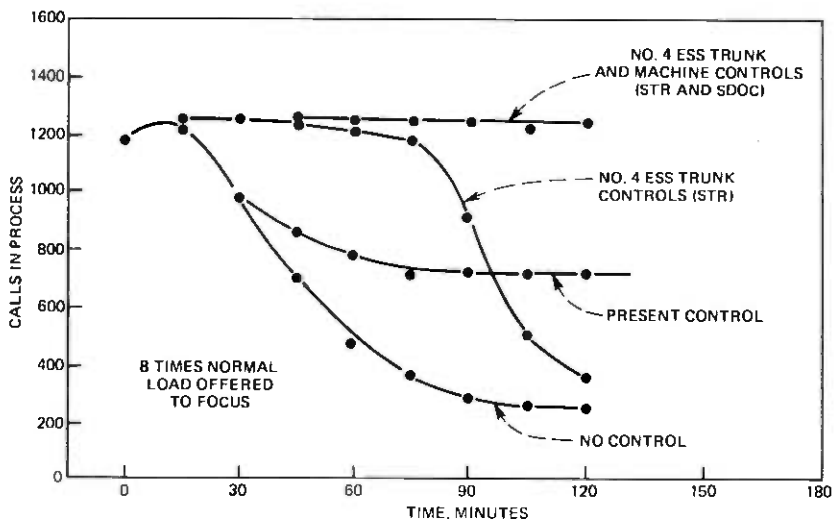


Fig. 4—Control of focused overload.

from the overloaded offices are automatically given the desired preference over incoming calls. Simulation results also show that worthwhile improvements can be attributed to SDOC and STR during their initial deployment when only a small portion of the network consists of No. 4 ESSs.

### 3.2.2 Expansive Automatic Controls

In contrast to the protective controls which restrict access to overloaded facilities, expansive controls take advantage of idle capacity on "out-of-chain" routes, i.e., on routes which are not within the design of the hierarchical routing structure. Whereas expansive controls have been exclusively manual up to now, the No. 4 ESS permits out-of-chain routing on an automatic basis.

The new feature which accomplishes this is called Automatic Out-Of-Chain (AOOC) routing. It permits calls which overflow the final in-chain trunk subgroup to be offered to out-of-chain trunk subgroups. Out-of-chain trunk subgroups are preassigned according to destination codes so that a call overflowing the final in-chain trunk subgroup will be offered to one of up to seven out-of-chain trunk subgroups that have good connectivity toward the code's destination. When more than one out-of-chain trunk subgroup is provided, traffic will be spread evenly over all trunk subgroups.

AOOC routing is allowed only if there is ample idle capacity in the out-of-chain trunk subgroup, in the "via" office at the far-end of that trunk subgroup, and in the outgoing trunking field from the via office

towards the call's destination. This controlled use of out-of-chain routes is accomplished by turning off the availability of an out-of-chain trunk subgroup for a timed interval when any of the following conditions occur: less than a predetermined number of trunks idle towards the via office; the receipt of a "low" or "high" machine congestion signal from the via office; or the receipt of a CCIS trunk congestion message indicating that the last call encountered an all-trunks-busy (no-circuit) condition at the via office or a subsequent office.

The full protection and discipline inherent in AOOC routing will be obtained if the out-of-chain trunk subgroup to the via office is equipped for CCIS. With CCIS, each call forwarded to the via office will have a special traveling classmark attached. This classmark instructs the via office to restrict outgoing trunk selection to the most direct route toward the call's destination. If no idle circuits are found in a direct route, the via office will return the above-mentioned CCIS trunk-congestion message, even when alternate routes are idle. This forces the temporary turn-off of out-of-chain routing for traffic that the via office cannot complete on a direct route. The dynamics inherent in AOOC routing promises to be the key to a more effective utilization of idle capacity on out-of-chain routes than has ever been possible with manual methods.

### **3.3 Manual controls**

Even though the emphasis is on improved automatic controls, manual controls are needed for solving problems which are not recognized by the automatic system because they require human judgment and interpretation of environmental data. In addition, the network manager must be allowed to override the automatic system and fine-tune its response.

The extensive set of manual controls provided with the No. 4 ESS is quite similar to that available in other major toll switchers. It includes the ability to block calls to any 3-, 6-, 7-, and 10-digit destination code specified by the network manager and to route blocked calls to a specially worded announcement. It also includes protective and expansive controls on a per trunk subgroup basis.

Traffic offered to a trunk subgroup can either be canceled and routed to announcement, or forced to skip the trunk subgroup and advance to an alternate routing choice. Traffic overflowing a trunk subgroup can be canceled or rerouted to an out-of-chain route. The application of manual trunk subgroup controls can be limited to HTR codes which is a new capability and provides a high degree of selectivity. Other options provided are the ability to specify control percentages, ranging from 25 percent to 100 percent, and a choice between direct and alternate-routed traffic for trunk subgroup skip and cancel controls. Manual override

capability is provided for all automatic controls as well as for the HTR code detection systems. A code can manually be declared as hard to reach—or excluded from HTR treatment—for any outgoing trunk subgroup selected by the network manager.

#### **IV. NETWORK MANAGEMENT SURVEILLANCE**

The No. 4 ESS network management display system provides the network manager with the data needed for the surveillance and manual control of the network. The system has access to a large No. 4 ESS-maintained data base consisting of both traffic and plant measurements as well as measurements specially collected for network management. The latter includes 5-minute per code completion data, 5-minute trunk subgroup performance measurements, and measurements of the number of calls affected by each automatic and manual control action taken by the No. 4 ESS. The display system also has access to control and equipment status data which are updated on a per-event basis, as well as to information that shows the outgoing trunk subgroup choices for each destination code.

Since the data base is large, the surveillance function is done in two steps. The first step involves the automatic detection of potential network management problems and the alerting of the network manager to these problems through visual and audible indicators. The second step consists of problem investigation through data analysis in which the network manager employs the capabilities of an interactive CRT display system.

##### **4.1 Surveillance arrangements**

Network management problems are diagnosed by the No. 4 ESS through "exception calculations" and recognition of critical events. Exception calculations consist of a set of calculations that the system performs typically every 5 or 15 minutes on a large number of machine and trunk subgroup performance measurements, which are compared against preset thresholds. This machine processing of large volumes of data permits all potential problems (called "exceptions") to be compressed and displayed on about 160 ON-OFF indicators and about 20 numerics which make up the exception panel. The onset of an exception indicator shows that an associated exception calculation exceeded its threshold or that a critical event has taken place.

The onset of an indication on the exception panel will direct the network manager to one of ten CRT pages which provide an overview of the problem. These overview pages are supported by about 50 detailed pages which are organized so that the network manager searches in a pyramidal fashion from a gross indication of a network problem to a detailed description of the problem.

The CRT system is supported by software programs which allow data to be retrieved, processed, and formatted for display. No longer will the network manager have to search through raw data to investigate a problem. The method of presenting data and the page layout has been carefully optimized from a user-oriented point of view. All CRT pages are interactive and consist of a fixed background and data windows, as described in Section 4.3.

Manual controls are activated and deactivated through a standard set of ten interactive CRT displays called "CRT control pages." Control pages offer greater flexibility over hardware control consoles and provide the opportunity for an immediate feedback of data pertinent to the control action.

#### **4.2 Exception reporting**

As shown in Fig. 5, the exception panel contains functional groupings of lamps. The exceptions represented on the panel are indicating either the occurrence of events or the analysis of measurement data that reflect key aspects of system performance. For statistical reliability, classes of performance data are analyzed at different rates, namely 30 seconds, 5 minutes, and 15 minutes.

Normally, the only lighted indications are the numerics in the upper left of the panel which are indicating the volume, measured in thousands, of incoming and outgoing traffic in the past 15 minutes. As exceptions occur, green/red/amber/white lamps are illuminated. Green lamps indicate that manual controls or manually initiated overrides are in effect. These lamps serve not only to remind the network manager who invoked the action but also to inform the other center in a local/remote arrangement that manual control actions are taking place. Exceptions related to system performance are indicated by red/amber/white lamps. The color of the lamp is keyed to the significance of the exception. To draw attention to an important change on the exception panel, a spurt audible alarm accompanies the lighting of a lamp.

The majority of exceptions are determined by checking various aspects of traffic-handling performance against a set of acceptable criteria or thresholds. Since exceptions are meant to alert the network manager to potential problems which may warrant manual intervention, the selection of thresholds is an ongoing process. For instance, network performance differs for an average business day and a peak day, such as Christmas. Consequently, CRT pages were designed to allow threshold values to be easily changed.

An intrinsic part of the exception-reporting system is an exception printer. This printer provides a chronological record of the key machine status exceptions, network performance exceptions, and manual and automatic control actions. These printouts provide details on the ex-



| NETWORK MANAGEMENT DISPLAY   |  | NETWORK PERFORMANCE   |  |   |  |   |  |   |  |   |  |
|--|--|---|--|---|--|---|--|---|--|---|--|
| <b>TRAFFIC PATTERNS</b><br>INCOMING      OUTGOING<br><input type="checkbox"/> 000 <input type="checkbox"/> 000<br><input type="checkbox"/> FOR REG <input type="checkbox"/><br><input type="checkbox"/> HOME REG <input type="checkbox"/><br><input type="checkbox"/> SUBTEND <input type="checkbox"/><br><input type="checkbox"/> HOME TCA <input type="checkbox"/> |  | <b>COMPLETION DATA</b><br>IMA      INEFFECTIVES<br><input type="checkbox"/> NC <input type="checkbox"/> OUT FAIL <input type="checkbox"/> NPA IMA<br><input type="checkbox"/> TIMEOUTS <input type="checkbox"/> INC FAIL <input type="checkbox"/> NPA INA<br><input type="checkbox"/> INT 0 <input type="checkbox"/> PSTO/FSA <input type="checkbox"/> HOME IMA<br><input type="checkbox"/> CAMA <input type="checkbox"/> VAC/INX <input type="checkbox"/> HOME INA<br><input type="checkbox"/> CONTROL <input type="checkbox"/> FNPA NXX<br><input type="checkbox"/> SPECIAL |  |   | <b>TRUNK SUBGROUP PERFORMANCE</b><br>NC      OFL      ACH      CCH      NSO      OOS<br><input type="checkbox"/> 00 <input type="checkbox"/> 00 <input type="checkbox"/> 00 <input type="checkbox"/> 00 <input type="checkbox"/> 00<br><input type="checkbox"/> 00 <input type="checkbox"/> 00 <input type="checkbox"/> 00 <input type="checkbox"/> 00 <input type="checkbox"/> 00<br><input type="checkbox"/> 00 <input type="checkbox"/> 00 <input type="checkbox"/> 00 <input type="checkbox"/> 00 <input type="checkbox"/> 00<br><input type="checkbox"/> 00 <input type="checkbox"/> 00 <input type="checkbox"/> 00 <input type="checkbox"/> 00 <input type="checkbox"/> 00 |   |  |   |  |   |  |
| <b>NETWORK CONTROLS</b>  |  |   |  |   |  |   |  |   |  |   |  |
| <b>MANUAL CONTROLS</b><br><input type="checkbox"/> CODE BLOCK<br><input type="checkbox"/> ROUTE CONTROL<br><input checked="" type="checkbox"/> REDROUTE CONTROL<br><input type="checkbox"/> MANUAL CHANGE  |  | <b>HARD TO REACH LISTS</b><br>TRANSMIT      CONTROL      CHANGES<br><input type="checkbox"/> AUTO <input type="checkbox"/> AUTO <input type="checkbox"/> INTERNAL ADD<br><input type="checkbox"/> MANUAL <input type="checkbox"/> MANUAL <input type="checkbox"/> INTERNAL DELETE<br><input type="checkbox"/> OVERRIDE <input type="checkbox"/> REMOTE <input type="checkbox"/> REMOTE ADD<br><input type="checkbox"/> LIST OVERFLOW <input type="checkbox"/> OVERRIDE <input type="checkbox"/> REMOTE DELETE   |  |   | <b>TRUNK SUBGROUP CONTROLS</b><br>STR      AUTO DER      DOC RECEIVED<br><input type="checkbox"/> ACTIVE <input type="checkbox"/> ACTIVE <input type="checkbox"/> MC 3<br><input type="checkbox"/> OVERRIDE <input type="checkbox"/> ROB NA <input type="checkbox"/> MC 2<br><input type="checkbox"/> TURN OFF SENT <input checked="" type="checkbox"/> MC 1<br><input type="checkbox"/> TURN OFF RCVD <input type="checkbox"/> OVERRIDE<br><input type="checkbox"/> TURN OFF RCVD <input type="checkbox"/> OVERRIDE   |   |  |   |  |   |  |
| <b>MACHINE STATUS</b>  |  |   |  |   |  |   |  |   |  |   |  |
| <b>EOPT STATUS</b><br><input type="checkbox"/> CRITICAL<br><input type="checkbox"/> MAJOR ALARM<br><input type="checkbox"/> INTERRUPTS<br><input type="checkbox"/> CARRIER ALARM<br><input type="checkbox"/> CAMA<br><input type="checkbox"/> TELEMETRY  |  | <b>CAPACITY</b><br>DFL      OOS<br><input type="checkbox"/> MF TRMTR <input type="checkbox"/><br><input type="checkbox"/> MF INC <input type="checkbox"/><br><input type="checkbox"/> CAMA OPR <input type="checkbox"/><br><input type="checkbox"/> DP/CCIS <input type="checkbox"/><br><input type="checkbox"/> DISK <input type="checkbox"/><br><input type="checkbox"/> CR   |  | <b>TEST CALLS</b><br>SIZE TIME      ADDRESS TIME<br><input type="checkbox"/> MF <input type="checkbox"/> RESPONSE TIME<br><input type="checkbox"/> DP <input type="checkbox"/> INCOMPLETES<br><input type="checkbox"/> CCIS |  | <b>OPERATIONS</b><br>PHASE      OVERLOAD<br><input type="checkbox"/> P4 <input type="checkbox"/> REAL TIME<br><input type="checkbox"/> P3 <input type="checkbox"/> MF INCOMING<br><input type="checkbox"/> P2 <input type="checkbox"/> DP INCOMING<br><input type="checkbox"/> P1 <input type="checkbox"/> CCIS INCOMING<br><input type="checkbox"/> INTERNAL |  | <b>INTERNAL CONT</b><br><input type="checkbox"/> DENY ACCESS<br><input type="checkbox"/> LIMIT ACCESS<br><input type="checkbox"/> LIMIT TASKS<br><input type="checkbox"/> DEFER M TCE<br><input type="checkbox"/> ALARM OFF |  | <b>DOC SENT</b><br>MC 1      MC 2      MC 3<br><input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> MF<br><input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> DP<br><input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> CCIS<br><input type="checkbox"/> MAN <input type="checkbox"/> DOC<br><input type="checkbox"/> OVRD <input type="checkbox"/> ACK |  |

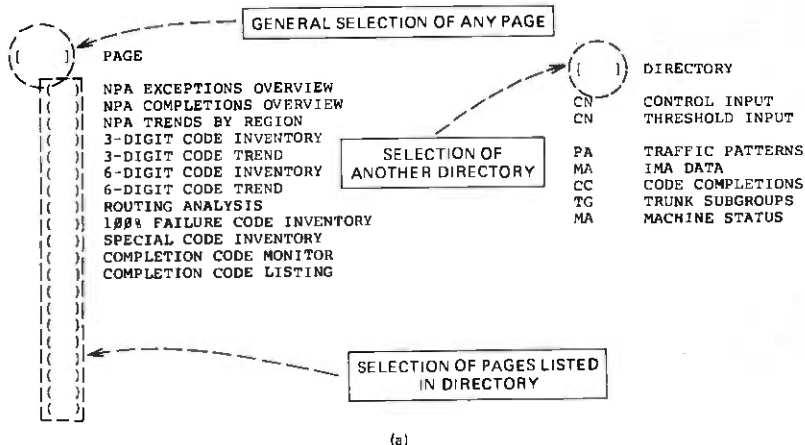
Fig. 5—Network management exception display panel.

ception analysis which resulted in a lamp indication. For instance, the lighting of an OFL lamp in the TRUNK SUBGROUP PERFORMANCE section of the panel would be accompanied by a printout identifying the trunk subgroups which exceeded the "percent overflow" threshold. In a similar manner, other printouts detail the exceptions for the network manager. Besides supplying information for the real-time analysis of problems by network managers, the exception printouts provide a historical record of exception conditions and control responses for post-problem critiques.

### 4.3. CRT display system

The CRT display system provides a flexible capability for investigating problems that are brought to the network manager's attention by the exception panel. Since effective problem solving is based, to a large extent, on empirical knowledge, the design is aimed at providing a flexible set of general-purpose capabilities for the analysis, presentation, and interactive querying of system data rather than a set of generically imbedded investigative sequences. The specified displays of information which are presented on a CRT screen are determined by specifications which are stored in the No. 4 ESS as sets of data that can be interpreted by the generic software system. These page specifications can be readily changed to accommodate modifications to problem-solving approaches based upon experience or upon changing characteristics of the network.

The displays presented on the CRT screen can be categorized into three



CC6 6-DIGIT CODE INVENTORY DATA TIME 12:05 SENT AT 12:07:32

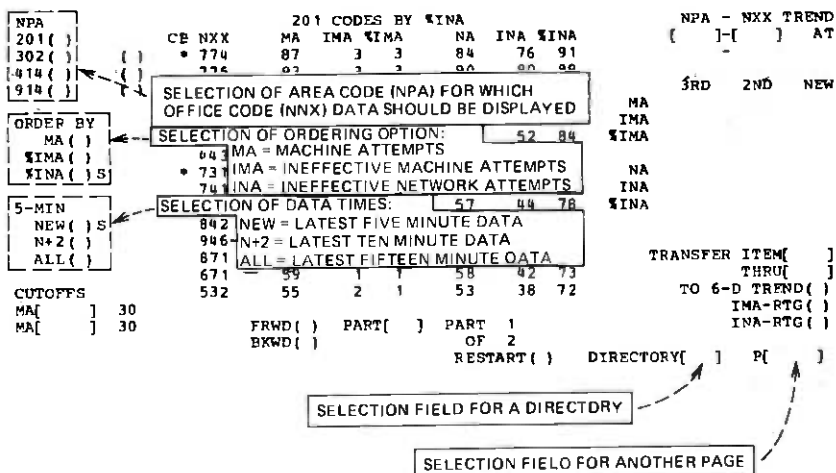


Fig. 6—(a) Example of a directory page. (b) Example of a display page.

general types: directory pages, display pages, and control pages. The directory pages provide a listing of the control and display pages that are contained within the system. The display pages provide processed system information on the CRT screen while the control pages allow the selection of the controls discussed in Sections 3.2 and 3.3.

CRT terminals are dedicated to this display function and the user operates in a "closed" system in the sense that the only valid input requests are either for another page or an interaction on the present page. The user makes selections of displays using the mobility capabilities

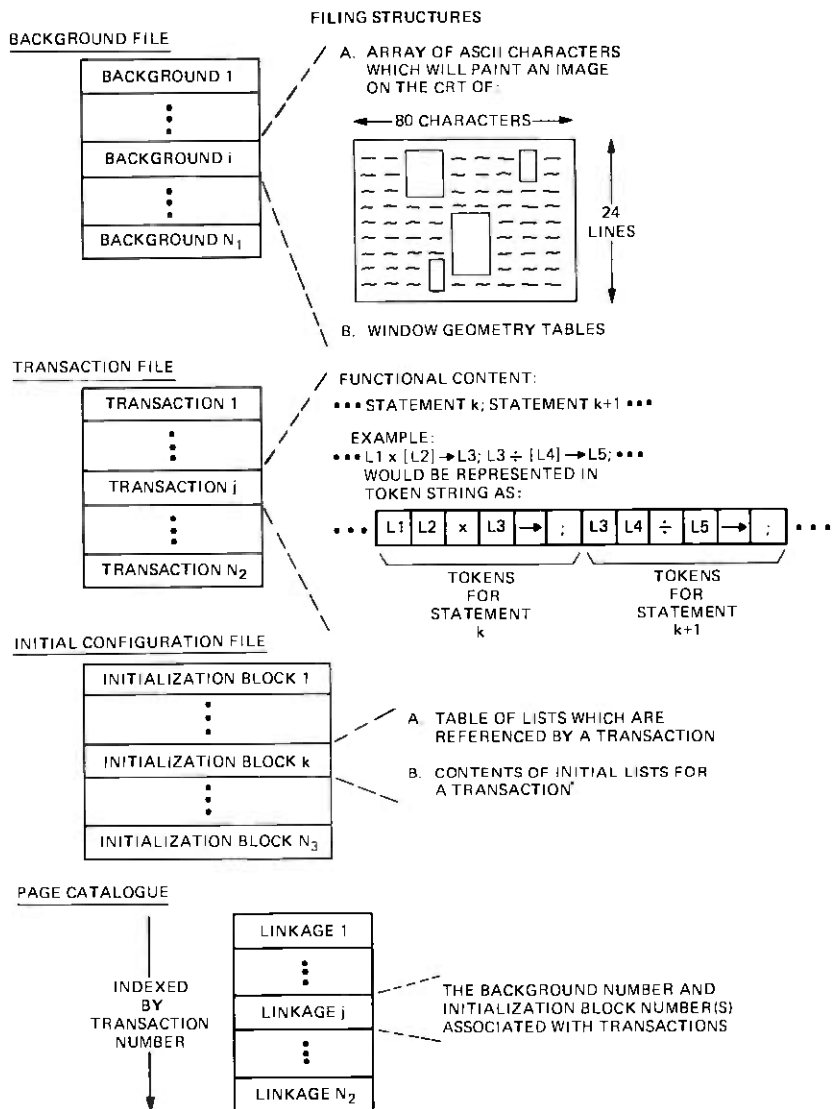


Fig. 7—Page data.

shown in Fig. 6. On a directory page, the user can select a page listed in the directory by placing a check mark next to the desired page and transmitting the request into the system or by specifying the page directly, e.g., CC6. On any nondirectory page the user can make a direct selection of another display/control page or a directory page. Whenever the system cannot interpret a request that has been transmitted from the CRT, the first directory page is displayed.

Display and control pages are generally designed to allow user interactions in order to realize a simpler and more orderly investigation of system data. As shown in Fig. 6, the user can request the same functional type of system data to be analyzed in several ways.

#### **4.3.1 Page data**

The information describing a page is contained in a disk filing structure as shown in Fig. 7. The specification for a page has three basic constituents: background information, a set of processing instructions called a transaction, and initialization information. In addition, the filing structure for pages contains a catalog which provides the basic linkage information to locate the page data for a user request.

The background data contain a description of the display as it is to appear on the face of the CRT. The ASCII image of the display background is stored along with window geometry tables. This background information will be displayed on the CRT screen as protected characters while the window areas will be in the unprotected mode. (Protected characters on the CRT screen cannot be overwritten by keyboard operations at the terminal). Two window geometry tables are provided. The first table provides the data for positioning the cursor of the Model 40 teletypewriter at window positions when displaying system data in window areas. The second table describes the sequence of window elements which will be received on a transmission from the Model 40 teletypewriter.

A transaction describes the processing actions associated with a CRT display. The description for the actions is contained in encoded statements of list-processing instructions which will be interpreted by the display system software for execution. The encoded form of a statement has each element represented by a byte called a token. The set of instructions in a transaction compose a program for retrieving system data, processing this data, and controlling the sequencing of actions to provide interactive responses. The instruction set available for encoding into the transaction includes arithmetic, list manipulation, conditional branching, and data retrieval operators. For example, the processing statement,  $L1 X[L2] \rightarrow L3$ , will cause an element-by-element multiplication of the contents of List 1 (L1) and List 2 (L2) and the results stored List 3 (L3).

At the start of execution for each transaction, the dynamic storage area for list information is initialized.

#### **4.3.2 Display processing**

In order to provide an understanding of the processing associated with a CRT display, a typical sequence for display processing is presented.

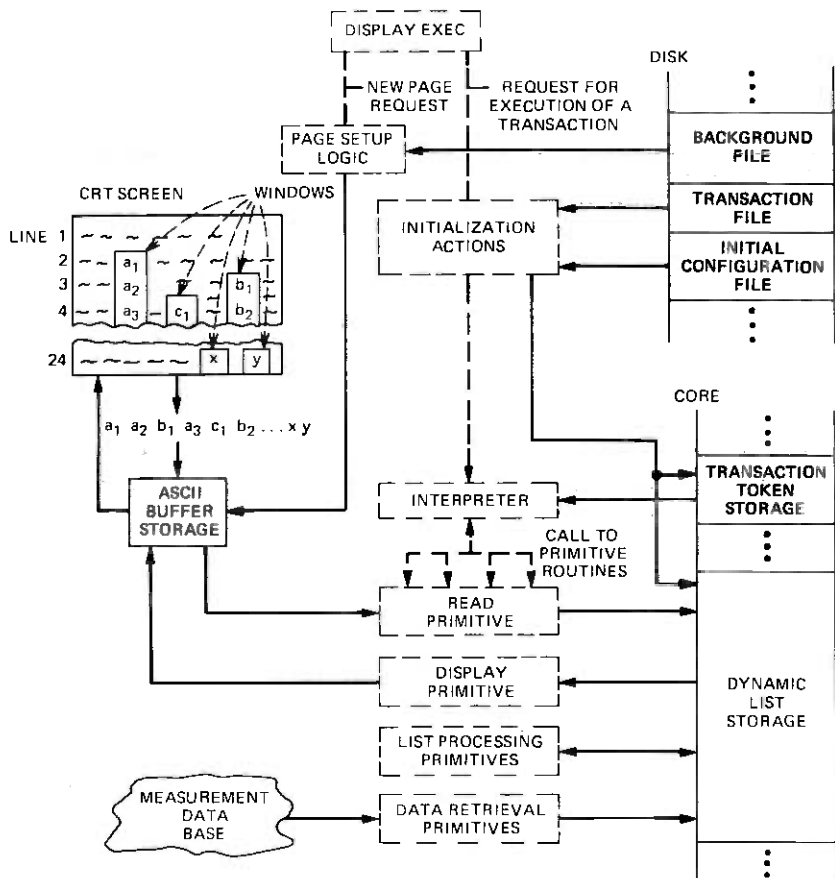


Fig. 8—CRT page processing.

The starting point for this sequence is a transmission from the CRT device. As shown in Fig. 8, the information in the unprotected window areas is transmitted as a serial string of ASCII characters resulting from a left-to-right, top-to-bottom scan of the unprotected areas by the device. These transmitted characters are placed into a dedicated buffer by the I/O system software and the executive function of the display system software receives an entry at the end of the transmission.

On receiving the stimulus from the I/O system software, the display system software checks if another page has been requested by consulting the last two strings of ASCII characters (x and y in Fig. 8). If a new page has been requested, the CRT background for the new page is fetched from filing structure on disk and placed into the ASCII buffer for transmission to the CRT by I/O system software.

After transmission of the background information to the CRT has been

completed, the execution of the transaction is instituted through the following actions:

- (i) Retrieve the initialization data from disk and establish the starting configuration for list storage.
- (ii) Retrieve the transaction information from disk and set up the token pointer to the start of the transaction data.
- (iii) Pass control to the interpreter function.

The interpreter starts a scan of the tokens in a transaction. The tokens are identified as either operators or arguments. The arguments are placed on a pushdown stack in preparation for encountering an operator. (Postfix notation is used in describing a processing statement and, therefore, all the arguments for an operation are encountered before the operator.) When an operator is encountered, the argument stack is "popped" to supply the arguments for the operation and an appropriate primitive routine is called by the interpreter. The primitive can be one of four basic types:

- (i) Data retrieval, which provides access to the measurement data stored in the No. 4 ESS and, as a function of their arguments, generates lists of various sizes which are dynamically allocated space in list storage.

- (ii) Processing, which provides the capability for data analysis and control of processing actions.

- (iii) Input, called Read, which constructs lists of window elements which have been transmitted from the CRT and reside in the ASCII buffer. This operation will also convert the ASCII information into an appropriate form for list storage. As shown in Fig. 8, the window information contained in the ASCII buffer is in a scrambled form and the unscrambling algorithm uses the window geometry tables mentioned above.

- (iv) Output, called Display, which fetches lists from the list storage area and, after doing an appropriate transformation into ASCII representation, places the information into the ASCII buffer for transmission to the CRT. Since windows on the CRT screen are filled one at a time, the output primitive incorporates cursor-positioning characters into the ASCII buffer along with the window information.

The end of a transaction is indicated by an encoded EXIT statement in the token string. When the interpreter encounters the EXIT statement, it returns control back to the executive function which then has the ASCII buffer transmitted to CRT. The executive function now will honor a request from another channel for the execution of a transaction.

When another transmission from the CRT is received, the I/O system software informs the executive function of the display software system.

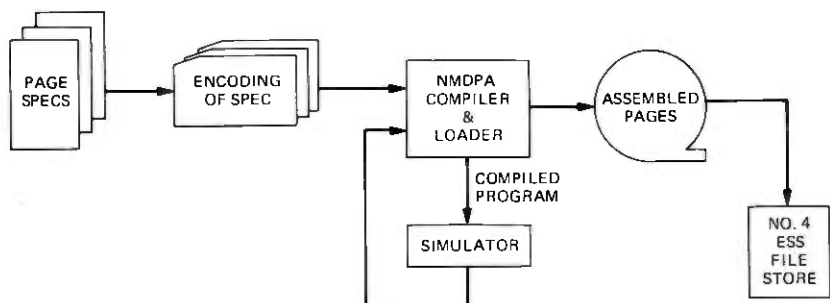


Fig. 9—Generation of CRT display pages.

If the transmitted data do not contain a request for a new page, the execution of the transaction is reestablished. In this case, the background is not transmitted to the CRT and the interpreter starts its scan of the tokens in the transaction at a point programmed to handle the interactions with the user.

#### 4.3.3 Generation of page specifications

As shown in Fig. 9, the information for CRT displays that resides in the No. 4 ESS memory is generated by means of a network management display page assembler, NMDPA. This program, which is executed in a general-purpose computation center, operates on an input specification of the following form:

(i) Background text and window geometry for a CRT page is described by an 80- by 24-character array (similar to its presentation on the 80- by 24-character screen of the CRT).

(ii) The transaction is described by high-level language statements.

(iii) The initial lists for a transaction are described by a set of text strings.

The NMDPA program performs extensive format and syntax checks on this input data and provides an assembled version of the input information. A collection of assembled pages can be provided as input for a loader run which produces a tape containing the image of the filing structure discussed above. The display page specifications on this tape can be read into the No. 4 ESS memory by the standard tape input facilities of the processor.

## V. MEASUREMENT DATA FOR MACHINE AND NETWORK ADMINISTRATION

The administration of a No. 4 ESS and the long-distance telephone network connecting it to other toll offices depends heavily on the mea-

surement data made available to the operating company personnel by the network management and traffic and plant measurement systems. These systems establish a data base of over 200 k call-store and file-store words and provide a retrieval system for the timely reporting of this data.

### **5.1 Groups of measurement data**

Conceptually, the entire data base can be divided into four broad categories: engineering data, machine performance data, network performance and control data, and division of revenue data.

#### **5.1.1 Engineering data**

Peg count, usage, and overflow measurements are provided on each of the facilities that must be engineered for the No. 4 ESS office. Every message trunk subgroup has peg counters which register incoming seizures, outgoing seizures, and overflows, and an occupancy counter to provide a usage measurement representing both service and maintenance usage.

Each of the sets of memory registers in the No. 4 ESS, such as Call Registers (CRs) and Output Message Registers (OMRs), and many of the software queues, such as the MF transmitter queue, have peg count and usage measurements generated for them. For those software facilities for which attempts can exceed available resources, overflow peg counts are provided. Facilities that can be idled by call abandonment have associated abandon peg counts.

#### **5.1.2 Machine performance data**

Measurements are available for the machine administrator and for the plant manager that detail the ineffective attempts in the office—both equipment-related, such as permanent signal time-outs, partial dial time-outs, and partial dial abandons, and traffic-related, such as no circuits and vacant codes. Measurements of the occurrences and duration of phases and interrupts and of the maintenance usage for both processor and peripheral equipment serve as indicators of the quality of service provided to customers.

#### **5.1.3 Network performance and control data**

Peg count data is accumulated for each Numbering Plan Area (NPA) code and for each office code in the home NPA and in a maximum of six foreign NPA's that are specified by the network manager. The peg counts reflect the number of calls that failed to be switched through the office, the number of calls that were successfully forwarded out of the office,



and of the latter, the number of calls that failed to receive answer. As discussed previously, these data provide the information for determining codes which are hard to reach.

Counts of calls that are affected by the network management automatic and manual controls on a trunk subgroup and maintained on a trunk subgroup basis. This allows for the monitoring and evaluation of these control actions as well as providing data which can be used to adjust the normal engineering data for trunk subgroups.

#### **5.1.4 Traffic separations data**

The "from-to" relationships of traffic flowing through the No. 4 ESS can be analyzed with the traffic separations data. Each incoming trunk subgroup in the office is assigned to 1 of 32 incoming categories and each destination code is assigned to 1 of 64 outgoing categories. For each call switched through the No. 4 ESS, peg count and usage measurements are registered in a cell of the 32 by 64 matrix of traffic separations data. A primary use of this data is to develop the separations factors which are used each month to divide the interstate revenues based on the plant investment attributable to interstate usage.

#### **5.2 Organization of the data base**

Both 5-minute and 15-minute data are made available in the network management and traffic and plant measurements data bases.

The 5-minute data base is generated primarily for the use of the network manager. Every 5 minutes the data are collected and stored on file store in one of four 5-minute blocks so that the current quarter hour's data is always available for reference.

The 15-minute measurements encompass over 8000 counters required in all offices plus several thousand counters which are dependent on the office size. Associated with each measurement is a minimum of two and in several cases three distinct areas of memory: a required call store counter, an accumulating register in call store or file store, and four required file store quarter-hour holding registers. Each call store counter contains either a peg count, which is incremented each time an event occurs, or an occupancy count, which is incremented each time a facility is seized and decremented when it is released. Accumulating registers contain cumulative totals of occupancy counts over the 15-minute interval.

#### **5.3 Data collection and storage**

The starting point for the collection of data is the registering of event occurrences in call store counters by programs throughout the No. 4 ESS software system, as shown in Fig. 10. Occupancy counters are maintained

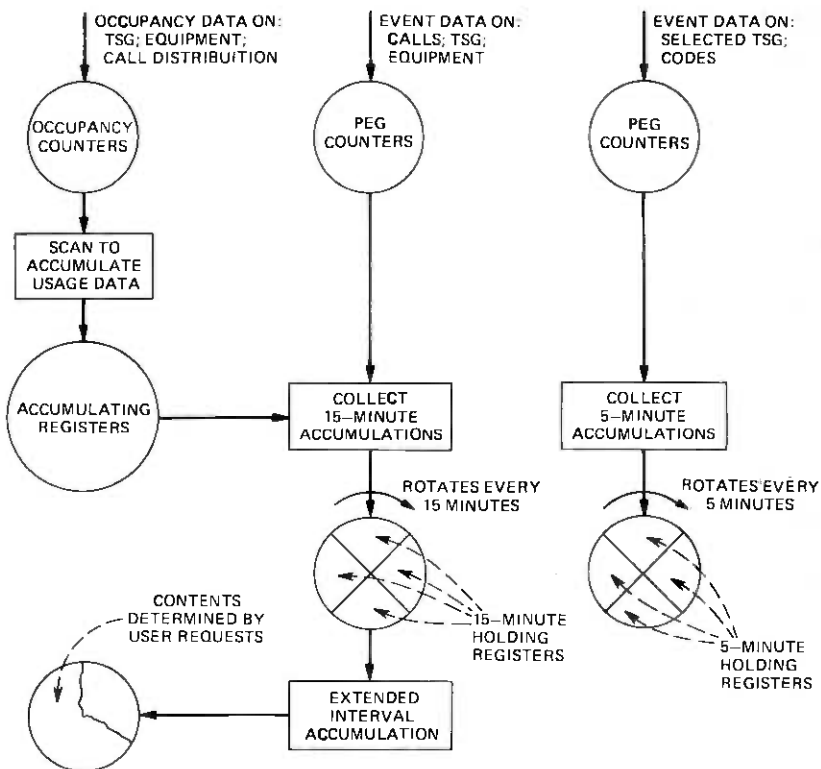


Fig. 10—Measurement data base.

to reflect the current number of facilities busy at any point in time. To obtain usage measurements, the occupancy counts are scanned at an interval that approximates the holding time of the facilities. Thus the occupancy counts for memory registers such as CRs or OMRs are scanned every 10 seconds. The value of the count at the time of the scan is added to an accumulating register which accumulates over a 15-minute interval. Counters for trunk subgroups are scanned every 180 seconds and similarly are stored in accumulating registers for 15 minutes.

Every 15 minutes the data in the counter blocks and the accumulating register blocks is collected and written into the holding registers on file store containing the oldest quarter-hour data. The counters and accumulating registers are recycled to zero to begin accumulating the data for the next quarter hour. The entire collection requires less than 1 minute to complete. In a similar fashion, 5-minute data, which only include peg counts, is transferred from call store to file store.

The primary source of measurement data is the last hour's peg counts and usage measurements stored in the four 15-minute holding register blocks on file store. Measurements from this data base can be accumu-

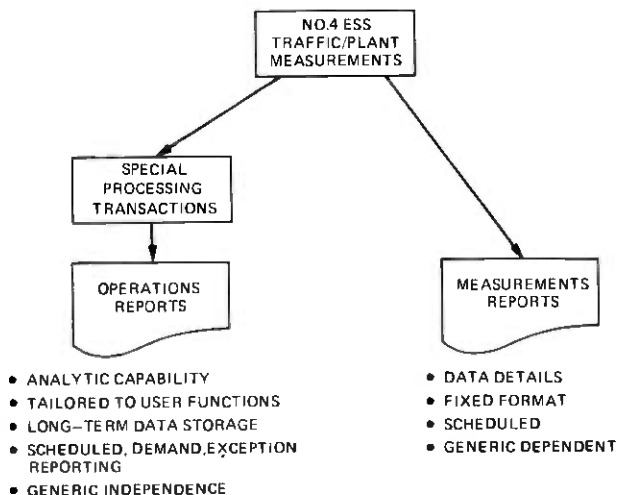


Fig. 11—Types of reports.

lated for extended intervals. The measurements to be summed into extended interval registers are specified by office personnel with teletypewriter input messages.

## VI. ADMINISTRATION AND ENGINEERING REPORTS

The No. 4 ESS provides on-site reports, containing processed data presented in various formats, to meet the needs of machine administration and network engineering. The quality and reliability of the traffic data in these reports benefits from utilizing the same overall system, with its equipment reliability and validity check features, that is used in the switching of calls. As shown in Fig. 11, two types of reports are generated: operations reports and measurement reports. Operations reports are formatted to meet specific user requirements concerning the performance of central office equipment. Measurement reports contain selected subsets of essentially "raw" data needed for monitoring trunk subgroups and performing other special studies.

### 6.1 Operations reports

The traffic engineering and administration reports generated by the No. 4 ESS list critical machine items (such as MF transmitters), related measurements, and other data in formats specifically tailored to user functions. Provisions are made for the printing of these reports on demand, on a scheduled basis, or automatically when manually preset thresholds are exceeded. When reports are demanded, they are immediately printed and contain data for specified past measurement inter-

vals, while scheduled reports are printed immediately after measurement intervals for which they were requested. Listings can be obtained on demand for all scheduling details.

Format flexibility is provided for all reports to allow the addition and deletion of measurements and the rearrangement of data on the report. This flexibility is independent of the No. 4 ESS generic and, therefore, allows changes to be made more frequently than generic updates.

### **6.1.1 Machine Service Report (MSR)**

The MSR is a comprehensive service report aimed at reflecting the quality of service being provided by the No. 4 ESS machine by focusing on the numbers and kinds of ineffective attempts and other service irregularities. The report provides a measure of the service level and a means of comparing that service level with other switching machines providing the same function. The report is partitioned to provide a brief, comprehensive presentation of the most sensitive service-level indicators in addition to providing details on the components of these indicators. Figure 12 shows the overview report which can be supplemented by reports on components, such as VACANT CODE. Hourly and daily MSR reports serve as the source of information for administering the office to meet service objectives. A monthly MSR serves to document whether or not an acceptable level of service is being achieved.

### **6.1.2 Machine Load and Service Summary (MLSS)**

The MLSS is used by the machine administrator and network engineer to monitor the traffic-load performance of the machine and interrelationships of traffic-sensitive equipment such as MF receivers and MF transmitters. Data, such as per-call holding times and maintenance usage on equipment, is provided to aid in identifying trends which could indicate potential problems. With this information, timely action can be taken to avoid service degradation. The MLSS serves as a common meeting ground for discussions between administration and engineering personnel.

The MLSS contains an hour's worth of data, beginning on the hour of half hour, and contains the data upon which the Load Distribution Report and Load Service Report (discussed below) are based. An MLSS can be scheduled for periodic printing and can also be printed on demand for any data-hour out of the preceding 24-hour period.

### **6.1.3 Load Distributing Report (LDR)**

The LDR is provided for use by the machine administrator and network engineer to determine the busy hour and distribution of load throughout a day for a category of traffic-sensitive equipment. Presently,

## MACHINE SERVICE REPORT

MSR

PART 1

REPORTING OFFICE: KSCY MO 09 04T DATE: 05/31/76 TIME: 0030 PAGE 1  
 OF 3 REPORT TYPE: HOURLY SCHEDULED REPORT  
 REPORT PERIOD: MAY 1976

|   | COUNT * | PERCENT |
|---|---------|---------|
| CONDENSED INEFFECTIVE MACHINE ATTEMPT (IMA) REPORT        |         |         |
|   | IS-FS   |         |
| TOTAL NO CIRCUIT (NC)                                     | 166     | 0.80    |
| NC-INTERTOLL (NC-IT)                                      | 166     | 0.80    |
| NC-TOLL COMPLETING (NC-TC)                                | 0       | 0.00    |
| NC-OTHER  | 0       | 0.00    |
| NC-NETWORK MANAGEMENT (NM) BLOCKED<br>CALLS               | 0       | 0.00    |
| NM BLOCKED AFTER AUTOMATIC OUT OF<br>CHAIN (AOOC) ROUTING | 0       | 0.00    |
| TOTAL VACANT CODE (VC)                                    | 0       | 0.00    |
| TOTAL REORDER (RO)  | 23      | 0.11    |
| EQUIPMENT CAUSED REORDER (REO)                            | 13      | 0.06    |
| NON-EQUIPMENT CAUSED REORDER (NRO)                        | 10      | 0.04    |
| TOTAL IMA   | 189     | 0.92    |
| TOTAL ADJUSTED IMA  | 189     | 0.92    |
| ANNOUNCEMENT GROUP OVERFLOW REPORT                        |         |         |
|   | IS-FS   |         |
| GROUP A OVERFLOW  | 0       | 0.00    |
| GROUP B OVERFLOW  | 0       | 0.00    |
| ABANDON CALL REPORT                                       |         |         |
|   | IS-FS   |         |
| TOTAL ABANDONS  | 2       | 0.00    |
| 05/31/76 00:33:07 #450                                    |         |         |

Fig. 12—Operations report output.

eight such categories have been identified. In general, the LDR contains data for 28 hourly intervals, overlapping on the hour and half hour, covering a span of 14½ hours. The busiest hour is flagged on the basis of a key measurement such as usage. An exception to the above is the LDR reflecting the real-time utilization of the processor which is based on 15-minute time intervals of data with no overlap and within a span of 14 hours. For this item, the busiest 15-minute interval is flagged.

Daily LDRs can be scheduled for selected traffic-sensitive equipment categories and are primarily used for studies of traffic on special days such as Mother's Day and Christmas. Five-day average (weekly) LDRs can also be scheduled for selected traffic-sensitive equipment categories. These reports employ the daily LDR format described above and assume a consistent span of contiguous hours throughout the week. Weekly LDRs are primarily used to determine busy hours for scheduling MLSSs and Load Service Reports.

#### **6.1.4 Load Service Report (LSR)**

The LSR is a specialized engineering report provided for traffic-sensitive central-office equipment. An LSR contains a yearly summary of busy-hour traffic data for a traffic-sensitive equipment category (similar to the LDR) in a form usable by the network engineer. For a defined busy hour, the LSR contains a descending listing of the 15 highest days within the past year, the average of the ten highest days, averages of each of the 3 highest months (busy season), an average of the 3 highest months (average busy season), and averages of each of the remaining 9 months.

Capability is provided for scheduling the printing of two sets of LSRs as a function of selected days of data. One set of LSRs summarizes only weekdays (Monday through Friday) and the other set considers only Sundays. For a set of LSR, capability is provided for the selective assignment of each traffic-sensitive equipment category to two different busy hours, e.g., a morning and afternoon busy hour. Provisions exist for the LSR to be based on data accumulated for the previous year with the capability of zeroing the historical data base when required. Capability also exists for manually changing the designated busy hours to accommodate any significant shifts in traffic concentrations.

Load Service Reports are also generated and printed based on peak intervals of traffic as opposed to a predetermined busy-hour basis, for the output message registers (OMRs) and the processor's real-time capacity. For each of these components, the peak interval of daily measurements is determined automatically and the associated data is processed.

#### **6.1.5 Exception reports**

Exception reporting is planned to alert the machine administrator in real time to individual abnormal conditions affecting service. Exception reporting occurs when manually preset thresholds are exceeded, and provides automatic monitoring of the performance of the switching machine during periods not covered by scheduled reports.

Machine status exception reports result from any of three conditions

that can cause traffic data to vary significantly from what is normally expected:

- (i) Equipment malfunctioning
- (ii) Inordinate variations in offered load from the network
- (iii) Data inaccuracies

Machine-status exception reports are based on measurement intervals used in the generation of central-office equipment reports. Exception reporting on the processor's real-time utilization is based on 15-minute measurement intervals, since this is the interval used for engineering purposes. Exception reporting on all other exception measurements is based on measurement intervals of 1 hour. Besides containing exception notification of "key" measurements used to detect abnormal conditions, machine-status exception reports contain "slave" measurements which comprise a predetermined grouping of related measurements to provide additional understanding regarding possible causes of exceptions.

Machine-status exceptions result in automatic flagging of exception measurements, the underlying data, and all other measurements based on the underlying data. For example, an exception report on holding time has directly related data of usage and peg count and all three measurements are flagged. Flagging of related measurements on the engineering reports is necessary since the network engineer does not monitor exception reports and should be aware of questionable data.

### **6.1.6 Reliability**

Provisions are available for the machine administrator to manually delete invalid data from the LSR historical data base. Questionable data cannot be automatically suppressed since human judgment is needed to verify that the data is unreasonable. A grace period is provided during which the machine administrator can delete invalid data before it becomes permanently imbedded in the data base.

Software and hardware are provided for purposes of data validity and reliability. Data base redundancy is provided for both the LSR and previous 24-hour historical data bases. Periodic audit checks of the data are made to ensure high reliability and data integrity. A tape backup system is provided for the LSR data base to give the machine administrator the capability of initiating an automatic overwrite of bad data. Should the tape backup fail, periodic hard copy outputs provide the network engineer sufficient information to manually construct data for engineering purposes. Data base protection mechanisms are provided to prevent the destruction of historical data by system users. The use of software "keys" and "policing of communications" are employed along with the use of a manual key to prevent storage wipe-out.

|                   |                       |               |     |     |     |     |     |     |     |
|-------------------|-----------------------|---------------|-----|-----|-----|-----|-----|-----|-----|
| NAME              | TYPE                  | PART OF DAY   |     |     |     |     |     |     |     |
| ( ) MPR1          | (+) HOURLY            | ( ) 0000-1100 |     |     |     |     |     |     |     |
| ( ) MPR2          | ( ) DAILY             | (+) 1200-2300 |     |     |     |     |     |     |     |
| (+) MLSS          | ( ) DAY TO HOUR       | ( ) 0030-1130 |     |     |     |     |     |     |     |
| ( ) MSR1          | ( ) MONTH TO DATE     | ( ) 1230-2330 |     |     |     |     |     |     |     |
| ( ) SEPSUM        |                       |               |     |     |     |     |     |     |     |
| ( ) MLSSDMD       | MLSSDMD DATA TIME [ ] |               |     |     |     |     |     |     |     |
| ( ) INIT MTD DATA |                       |               |     |     |     |     |     |     |     |
| (SEPSUM ONLY)     | ( ) INHIBIT           | TIME          | SUN | MON | TUE | WED | THU | FRI | SAT |
|                   |                       | 1200          | ( ) | (+) | ( ) | ( ) | ( ) | (+) | ( ) |
|                   |                       | 1300          | ( ) | (+) | (+) | (+) | (+) | (+) | ( ) |
|                   |                       | 1400          | ( ) | (+) | (+) | (+) | (+) | (+) | ( ) |
|                   |                       | 1500          | (+) | (+) | (+) | (+) | (+) | (+) | ( ) |
|                   |                       | 1600          | (+) | (+) | (+) | (+) | (+) | (+) | ( ) |
|                   |                       | 1700          | (+) | (+) | (+) | (+) | (+) | (+) | (+) |
|                   |                       | 1800          | (+) | (+) | (+) | (+) | (+) | (+) | (+) |
|                   |                       | 1900          | (+) | (+) | (+) | (+) | (+) | (+) | (+) |
|                   |                       | 2000          | (+) | (+) | (+) | (+) | (+) | (+) | (+) |
|                   |                       | 2100          | (+) | ( ) | ( ) | ( ) | ( ) | ( ) | (+) |
|                   |                       | 2200          | (+) | ( ) | ( ) | ( ) | ( ) | ( ) | (+) |
|                   |                       | 2300          | ( ) | ( ) | ( ) | ( ) | ( ) | ( ) | (+) |

|           |  |  |  |  |  |  |  |  |  |
|-----------|--|--|--|--|--|--|--|--|--|
| CHANNELS  |  |  |  |  |  |  |  |  |  |
| (+) TRF1  |  |  |  |  |  |  |  |  |  |
| (+) SREC1 |  |  |  |  |  |  |  |  |  |
| ( ) MTC   |  |  |  |  |  |  |  |  |  |
| ( ) TRF2  |  |  |  |  |  |  |  |  |  |
| ( ) NMPR1 |  |  |  |  |  |  |  |  |  |
| ( ) NMPR2 |  |  |  |  |  |  |  |  |  |
| ( ) TCA1  |  |  |  |  |  |  |  |  |  |
| ( ) TCA2  |  |  |  |  |  |  |  |  |  |
| ( ) SREC2 |  |  |  |  |  |  |  |  |  |

WED 01/19/77 RESTART( ) DIRECTORY[ ] P[ ]

REPORT BEING MODIFIED OR VERIFIED

I/O CHANNELS ON WHICH REPORT WILL BE PRINTED

TIME FOR PRODUCING THE REPORT

Fig. 13—Scheduling for operations report.

### 6.1.7 Report handling

The input of all related report messages regarding scheduling, demanding, editing, and setting of thresholds are performed by the machine administrator at the machine administration center (MAC). A report can be broadcast to as many as five teletypewriter channels. Should the printing device be inoperative, the reports are directed to backup printers to ensure data collection. Reports are printed in 8½ by 11 inch page increments on sprocket-fed fanfold paper to facilitate handling and storage.

### 6.1.8 Scheduling and generation

Operations reports can be scheduled for output to the various work centers in the No. 4 ESS. The scheduling and generation for these reports uses the capabilities of the network management display system discussed in Section 4.3.

Office personnel establish schedules for the printing of operations reports by interacting on CRT pages as illustrated in Fig. 13. After requesting this page on a display terminal, the user modifies the scheduling information through an interactive sequence. After the user specifies the report, the system responds with the scheduling information currently active. In Fig. 13, the system has responded with the MLSS schedule. The user can modify the scheduling for the MLSS by changing the channel and/or time indications and then activating this information.

Operations reports are generated with the processing capabilities



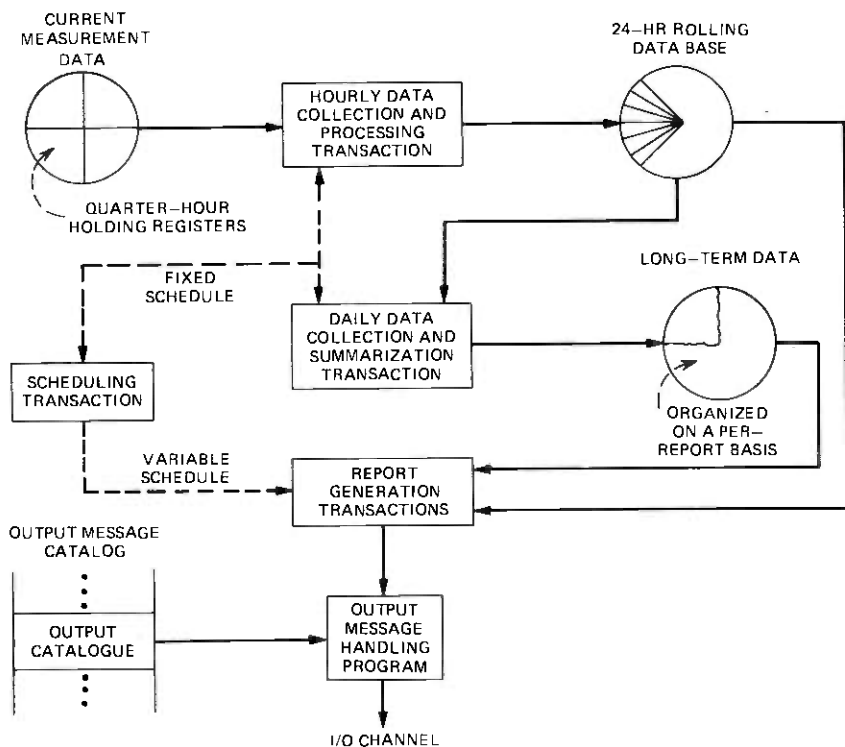


Fig. 14—Data flow for operations reports.

described in Section 4.3 for CRT displays. Transactions, as described in Section 4.3.1, are encoded to:

- (i) Establish and administer a data base tailored to the information needs of the operations reports.
- (ii) Process information for output on a report.
- (iii) Schedule the administration of the data base and the printing of reports.

The general flow of information for the reports is shown in Fig. 14. Transactions are routinely scheduled which generate a compacted data base to service the information needs of the operations reports. Hourly and daily reports on the performance of the No. 4 ESS access the processed data which is maintained in a rolling 24-hour data file. Data from this file is summarized to establish historical data files for monthly and yearly performance reports.

In producing hard-copy reports, the output lists of processed data, which are formed by a transaction, are merged with the background form in the processor before the ASCII characters are transmitted through an I/O channel to a teletypewriter.

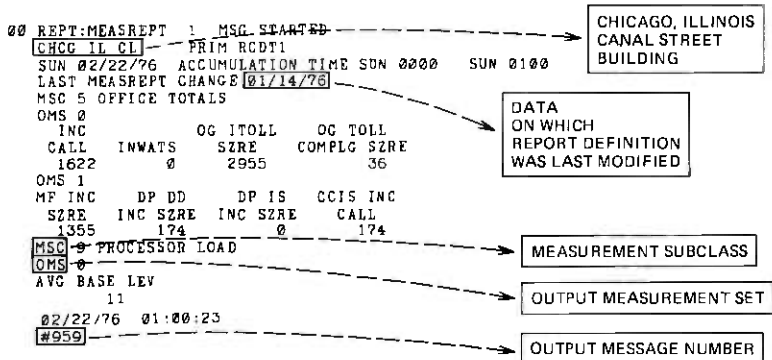


Fig. 15—Measurement report output.

## 6.2 Measurement reports

In addition to operations reports, the No. 4 ESS generates measurement reports that provide the machine administrator and other users with subsets of essentially "raw" data needed on a scheduled basis to monitor trunk subgroups, perform special studies, and monitor malfunctioning processor and peripheral equipment.

Twenty-four measurement reports can be produced by this system to meet user needs. The user defines a measurement report by specifying with input messages the report number and the primary input channel over which all subsequent changes to the report must be made. The accumulation interval of the report, which can range from 15 minutes to a full week, is input as well as one or more output times. The user can direct the output of each report to a maximum of five teletypewriter output channels as well as specify one output report to be written onto tape. The report output contains the specific sets of measurements requested on input messages. Measurement report 1 in Fig. 15 was defined using the RCDT1 channel as the primary channel (PRIM). It is a report produced by the Chicago 7 office in the Canal Street building in Chicago, Illinois on February 22, 1976, at 1 A.M. This is a 1-hour report whose definition was last changed on January 14, 1976. It contains several sets of related measurements which are identified in terms of measurement subclasses (MSC) and output measurement sets (OMS).

The input messages used to define a measurement report are processed by the measurement reporting system (Fig. 16). The definitions are stored on file store and remain in effect until changed by subsequent input messages. These definitions are backed up on magnetic tape. The definition of a measurement report reflects the need for additional memory if the report is required to output data accumulated over an interval longer than 1 hour. A sufficient number of extended interval

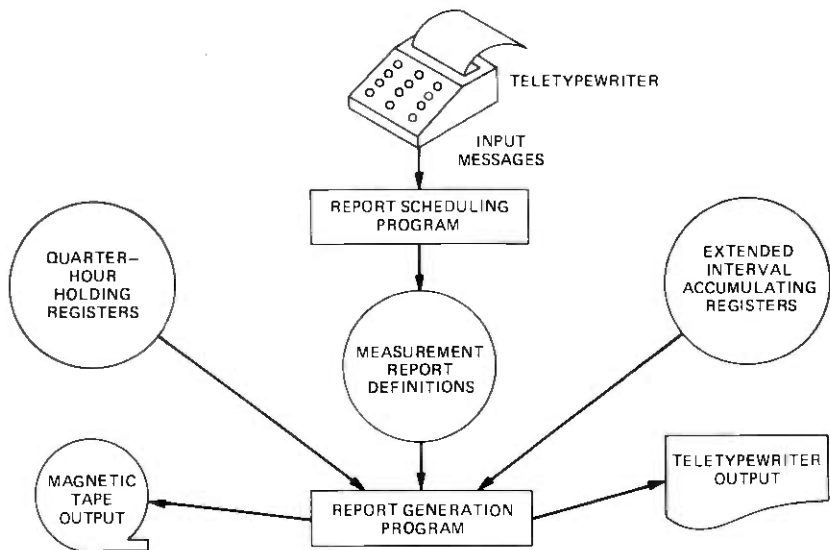


Fig. 16—Measurement reporting.

accumulating registers are allocated to the report to accumulate the measurements. If the report has an accumulation interval of an hour or less, it can be output entirely from the quarter-hour holding registers which are available to all reports.

Each quarter hour after the data is collected and stored on file store in one of four sets of quarter-hour holding registers, the extended interval accumulation program checks the measurement report definition on file store to determine if any reports are currently accumulating data for a period longer than 1 hour and adds an additional hour's data into the extended-interval accumulating registers for those reports. All accumulations are stored back on file store and then output processing begins.

The report generation program first checks the measurement report definitions to determine if any reports are scheduled for output. The reports are output in numerical order with one report being printed on all its assigned output channels before processing of the next report begins. If a report has extended-interval accumulating registers, the data are output entirely from the registers. If not, the data are gathered from one or more quarter-hour accumulating registers, added together if necessary, and output.

## VII. CONCLUSION

The service afforded by a No. 4 ESS relies upon effective engineering and administration in handling predictable traffic loads as well as un-

usual traffic conditions. In this paper, the features designed to provide these capabilities have been discussed.

For the unusual traffic conditions, automatic actions to control traffic overloads have been made more effective by controlling traffic parcels which have been determined to have low completion probabilities. To complement the automatic control system through manual control actions, a real-time surveillance system has been organized to provide problem detection by reporting exceptions and to allow problem investigation using interactive CRT displays of real-time performance data.

For engineering and administrative personnel the No. 4 ESS is generating reports on its switching performance. A comprehensive set of measurement data is gathered, screened, and processed to provide information in a directly usable form.

### VIII. ACKNOWLEDGMENTS

The authors wish to acknowledge the efforts numerous colleagues in the system planning and development of the traffic administration and network management features. Their contributions were equal to those of the authors. In particular we would like to acknowledge the contributions of M. J. DiCarlo-Cottone in the development of the network management features and R. I. Rosenthal for the system planning associated with the network management surveillance arrangements.

### REFERENCES

1. Y. Nakagome and H. Mori, "Flexible Routing in the Global Communication Network," Seventh International Teletraffic Cong., Stockholm, June 1973, Paper 426.
2. J. H. Weber, "Some Traffic Characteristics of Communications Networks and Automatic Alternate Routing," B.S.T.J., 41, No. 2 (March 1962), pp. 769-796.
3. P. J. Burke, "Automatic Overload Controls in a Circuit-Switched Communications Network," Proc. National Electronics Conf., December 1968, pp. 667-672.
4. W. B. Macurdy, "Network Management in the United States—Problems and Progress," Seventh International Teletraffic Cong., Stockholm, June 1973, Paper 621.
5. E. E. Lewis and E. E. Schwenyfefer, "Automatically Controlling Heavy Telephone Traffic," *Bell Lab. Rec.*, 44, No. 4 (April 1966), pp. 135-138.
6. L. A. Gimpelson, "Network Management: Design and Control of Communications Networks," *Elec. Commun.*, IT&T Corp., 49, No. 1 (1974), pp. 4-22.
7. C. A. Dahlbom, "Common Channel Signaling—a New Flexible Interoffice Signaling Technique," International Switching Symp. Record, 1972.
8. R. L. Franks and R. W. Rishel, "Optimum Network Call Carrying Capacity," B.S.T.J., 52, No. 7 (September 1973), pp. 1195-1214.

## **No. 4 ESS:**

# **Data/Trunk Administration and Maintenance**

By J. A. GIUNTA, S. F. HEATH III, J. T. RALEIGH,  
and M. T. SMITH, JR.

(Manuscript received July 8, 1976)

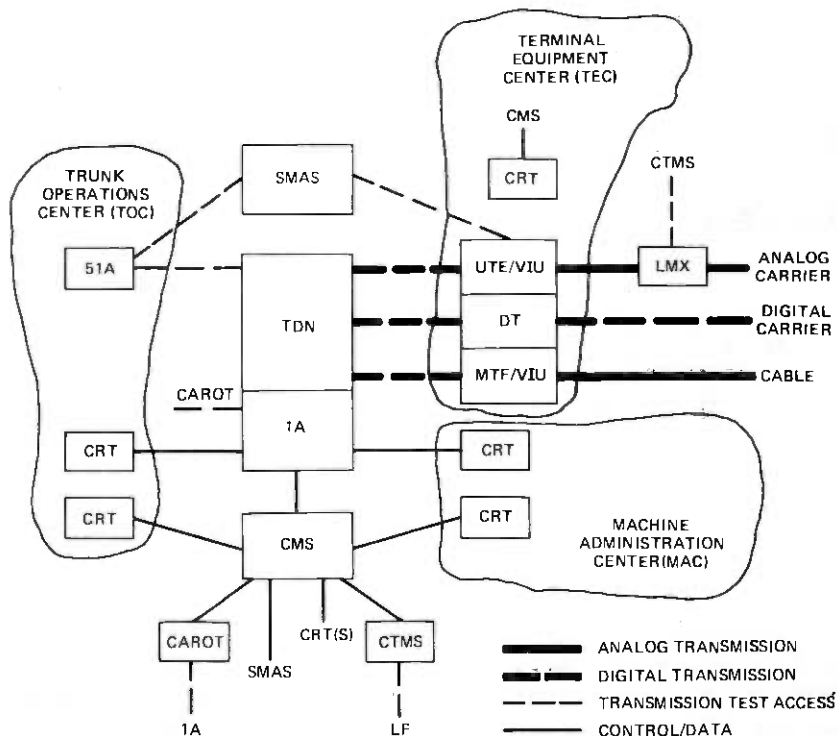
*A large data base supports both switching and maintenance functions in No. 4 ESS, a system which has the capability to switch a half million calls per hour on over 100,000 trunks. The data base is described here along with the subsystems that are used to support the installation and maintenance of both the data base and the trunks. The design and organization of these subsystems have been significantly impacted by the real-time needs of the call-processing system and the need to provide accurate, cost-effective ways to install and maintain the large number of trunks and their associated data bases.*

## **I. INTRODUCTION**

Two key elements in a No. 4 ESS are the trunks over which customer calls are placed and the data base which supports the call-handling function. Both the trunks and the data base require initial installation followed by ongoing maintenance and change processes. The data bases provided in No. 4 ESS and the subsystems that were developed to support the initial installation and maintenance of both the data bases and the trunks will be described.

The No. 4 ESS data/trunk administration and maintenance subsystems are depicted in Fig. 1. They are structured about the work centers, data bases, and operating features of the 1A Processor and two mini-computer systems, the Circuit Maintenance System 1A (CMS 1A) and the Centralized Automatic Reporting on Trunks (CAROT) system.

The 1A Processor contains the data base to perform the No. 4 ESS call-processing function and to access and control the trunks to perform maintenance functions. The CMS data base contains equipment-related, trunk-layout, and administrative information needed to perform the



- 1A - 1A PROCESSOR
- TDM - TIME DIVISION NETWORK
- CMS - CIRCUIT MAINTENANCE SYSTEM
- SMAS - SWITCHED MAINTENANCE ACCESS SYSTEM
- CAROT - CENTRALIZED AUTOMATIC REPORTING ON TRUNKS
- CTMS - CARRIER TRANSMISSION MAINTENANCE SYSTEM

- ANALOG TRANSMISSION
- DIGITAL TRANSMISSION
- TRANSMISSION TEST ACCESS
- CONTROL/DATA

- UTE - UNITIZED TERMINAL EQUIPMENT
- LMX - L-MULTIPLY EQUIPMENT
- DT - DIGROUP TERMINAL
- VIU - VOICEBAND INTERFACE UNIT
- MTF - METALLIC TERMINAL FRAME
- CRT - CATHODE-RAY TUBE/KEYBOARD

Fig. 1—Data/trunk administration and maintenance system.

circuit order and maintenance functions associated with trunks. CMS also contains a transient data base used to administer the work force involved in these activities. The CAROT system directs routine transmission testing of trunks and has a data base which identifies the trunks and associated test parameters.

Testing and test-access systems include the 51A test position, the Switched Maintenance Access System (SMAS), and the Carrier Terminal Maintenance System (CTMS). The 51A test position provides the ability to manually test trunks; access is via the time-division network, and/or via SMAS to an intermediate point on metallic or analog-carrier derived circuits. The CTMS system provides per-circuit measurements at various points in the multiplex equipment on broadband analog carrier systems.

Processes have been developed to support the initial installation,

growth, changes, and maintenance of the trunks and the associated call-processing and maintenance data bases. These processes are carried out in three work centers known as the Machine Administration Center (MAC), the Trunk Operations Center (TOC), and the Terminal Equipment Center (TEC). The MAC is responsible for the data base initialization and update functions as well as for the coordination of circuit order activity. The TOC is responsible for both the circuit order and maintenance testing of trunks and must ensure that both operational and transmission objectives are met. It also detects and then removes faulty trunks from service. TEC responsibilities include initial installation and repair of terminal equipment.

### **1.1 Objectives**

The primary objective is cost-effective administration and maintenance through the application of up-to-date but proven technology, capitalizing on the characteristics of the switching system. This is achieved through a uniform but flexible arrangement for maintenance and administration which was designed into the system and through increased productivity gained by expanding the use of automation. Particular examples are a reduction in paper records for both translation data and work administration, enhanced personnel interfaces, and automated trunk testing and data verification.

### **1.2 Switching system characteristics**

The degree to which the objectives for data/trunk administration and maintenance have been achieved is due in part to the characteristics of the switching system architecture. Since the network is nearly non-blocking, the need for load-balancing trunk assignments is eliminated. Conventional distribution frames and trunk circuits are eliminated by integrating switching, signaling, and transmission functions. This integration is in the form of Unitized Terminal Equipment (UTE) connectorized in its interface with the Signal Processor (SP) and the Voiceband Interface Frame (VIF), and the Digroup Terminal/Signal Processor 2 (DT/SP 2). Automated maintenance functions are incorporated into the common control equipment. Integration also encourages modularization of trunk assignments, which makes the storing and changing of data base information on a group rather than an individual circuit basis practical.

## **II. DATA BASES**

### **2.1 Overview**

The No. 4 ESS has, in addition to resident programs, a large resident data base that is used for switching calls and administering the circuit

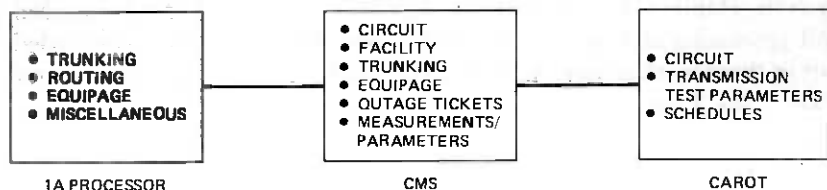


Fig. 2—No. 4 ESS data bases.

order and trunk maintenance processes. The data base used for switching calls is stored exclusively in the 1A Processor and consists of hundreds of thousands of words. The data base for administering the office is mainly stored in two minicomputer systems, CMS and CAROT. Figure 2 shows the major categories of data and their host processors. The following sections explain these data bases in more detail.

## 2.2 1A Processor resident data base

### 2.2.1 Data base structure

1A Processor memory primarily contains programs and data used for switching calls. This data includes such information as how calls are to be routed, what interoffice signaling format should be used, and on what equipment the interoffice trunk is terminated. The following presents where the data is stored, what are the capabilities of the data, how the data is stored, how the data is initialized, and what paper records are needed.

The data is stored in one of two ways. It can be stored in core store for fast access with a copy on file store for backup, or it can be stored only on file store. Both ways use electronically alterable memories. The actual choice between core and file store for the location of a particular type of data is engineered for each office depending upon its real-time needs. This flexibility could save up to one-third of the core requirements for office-dependent data in a 20k to 30k trunk office where the real time exists to access data on file store. The routing data provides for:

- (i) 3 through 9 digit translations
- (ii) 0 through 14 digit deletions
- (iii) 0 through 6 digit prefixing
- (iv) 1 through 14 direct and alternate routes plus up to seven automatic out-of-chain routes
- (v) 64 non-POTS domains in addition to subdivisions of the POTS domain
- (vi) 1 through 8 NPAs served on a 7-digit basis
- (vii) Screening based on a trunk subgroup characteristic that provides up to 16 different routing treatments per called number



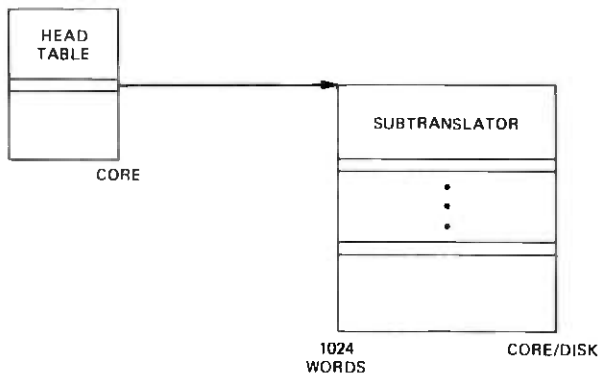


Fig. 3—1A Processor data structure.

(viii) Screening based on a trunk subgroup characteristic that provides either normal routing or a tone or announcement per called number.

The trunking data identify:

(i) Network location, scan point, and distributor point identification for non-CCIS trunks, band and label information for CCIS trunks, and a trunk name (in alphanumeric format) for up to 107,000 trunks

(ii) Circuit identification name (CIN) and trunk hunting data for up to 4000 trunk subgroups, each of which can have up to 1000 trunks and 38 class marks of data such as type of signaling (MF, DP, CCIS), echo suppressor requirements, etc.

The miscellaneous data contains:

(i) Floor location, equipage, vintage, and miscellaneous scan and distributor points for all hardware in an office

(ii) Parameters to allocate core store and disk to functions like call registers and input/output buffers

(iii) Space administration translators used to facilitate the changing assignments.

For most translators, a standard structure is used which is a connected, downward linked, multibranching list with two levels (Fig. 3). This structure provides efficient memory utilization, minimal real time for access (13 percent of per-call time), improved system integrity, ease in changing and growing structures, ease in initial generation, and direct generation of office records from the data using the verify system.

### 2.2.2 Data base initialization

The 1A Processor resident data is initially generated by an off-line system running on a general-purpose computer. This system, called the

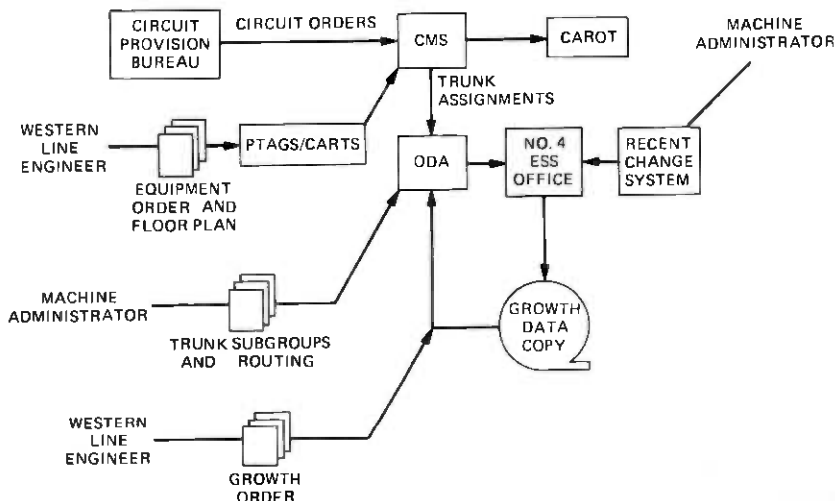


Fig. 4—Off-line generation of data base.

Office Data Assembler (ODA), was designed and implemented by Western Electric in PL1 and assembly language. The same system is used to reallocate storage for each growth interval (typically 1 or 2 years). The inputs to the system (Fig. 4) are the Western Electric specification of floor plan layouts and the telephone company specification of routing and trunking data. The floor plan information is combined with the equipment order for the office and processed by the Western Electric wiring assignment system [Provisional Trunk Assignment Generation System/Circuit Assignments Record Transfer System (PTAGS/CARTS)]. The output of this process is interframe wiring information for the office and a data base for CMS that describes the terminal equipment. This data base is loaded into the CMS and is updated with trunk assignment information as the assignments are made during the pre-cut installation and testing interval. Prior to cutover, the data base from CMS is output on magnetic tape and input to the Western Electric ODA program. Further assignment information, as well as routing information, is added. The output from ODA is the complete data base for the 1A Processor at the time the office is placed in service.

The input records were designed to eliminate unnecessary parameters by algorithmically generating related parameters and functionally grouping the input data. Less than 25 parameters need to be specified for No. 4 ESS. All the office-dependent data is generated by the ODA system. Actual trunk assignments are made manually and input to CMS on-site prior to cutover. This data is input to the ODA system via magnetic tape for conversion into the translator structures.

The routing data comes from two sources:

- (i) Manually completed records which are made by using the translation guide and traffic routing guide
- (ii) A mechanized vacant code file.

The output of the ODA is a magnetic tape containing the entire 1A Processor data base. This tape is loaded on-site into the 1A Processor. The ODA system also outputs wiring assignments. Once the data is stored in the 1A Processor, call-processing programs will access the data through a set of special-purpose routines which minimize the effect on per-call real time. Whenever possible, these routines self-detect data errors. The data base is modified, augmented, and verified through the recent change system.

### **2.3 Circuit Maintenance System data base**

The primary storage medium for the CMS system is disk. For a fully equipped No. 4 ESS office, over 300 million bytes of disk storage are provided in CMS. This data includes infrequently changed information such as office equipment layouts, circuit layout records, test records, and trunk group and per-trunk translation data which allows for communication with other systems. Dynamic data such as trouble tickets, circuit orders, and work lists are also maintained by the system. To build this large data base, CMS supports manual input, magnetic tape input, and data link interfaces. Whenever possible the data input is via data link. Most trouble reports are automatically input from the 1A Processor or CAROT when they detect circuit abnormalities. Circuit orders and circuit layout information is generally input through a data link interface to the telephone company circuit layout system.

The office equipment file, which describes the physical equipment in the No. 4 ESS office, is loaded from a magnetic tape generated by the Western Electric Circuit Assignments and Record Transfer System (CARTS), which is a part of the ODA system discussed previously. This tape is loaded in a new office and at the beginning of each growth interval (typically every 1 or 2 years).

Because the CMS data base is so large and overlaps data stored in other systems, CMS provides a data coordination function. It updates the CAROT per-circuit test file, stores test parameters associated with CTMS measurements, and provides precutover trunk assignments to the ODA, which uses this information for initializing the 1A Processor data base. The CMS returns in-effect reports on new circuit orders to the originating circuit layout system. The system also provides trouble ticket analysis data to the telephone company trunk service results plan which generates trouble index reports.

The CMS filing system is responsible for maintaining and securing the system data base. The filing system performs the allocation, accessing,

and deletion of files on the disks. Both fixed-length and variable-length files are supported. An integral part of the file system is data base protection software. This software provides for logging data base transactions, backing out updates which must be aborted, and provides various image copy and restore routines for both spare disks and magnetic tape. The filing system also provides absolute write features, directory and compaction routines, and file integrity and audit functions.

### **III. ADMINISTRATIVE AND MAINTENANCE SUBSYSTEMS**

The data/trunk administration and maintenance functions are accomplished through the interaction of several subsystems. Some reside in software in the 1A Processor, some are composed of a minicomputer and its software, while others are totally hardware entities. Each of the major subsystems will be described here.

#### **3.1 No. 4 ESS Recent Change subsystem**

The Recent Change (RC) system is a set of programs that interface between the craft at a *TELETYPE*<sup>®</sup> model 40 terminal and the data base in the 1A Processor. It allows office-dependent data which resides in the 1A Processor to be changed and verified on-site. Because of the volume of changes and the sensitivity of office performance to changes, the process could become time-consuming and error-prone if an efficient recent change capability with built-in consistency checks were not provided. In addition to reducing the administration costs, this system was designed to improve the data base integrity and the personnel interfaces relative to previous electronic switching systems. Major improvements in integrity were achieved by enhancing defensive software and by providing recovery strategies. All changes are interactively checked for range, format, and logical consistency. Only after a change passes these checks is the recent change message allowed to be buffered in the 1A Processor memory. This change procedure does not require knowledge of the translator structures; modification requires only the knowledge of the telephone function to be changed. All memory manipulation and space administration is done automatically.

Personnel interfaces were improved over previous systems by developing functionally oriented forms instead of translator-oriented forms, by eliminating the need for most of the paper records required in previous systems, and by developing an improved language which uses part of the common language identity [Circuit Identification Name (CIN)] of the trunk subgroup and alphanumeric abbreviations for data instead of arbitrary number encoding. The forms for the recent change system and the ODA system are similar in design. The user essentially needs to learn how to use only one type of form for specifying the original ODA data, retrieving existing data, or changing the existing data. These forms

RC: TRK; NEW: OPT (MANUAL) BUF: BTFN TOWN ST BL FBS NBS  
 ORNU ----- 1, TSG ----- 1 CHGO IL CL 57T 57T

TEC TESC ONR SEC  
 TOC -- 2, 1, 4, -- 1, ES ---,

| TAN  |      |      |     |     |     |
|------|------|------|-----|-----|-----|
| QTRK | FTFN | TSIF | SPC | LVL | FTS |
| 12   | 1    | 0    | 1   | 0   | 19  |
| ---  | ---  | ---  | --- | --- | --- |
| ---  | ---  | ---  | --- | --- | --- |
| ---  | ---  | ---  | --- | --- | --- |
| ---  | ---  | ---  | --- | --- | --- |
| ---  | ---  | ---  | --- | --- | --- |

REMARKS ----- "ADD 12 TRUNKS" ----- !

Fig. 5—Recent change message used to add 12 trunks.

are designed to be functionally self-sufficient. For example, only one form needs to be completed to add a trunk, a trunk subgroup, or an NXX code. Some of the forms allow multiple assignments on the same form (e.g., adding multiple trunks or NXX codes). Furthermore, the forms are designed to require a minimum of input. For example, on the add-trunk form (Fig. 5), only the network appearance needs to be specified and the RC system automatically assigns the universal scan points, distributor points, and miscellaneous points. This is possible because the ODA system captures and stores the connectorized cabling information for the office in the data base. The forms were also designed to be "English-language-like," using a subset of the common language name of the trunk subgroup that identifies the city, state, and building to which the trunk subgroup is connected. The forms use standard alphanumeric abbreviations, for example, MF, DP, and CCIS. Since these forms are functionally oriented and are instantaneously available to personnel via the CRT and hard copy, the large volume of paper records necessary for previous systems is no longer required. The need to manually update paper records, a process which is error-prone, is eliminated. Now the records reflect exactly what is stored in the 1A Processor. Furthermore, specialized retrieval routines for use of the administrator are provided to search the data base for information such as all the NXX codes that are routed to a particular routing pattern or all the trunk subgroups that have a particular screening class.

When the recent change messages are put into the recent change system, they actually go through three states, first a buffer state, then a test state, and finally an active state.

The recent change process is started when administrative personnel request via the CRT terminal a particular form needed to change some of the data. The RC system displays the form and the blanks are filled in by personnel using tab controls to advance from blank to blank. The form is then transmitted back to the system. The system performs extensive format, range, and logical consistency checks on the data. If an error is found on the form, it is redisplayed with an indication of where the error occurred. If the form passes all checks, it is stored in the buffer state on disk for later processing. This step serves as a "mail box" storage system and appropriate tools are provided to administer the messages in the buffer state.

Next, the messages can be put into the test state. This step changes the data so that the retrieval routines can display what the changes will be like when activated, but the normal call-processing routines cannot access the data. The output from the retrieval routines resembles the original change request. In this mode, personnel can see what the data presently look like and what the data will look like after the change. Further logical consistency checks are performed at this time by comparing the input data to the actual data in the system (e.g., is there a trunk subgroup in existence to which these trunks are being added). In addition, the trunking data which was changed can actually be used to make test calls and verify working circuits before the data is made available to normal call processing.

When the recent change message is to be activated, the changed data is released to normal call-processing routines. This is called the activate state.

In the event that a software trouble is introduced into the data base, No. 4 ESS has two recovery procedures. The first reinitializes the system from a tape that contains a snapshot of the data base at some previously known "safe point" in time. The second procedure, called rollback, automatically reinitializes the new data from a disk copy of the old data. The action is manually requested from the master control console along with an accompanying phase of software initialization. This procedure can selectively and quickly return the system to various points in time since the last snapshot tape.

If either procedure is used, another procedure, called rollforward, allows the system to reinsert selected recent change messages. This can be done because each message is automatically stored on a cassette tape when it is accepted into the test state. After reinitialization or rollback, this rollforward cassette tape is used to reinsert the recent change messages.

Circuit order work is the responsibility of the dial administrator. However, the initiation of a routing change is the responsibility of the routing supervisor who typically is responsible for all the routing changes in a region. When the routing supervisor determines that a change is needed, a remote terminal can be used to verify the existing data in a particular No. 4 ESS and to enter a recent change message into the buffer state; the activate date and coordinating data must also be entered so that the dial administrator can eventually activate the message.

The routing supervisor must administer groups of routes called routing data blocks. The CINs for all routes, digit deletes, digits to be prefixed, out-of-chain routing information, and INWATS data must be specified. Code routing treatment must also be specified as to: routing data blocks, announcement, tones, and screening. The code translations also resolve the numbering plan conflicts where the NPA and the office code NXX are identical. They are capable of handling toll connecting trunks which originate in any of eight NPAs. The eight NPAs include the home NPA and seven served NPAs. The No. 4 ESS allows a customer in the home NPA and each served NPA to dial seven digits for intra-NPA calls. The routing translators are also capable of handling 64 CCSA domains.

### **3.2 No. 4 ESS trunk maintenance subsystem**

The 1A Processor resident trunk-maintenance software subsystem supports all of the activities that actually involve control, configuration, or data collection associated with circuits that terminate on the No. 4 ESS switch. As described elsewhere in this paper, other subsystems such as the CMS, CAROT, and the 51A test position provide administrative and testing capabilities as a part of the total system plan. However, actual access to the trunks and test circuits that terminate on the No. 4 ESS switch (that is, the ability to set up network connections and read and write the transient data structures used to administer the circuits in conjunction with the call-processing function) is only possible with 1A Processor resident software.

Trunk maintenance software accesses trunks through network connections and through the scan and distribute function in the universal matrix of the signal processor. This is the same interface which is used by the call-processing programs. It accesses the test position and other test circuits as it does trunks except that in some cases there are additional scan and distribute functions via the miscellaneous matrix in the signal processor.

Three communication interfaces exist into the 1A Processor resident trunk maintenance software. The first is a pair of data links (one for each direction) between the 1A Processor and the CMS. This link is similar to the input/output channels used to interface the 1A Processor with *TELETYPE* model 40 terminals. Major differences are: all messages

require a positive response, retransmission is provided, and backup via protection-switched spare channels is provided.

The second communication interface is via the Remote Office Test Line (ROTL). This is a software-controlled hardware interface through which the CAROT system can, via analog signals, control trunk transmission testing in No. 4 ESS.

The third and most comprehensive interface is to conventional CRT and teletypewriter devices in the office that are supported by the 1A Processor input/output subsystem. The functions that are supported through this interface are a superset of those supported on the CMS data link.

Trunk maintenance software is a multiprocessing system that uses an engineered number of memory registers similar to the call registers used by call processing. This makes it possible to process concurrent requests and to use the system facilities designed into the system for call processing. A scheduler is provided as one task under the system executive program. This scheduler provides a single control point for the initiation of most tasks. Hence, it is possible to limit deferrable work when the system is operating in real-time overload.

### **3.3 Circuit Maintenance System**

CMS functions as the hub of trunk maintenance activities by interconnecting personnel and other automated systems. It consists primarily of dual processors and their associated peripheral equipment. As shown in Fig. 6, the system interfaces with:

- (i) Personnel in various administrative work centers of the No. 4 ESS office
- (ii) The wide variety of test and measurement systems that personnel use
- (iii) An extensive data base of equipment, test and trouble data, and information on circuit orders
- (iv) Data processing and operational systems.

Through CMS, work center personnel have access to the operational systems which perform various testing and control functions. Both people and automatic systems can access the CMS data base, either to update it or to obtain information. This arrangement allows craft personnel to perform many maintenance functions through a common interface so that they need not learn individual command languages and functions for many machines.

Features provided by the CMS to the various work centers utilize the *TELETYPE* model 40 CRT. The interface has been designed so that it may be used by inexperienced craft personnel with a minimum of



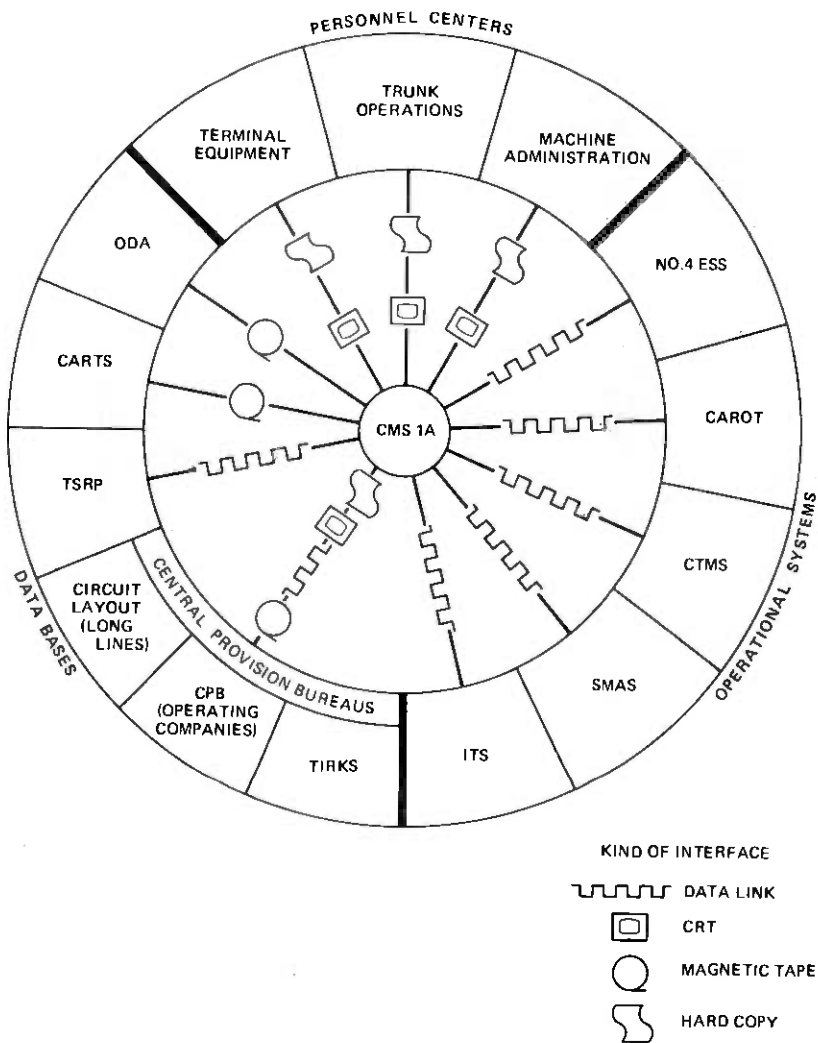


Fig. 6—CMS 1A interfaces.

training and yet it is not restrictive to more knowledgeable personnel. Illustrations demonstrating this interface will be given later.

Personnel interconnected by CMS work in three major administrative areas in a No. 4 ESS office: the Trunk Operations Center (TOC), the Terminal Equipment Center (TEC), and the Machine Administration Center (MAC). Personnel in the TOC (Fig. 7) interact with the CMS via a CRT on the 51A test position.

Ordinarily, the TEC has a stationary position with a CRT screen and a printer, as well as a number of mobile CRT units (Fig. 8) that furnish

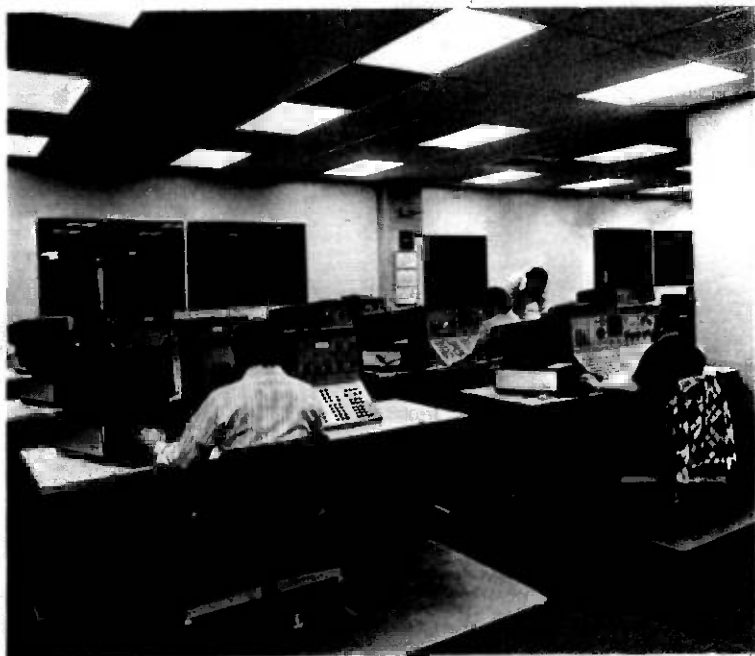


Fig. 7—Trunk Operations Center (TOC).

the same basic features as the 51A test position CRT, only at the frame location.

The MAC administers the circuit order process. Through CMS, personnel in this center obtain new circuit orders which are automatically correlated with existing orders and placed on work lists in the CMS data base. Personnel in the MAC (Fig. 9) can administer and control the completion of the orders via CMS.

Outage tickets and circuit orders are automatically distributed to craft personnel by the system. The algorithm used is based on the premise that maintenance responsibility for each trunk subgroup is assigned to a specific craftsperson. If a trouble occurs on any trunk in the subgroup, a trouble or outage ticket is made out and automatically assigned to the worklist of the responsible craftsperson.

The system takes care of the trouble ticket function which includes creation of the ticket, time-stamping pertinent events (such as turndown and turnup times, referral times), and times when automatic tests were run. The ticket also stores a log of comments made by the craftsperson, test results, trouble reported, and trouble-found data. All of these data



Fig. 8—Terminal Equipment Center (TEC).

are stored and used for the administrative analysis work that the system performs each evening.

CMS has personnel-machine interfaces in the various work centers in the office. When a tester has sectionalized a trouble and wants to refer it to the repair work center, the tester enters a command to the system which will automatically send the trouble ticket to the work center and put it on the responsible craftsperson's work list.

The CMS system provides personnel with access to its data base. This



Fig. 9—Machine Administration Center (MAC).

data base includes trunk, equipment, test and trouble data, and circuit order information.

Administrative features of the CMS system support the "control area" organization, which allows a local assignment of craft personnel to control areas. The control areas consist of a designated control position and up to nine other positions. These positions make up a unit whose activity can be managed by the control position and upon which administrative reports are grouped and distributed. Typical TOC control areas might be set up in an office with one control area responsible for all intertoll trunks and another responsible for all of the toll connecting trunks.

The system generates administrative reports for each control area.

These reports include: the morning report of all trunks that are out of service as of 8 o'clock that morning; a daily outage ticket analysis listing by trunk subgroup, type of trouble, and outage time for all of the troubles cleared the previous day; also various circuit order activity reports.

Another administrative feature is the display of the activity log of each position assigned to the control position. This display indicates the work load of a given craftsperson and how many troubles have already been cleared. If the work load is too large, the control position can redistribute some or all of the work to other positions. This feature can be used to set up a dispatcher mode of operation, to facilitate night transfers of work, and to redistribute work when a craftsperson is ill or on vacation.

The CMS is designed to continue operating with varying degrees of degradation after any single hardware failure and even after certain multiple failures. For this reason each CMS system has two processors which allow for a switchover to single-processor operation. Some data are stored in duplicate on disks so that a craftsperson can restore full capacity within a short time of a single disk failure. Reconfiguration capability in the event of multiplex equipment failure is built into the software. Input/output channels are automatically reassigned to keep all essential communication links active in the event of limited failures.

### **3.4 Test position 51A**

The 51A test position (Fig. 10) provides the capability to manually test trunks. It is a console at which the craftsperson can sit and conduct manual trunk testing and maintenance of toll and intertoll trunks terminating on the No. 4 ESS. The 51A has a sloping key panel and a number of built-in test equipment items. A CMS CRT sits on the left-hand side of the test position and is used as the major input/output device during trunk testing.

The key shelf is divided into three major parts; two are identical and known as Test Access Trunks (TATs), and the third provides communications functions. Each of the three parts of the key shelf is equipped to handle communications between the test position and test positions or test boards in other central offices. The TATs provide a set of keys and lamps to access and control any trunk appearance on the No. 4 ESS and/or a trunk accessed via the Switch Maintenance Access System (SMAS) 3B. The TAT trunk is similar to a standard E&M-lead trunk. Control of access to any trunk on the No. 4 ESS switch is via CMS to the 1A Processor. The SMAS access (SMAS is optional) is to the voice-frequency level point of trunks as they appear at the analog carrier system or the metallic facilities.

Test access is primarily requested through CMS. Test calls to specific trunks are established via the 1A Processor or through the CMS-SMAS

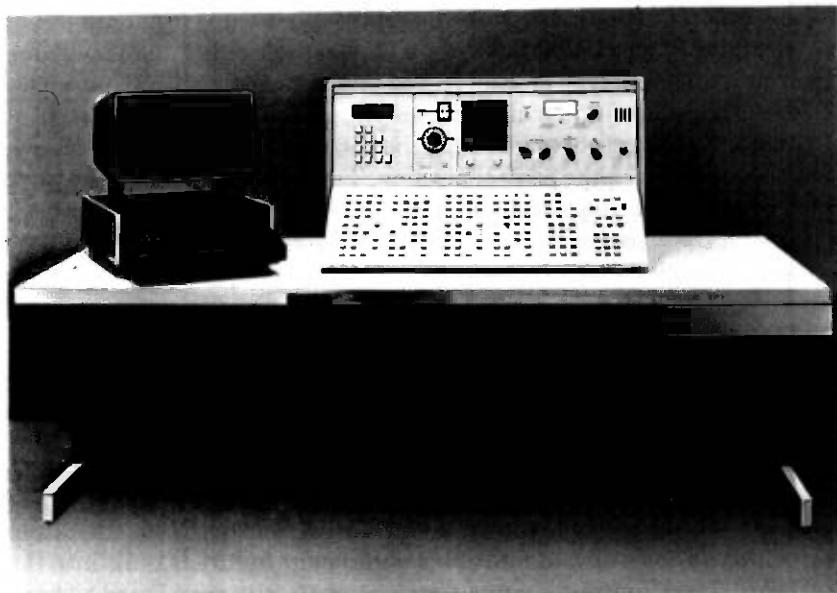


Fig. 10—51A test position.

register bank. Alternate methods are via the No. 4 ESS TTY channels or via the test position keyset to SMAS. The function of each kind of access can be further understood by a review of the functional block diagram (Fig. 11).

The keys and lamps are divided into six areas. These are TAT and circuit status, transmit and receive functions, and signaling and transmission test and control. These keys and lamps accomplish specific functions or access specific test equipment. The major test equipment measures signal level and frequency or noise. Impulse noise can also be measured. For intertoll trunks, direct connections can be made for measuring the performance of the echo suppressors. For toll connecting trunks, the four-wire return loss can be measured. Via SMAS 3B metallic trunk access, digital multimeter functions can be accomplished. For other measurement functions beyond those capabilities built into the test position, jacks are provided on the right side of the test position with work space and ac power outlets.

Via the No. 4 ESS network, any trunk in the office can be accessed, and then via the SMAS access direct cross-office measurements of the terminal equipment can be made. With the CMS CRT, demand tests can be made via CAROT and CTMS.

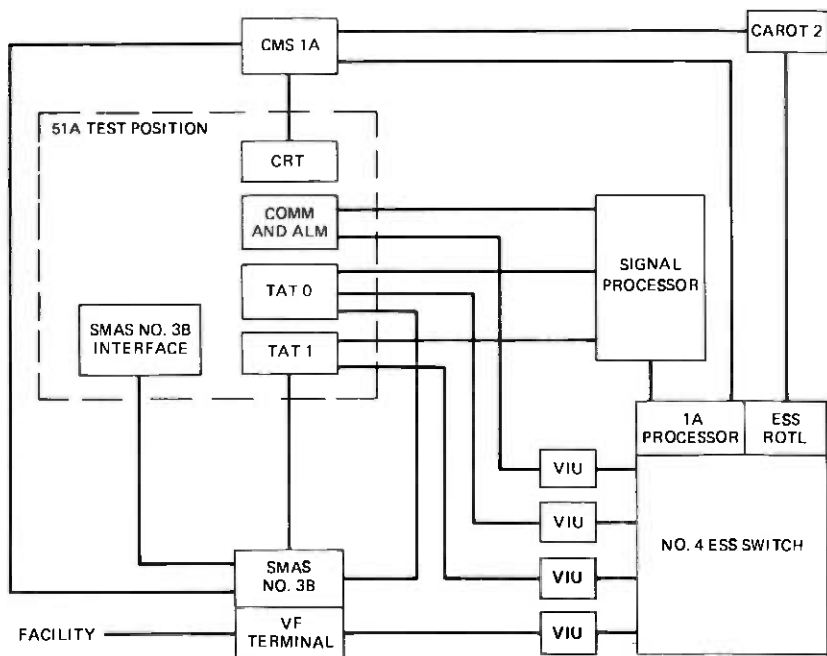


Fig. 11—Trunk accessing diagram.

The communications portion of the test position contains appearances for specific interoffice and intraoffice communications as well as standard telephone services. The keyset that is used for communications, SMAS access and other TAT functions contains *TOUCH-TONE*<sup>®</sup>, dial pulse and multifrequency signaling capability. A register is provided to store dialing digits and outpulse them when required.

### 3.5 Switched Maintenance Access System 3B

The Switched Maintenance Access System (SMAS) 3B is optional in the No. 4 ESS. If provided, it is available on analog trunks as well as other unitized facility terminals. On trunks employing analog carrier systems, SMAS provides monitoring (bridging) and splitting access to the +7, -16 transmission level points on the transmission leads (T, R, T1, R1) and to the E&M signaling leads. For metallic trunks, dc access is provided to the transmission leads and the signaling leads. SMAS is not applicable to the digital carrier trunks that terminate on the digroup terminals.

### 3.6 Centralized Automatic Reporting on Trunks (CAROT)

CAROT is a minicomputer system that provides a means for automatically scheduling and performing routine transmission tests on

outgoing trunks at the proper facility-dependent test intervals. Trunks beyond the turndown limit are reported to CMS after the trunk has been reaccessed and a confirmation test has been performed. Trunks with less severe impairments or trunks which could not be tested in four test attempts are reported at the end of the routine test period.

Data for the computation of the trunk transmission maintenance index is compiled from the results of routine testing. Statistics concerning the number of trunks tested and the number of various maintenance conditions found are provided. An additional report gives the operational health of the CAROT controller, ROTLs, and terminating test lines.

Real-time, on-demand tests of single trunks or trunk subgroups can be requested by personnel via CMS and will be performed as specified by data in the request. Upon completion of circuit order tests, CMS will pass the data describing the new or disconnected trunk to CAROT. The appropriate action will be automatically taken to update the CAROT data base.

### **3.7 Carrier Terminal Maintenance System**

CTMS (optional in No. 4 ESS) provides testing and maintenance for L-multiplex, M-multiplex, and other coaxial and radio facility equipment in the office. CMS and CTMS interact via a data link to provide trunk trouble sectionalization features on those trunks which CTMS accesses. The types of tests provided by CTMS to CMS (hence, at the 51A) are tone and noise level measurements and the detection of the presence or absence of SF tones on a specific trunk in the carrier terminal. This set of tests will provide positive trunk trouble sectionalization data on most of the troubles encountered.

## **IV. CIRCUIT ORDER PROCESSING**

A circuit order is the official request that an office receives to install, change, or remove trunks between itself and some far-end office. The request may include one or many trunks. The procedure used to accomplish the requested changes, additions, or deletions is known as circuit order processing. The following describes the equipment installation, data-base updating, testing, and coordination functions as they are performed in No. 4 ESS.

### **4.1 Circuit order initiation**

Circuit orders are received by the CMS from the telephone company circuit provision bureau via data link, magnetic tape, as as paper requests which are entered locally via CRT. CMS queues these requests until the machine administrator can inspect the list of new circuit orders and



schedule them for assignment. Also, a current list of all pending circuit orders is available via CRT. If a group of trunks is a modular engineered group (12 or 24 trunks, which all have the same characteristics), the circuit provision bureau will have submitted one circuit record layout card, and the machine administrator will be requested to make only one network assignment for this modular group. The assignments for the other circuits will be automatically generated. If the trunks are not modular engineered, separate circuit record layout cards exist and separate assignments are made for each.

#### **4.2 Data base changes**

The administrator assigns the circuit to the No. 4 ESS time-division network (this automatically assigns terminal equipment because of the connectorized arrangements) and enters this information into the 1A Processor data base by means of the recent change "add trunk" message (Fig. 5). For this message, the craftsperson simply specifies part of the common language name for the TSG (from the circuit order), the type of equipment needed (e.g., A6 without echo suppressor), the quantity of trunks to be added (e.g., 12), and the traffic number of the first trunk.

This network assignment information and other translation data is also entered into CMS via CRT. The machine administrator schedules the two work centers (TEC and TOC) to perform their respective jobs. The circuit order administrator then refers the order to the TEC and TOC work lists. CMS maintains a list of all pending circuit orders which can be displayed on a CRT.

#### **4.3 Testing activity**

The TEC has a list of pending work including circuit orders. (See Figs. 12 and 13.) In response to entries on the work list, the TEC makes the cross-connects, installs plug-ins, and runs office tests as required. After this work is completed, the TEC personnel update the TEC status (as it appears on the work list) to "CMP" for completed.

At some later time, determined by the dial administrator, the TOC work list will identify that this trunk on the circuit order is to be tested. The craftsperson can verify that the equipment has indeed been cross-connected and plugged in by checking the status of the circuit order which was set by the TEC. The craftsperson can then perform trunk testing from the 51A test position. When all circuit order tests pass for this trunk, the maintenance status of the trunk will be changed to "active" in the 1A Processor memory (by an input message submitted in the TOC via CMS) and CMS will be notified that the work is completed for this order.

MESSAGES:  
CIRCUIT:

220 WORK LISTS FOR PORT 1

221 TROUBLE REPORTS-----010

222 TROUBLE REPORTS CLEARED TODAY-----002

223 CIRCUIT ORDERS-----073

STATUS:

WORK I . . . . .

Fig. 12--TEC work lists.

MESSAGES: . . . . .  
 CIRCUIT: . . . . .

223 CIRCUIT ORDERS PAGE: 1 OF 10  
 CON ORDER NUMBER CIN

| CIRCUIT ORDER NUMBER | CIN                | STATUS: |                       | EARLIEST ITEM DUE DATE |
|----------------------|--------------------|---------|-----------------------|------------------------|
|                      |                    | ITEMS   | STATUS<br>MAC/TEC/TSC |                        |
| 200)Y7CCW            | 13GRELC6JC01T04T   | 01      | CMP/CMP/TST           | 07/03/76               |
| 198)W500ZD           | 1161WCHTKSBR14T04T | 01      | CMP/CMP/CMP           | 07/03/76               |
| 191)W500VG           | 25STLSM0914T04T    | 12      | CMP/XCN/TST           | 07/03/76               |
| 207)KCMBLTS8         | 8300KSCYH098TB04T  | 02      | CMP/XCN/TST           | 02/17/76               |
| 206)KCMBLTS7         | 8301KSCYH0AT39104T | 02      | CMP/XCN/TST           | 02/17/76               |
| 205)KCMBLTS6         | 8300KSCYH0AT39104T | 02      | CMP/CMP/TBL           | 07/03/76               |
| 204)KCMBLTS5         | 8301KSCYH00492A04T | 02      | CMP/CMP/CMP           | 07/03/76               |
| 203)KCMBLTS4         | 8300KSCYH003MGC04T | 02      | PND/XCN/TST           | 07/03/76               |

Fig. 13—TEC circuit order list.

#### **4.4 CAROT data base changes**

The machine administrator is notified that the work has been completed on this circuit order and can request CMS to issue an "in-effect" report to the circuit provision bureau and to update the CAROT data base for subsequent routine transmission testing. The transmission of an in-effect report and the update of test parameters to CAROT will be automatically effected across the CMS 1A interfaces to CAROT, and to the circuit provision bureau.

### **V. ONGOING TRUNK MAINTENANCE**

Major elements of the ongoing trunk maintenance function are trouble detection, removal from service, and trouble sectionalization and repair. The trouble-detection function is performed by the 1A Processor resident trunk maintenance software and by the CAROT system. When appropriate, the trunk involved is automatically removed from service. Detected troubles are always referred to the CMS for further resolution. The CMS provides automatic and semiautomatic features to notify craft personnel about the trouble and to support the testing and administrative processes that ensue through restoral to service of the then repaired circuit.

#### **5.1 Trouble detection and removal from service**

Trouble detection in No. 4 ESS is primarily accomplished through several automatic processes. These include:

- (i) Trunk tests or analysis stimulated by call failures
- (ii) Circuit supervision monitoring
- (iii) Routine trunk tests
- (iv) Carrier failure processing
- (v) Equipment monitoring.

In addition, summarized call failures and other data are periodically output to craft personnel for their inspection to ensure that undetected trouble conditions do not exist. Upon request specific, detailed data pertaining to the use of trunks and the disposition of calls is available in real time to the craft personnel. Manual trunk tests can be performed at the 51A test position in the TOC.

##### **5.1.1 Automatic trouble detection**

The call-processing software uniquely identifies every trouble condition it encounters to trunk maintenance, along with pertinent data. Each trouble condition is categorized as to whether or not it indicates a potentially faulty trunk and/or service circuit. For each trouble con-

dition that implicates a trunk or service circuit, one of three processes are followed:

- (i) Run a test on the trunk and/or service circuit.
- (ii) Return it to service and test it only if it fails again within the next "n" calls.
- (iii) Error analyze by returning to service and counting the number of successful and failed call attempts that occur. Compare the failure rate with that for its peer group (e.g., trunk subgroup) and, using a set of thresholds, determine if the individual trunk or circuit should be removed from service.

If a test is run and it fails, the trunk or service circuit is removed from service. If the test passes, the trunk or service circuit is entered into the error analysis process to protect against trouble conditions that the test does not detect. The results of the error analysis process can be to leave the trunk in service, leave it in service but report it to CMS as being suspect, or remove it from service.

Routine transmission tests and routine operational tests are automatically performed on all one-way outgoing and two-way trunks. The transmission tests are scheduled and performed under the control of CAROT (see Section 3.6). CAROT interfaces to the 1A Processor through a hardware interface known as a Remote Office Test Line (ROTL). The ROTL is controlled by software in the 1A Processor and connects to both the CAROT and the No. 4 ESS Time-Division Network (TDN). By controlling the ROTL hardware, the 1A Processor is able to communicate with CAROT via multifrequency and other tones, to connect transmission test equipment in the ROTL via the TDN to the trunk under test, and to allow CAROT to control that test equipment. The CAROT system, upon finding a faulty trunk, can request that a trunk be removed from service by the 1A Processor. The 1A Processor immediately notifies CMS of the outage, and CMS creates an outage ticket which it assigns to a craftsman. Subsequently, CAROT will notify CMS of the test results; these results will be included on the outage ticket. When CAROT identifies a trunk as marginal (i.e., not bad enough to remove from service), it notifies CMS and a trouble ticket is created and assigned to a craftsman.

Routine operational tests include the typical types: 103, synchronous, centrex II, centrex III, and step-by-step types. They are scheduled and run by the 1A Processor. The data base can specify the test types and directory numbers for up to two tests per trunk subgroup. This flexibility is needed when the far-end office utilizes different trunk equipment for through and terminating switched calls. Trunk failures are reported to CMS. Protection is provided against the removal of an excessive number of trunks because of faulty test data or far-end test lines. Up to 24 trunk subgroups can be tested concurrently in the routine mode.

Carrier failure alarms can be reported to the 1A Processor via miscellaneous scan points in a signal processor. Each scan point is related via translation data to either 12 or 24 trunks, all of which are assigned to consecutive appearances on a Voiceband Interface Unit (VIU). This modularity is inherent in the connectorization plan used for the terminal equipment. The trunks will be removed from service and restored to service in response to the alarm. Each transition is reported to CMS.

Trunks terminating on a Digroup Terminal (DT) are not assigned to an alarm scan point. Rather, the digroup terminal has built-in alarm detection capabilities for reporting directly to the 1A Processor over the peripheral unit bus. In the case of a T-carrier failure, the trunks are removed from and restored to service and all transitions are reported to the CMS. Other impairments on T-carrier are also detected by the DT (e.g., slips) and are reported to the CMS by the 1A Processor.

The No. 4 ESS has the ability to continue to process calls even though one or more common control peripheral equipment units have totally failed. The trunks associated with the failed equipment are inoperable and are removed from service to prevent their seizure for use in a call. One or more modules of trunks, each of which contains 120 to 4096 trunks, is processed either in base-level processing or in a phase of software initialization. The trunks are automatically restored when the equipment is restored. For common control equipment failures, the CMS is not notified of the trunk outages. Rather, a message is sent to a teletypewriter located in the TOC to notify personnel of the outage. There is no repair function to be performed on the trunks. However, notification of far-end offices and coordination of the restoral process may be necessary.

### ***5.1.2 Data for manual trouble detection***

The plant and traffic measurement systems periodically present data organized by control area, trunk type, and failure type on all call failures in the office. These counts, together with base counts of the traffic volume, can be manually used to monitor trunk performance. This information can be scheduled as it is needed. In addition, once each day, or on demand, a list of the trunk subgroups with the worst performance is output on hard copy to the TOC. The list contains the eight trunk subgroups, in each of eight call-attempt ranges, that had the largest number of call irregularities in any 15-minute period since the list was last periodically output. Two-way trunks are considered separately for input and output. The call irregularity rate of a trunk subgroup on the list is manually determined to be acceptable or unacceptable.

In response to plant/traffic measurement reports, the worst trunk subgroup list, or other trouble detection processes, there can be a need

to collect specific or exhaustive data or counts of call irregularities on trunks. The "data trapping" feature provides the capability to collect data for manual analysis. A "trap" is established by inputting a teletypewriter message to the system that identifies:

- (i) The type of call irregularities (e.g., permanent signal, partial dial or "all")
- (ii) The trunk(s) or trunk subgroup(s)
- (iii) The sample rate (can be 100 percent)
- (iv) The schedule (start time, stop time, repeat daily option).

Each time a call irregularity that is being "trapped" happens, a printout will be generated containing all pertinent data. This would include trunk identity, digits, service circuits, call-processing software state, failure identity, and other data as applicable. An option exists to count the events in addition to or in lieu of the data printouts. Up to 31 such "traps" may exist at any one time. In addition, up to four "office traps" may be established. Additional features of the "office trap" are that they each apply to all trunks in the office and a separate sample rate can be requested for each call irregularity type requested.

### **5.1.3 Removal from service**

Faulty trunks are removed from service to prevent their use for a customer call. Except for removals due to common control equipment failures, all outages are reported to CMS, which creates a trouble or outage ticket for each occurrence. When trunks are removed because of common control equipment outages, messages are printed on a teletypewriter in the TOC. The number of circuits involved may be from 120 to several thousand per occurrence. No repair action by TOC personnel is needed; however, notification of far-end offices may be required. Trunks may also be removed from service because of a manual request via the CMS or by CRT directly to the IA Processor. As with automatic removals, the CMS is always notified when a trunk is manually removed from service.

Two mechanisms are used to ensure that an excessive number of trunks is not removed from service by the automatic system. The first is that automatic processes, which might be the source of failure while the trunks are good (e.g., a bad far-end test line used for routine tests), cannot remove more than a fixed percentage of a trunk subgroup from service unless manually permitted to do so. The second safeguard is that a continuous audit of all out-of-service trunks is made against redundantly stored information that authorizes the outage. When an automatically detected condition (e.g., carrier failure alarm) is determined to be the only reason for a trunk to be out of service, the trunk will be

automatically returned to service (and CMS notified) when the condition comes clear.

## **5.2 Trouble sectionalization and repair**

Troubles can be automatically detected and reported to CMS by the No. 4 ESS or CAROT, or they can be manually reported to CMS via a CRT input. For example, suppose that CAROT detects excessive loss on the incoming path of a trunk between Chicago, Illinois, and Orlando, Florida, during routine testing. After repeating the test to verify the trouble, CAROT notifies the No. 4 ESS to remove the trunk from service and then sends a report to the CMS identifying the trunk and the trouble encountered. CMS creates a trouble ticket, assigns it to a work list, and sends the TOC a work message on the CRT screen (Fig. 14).

The heading of this display shows a "1" in the WORK field; this indicates new work on the first work list. The lack of other entries in the remainder of the heading indicates that there are no other messages pending and no individual trunks currently being worked on.

The "WORK 1" message can be sent to the CRT independent of any display that is on the CRT. In the general case, the CRT will have a display other than that shown in Fig. 14. If the craftsperson responsible for the work list does not know what to do next, the 3-digit input command "100" should be entered via the keyboard. This command generates the display in Fig. 14, which is an index of possible commands.

Entering "201" causes the current list associated with work list "1" to be displayed, with the most recent entry at the bottom of the list (outage number 273 in Fig. 15).

The craftsperson selects it to work on by entering "110/0273" (the "110" activation command and the assigned outage number) and then enters "300" to see the outage ticket (Fig. 16).

Noting the nature of the problem (see comments field in Fig. 16), the craftsperson decides on a strategy for locating the trouble and inputs a command (710/102) to connect the faulty trunk to a test tone in Orlando, Florida (Fig. 17).

CMS communicates the command to the No. 4 ESS, which seizes the trunk, outpulses the directory number for the far-end test line, and connects the trunk to the 51A test position.

The craftsperson then requests sectionalization measurements by inputting the command "631/1/R" (Fig. 18). The CMS sends a message to the Carrier Terminal Maintenance System (CTMS), which makes a series of tone measurements in the carrier equipment and passes the results back to the CMS. The CMS processor compares measured values with expected values and determines whether the trouble is in the near-end equipment, the carrier equipment, or the distant office. In this



```

MESSAGES:
CIRCUIT:
100 TRUNK OPERATIONS CENTER (TOC) -- CHS 1A FUNCTIONS
110 ACTIVATE CIRCUIT
120 ACTIVATE CIRCUIT ORDER
130 RELEASE
150 POSITION LOG
200 WORK LISTS
300 OUTAGE TICKET
400 CIRCUIT DATA
500 CIRCUIT STATUS
600 CIRCUIT TESTS
700 MANUAL ACCESS
800 MISCELLANEOUS
900 WORK SUMMARY
920 OFFICE DATA
940 ADMIN REPORTS
000/PDS NO. - SIGN ON 001/PDS NO. - SIGN OFF
201

```

WORK: 1

STATUS:

320 OUTAGE LOG      350/C6N      CKT ORD TICKET

--- CONTROL POSITION ONLY

4

Fig. 14—CMS functions.

WORK: . . . . .  
 STATUS: . . . . .  
 MESSAGES: . . . . .  
 CIRCUIT: . . . . .  
 201 OUTAGE REPORTS PAGE: 01 OF 01  
 CIRCUIT IDENTIFICATION      OUTAGE REPORTED      REPORTED BY      WHEN      TURN DOWN      FAR  
 #0028) 1020RLDFLMA01T57T      HAW      ESS      0212 0906      0212 0906      -----  
 #0217) 16LSANCA0307T57T      OPR      ESS      0212 1005      0212 1005      -----  
 #0273) 1050RLDFLMA01T57T      LVL      KRT      0212 1230      0212 1230      -----

110/0273

4

Fig. 15—Outages work list.

```

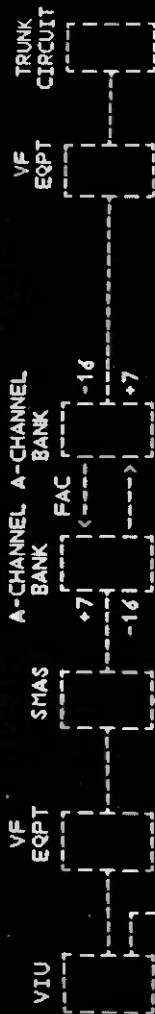
MESSAGES:
CIRCUIT: 0273) 1050RLDFLMA01T57T      WORK:
STATUS: MTC.LK0
300 OUTAGE TICKET: 105      CHCGILCL57T -M 0RLDFLMA01T
301 OUTAGE REPORTED: LVL      BY: KRT      TO: 15      WHEN: 0212 1230
      NEAR END
      FAR END
302 TURN DOWN:      WHO: --      WHEN: 0212 1230      WHO: ---      WHEN: ---
303 TURN UP:      WHO: --      WHEN: ---      WHO: ---      WHEN: ---
304 REFER OUTAGE:
305 MAKE GOOD:
306 CHECK ON REFERRED OUTAGE:
307 RETURN REFERRED OUTAGE:
308 REASON FOR OUTAGE: ---/
309 COMMENTS:      CAROT LEVEL MEAS: Z-A 10.1 DB* (EML 6.0)
310
311
312 CLEAR OUTAGE TICKET

```

Fig. 16—Outage ticket.

MESSAGES: 0273) 1050RLDELMA01T57T WORK: MTC.LK0  
 STATUS: MTC.LK0

700 MANUAL ACCESS CIRCUI TID: 105 CHCGILCL57T -M 0RLDELMA01T



TEST LINE  
 1100 QUIET TERM  
 1101 TEST BOARD  
 1102 MILLIWATT  
 1103 OPERATIONAL  
 1104 TRANSMISSION  
 1108 ECHO SUPR  
 1000 NO OUTPULSE

TAT(X) TEST POSITION CONNECTIONS:

710/X/TEST LINE NO. OR FAR END TELEPHONE NO.  
 711/X TERMINATING MODE FOR 103  
 712/X/FAR END NO. AUX MONITOR  
 713/X SMAS  
 720/X

710/1/102

Fig. 17—Manual access commands.

```

MESSAGES:
CIRCUIT: 0273) 1050RLDFLMA01T57T          STATUS: MTC.LK0      WORK:
630 FACILITY TESTS
CIRCUIT ID: 105          CHGILCL57T -M 0RLDFLMA01T
FACILITY ID: 13 A      0112 CHGILCL57T0RLDFLMA

VF EQPT  SMAS  A-CHANNEL BANK  FAC  VF EQPT  TRUNK
VIU  [-3]  [-3]  [+7]  [-16]  [-16]  [+7]  [-3]  [-3]
|<-----NE EQPT----->|<-----LMX EQPT----->|<-----FAC OR FE EQPT----->|
|<----->|<----->|<----->|<----->|<----->|<----->|<----->|<----->|<----->|
631/X/T OR R  RUN FACILITY TEST ON TRANSMIT OR RECEIVE
T
| 1 - LEVEL SECTIONALIZATION | 1 - TEST TYPE
| 2 - NOISE SECTIONALIZATION |
| 3 - SF LEVEL SECTIONALIZATION |
| 4 - SPREAD
631/1/R

```

Fig. 18—Sectionalization test.

```

MESSAGES:
CIRCUIT: 0273) 1050RLDFLMA01T57T
320  OUTAGE TICKET LOG
      TIME      EVENT
      02/12 12:30 CAROTTST
      02/12 14:08 CTMSTST

      WORK:
      STATUS: MTC.LK0

      DATA
      LEVEL  A--Z  Z--A
            6.0 DB 10.1 DB*

      TBL RCV: NEAR END EQPT!!
      LEVEL DEVN  4.1 DB*
      MG PLT DEVN +00.5 DB
      GP PLT DEVN -00.6 DB
  
```

Fig. 19—Sectionalization results on outage log.

case, the trouble was in the No. 4 ESS office as noted on the outage log (Fig. 19).

The craftsperson can then enter the results of the tests on the trouble ticket and refer the trouble to the TEC. The personnel in the TEC can repair the faulty equipment, utilize the CMS to verify the problem has been fixed, put the trunk back in service, and clear the trouble ticket.

The preceding example was based on the assumption that the craftsperson was familiar with the CMS system and knew exactly what commands to enter to perform the desired functions. However, at any step it is possible to recall the "100" display (Fig. 14) to obtain a list of functional areas and then to enter the code (e.g., "700") for a functional area to obtain a list of functions (Fig. 17) within the area.

## VI. SUMMARY

A No. 4 ESS can have over 100,000 trunks and the associated data base to support call processing. Both the trunks and the data base require installation, updating, and maintenance to meet initial and changing needs and to resolve and correct trouble conditions. The data base has been described.

The large quantities of data and trunks concentrated in a single switching machine created a concern that the cost of installation, update, and maintenance not grow disproportionately when compared with existing switching systems. Also, there were objectives to improve these processes to make them more efficient and less error-prone. The large concentration of trunks and data in No. 4 ESS that created these concerns also made it economic to develop auxiliary systems and subsystems to deal with them.

The systems developed include a Western Electric software system to generate, on a general-purpose computer, the initial data base from office wiring and trunking information, 1A Processor resident software to administer and maintain both the data base and the trunks, and three minicomputer systems (CMS, CAROT, and CTMS). The CMS provides a common interface to office personnel for all trunk maintenance activities and provides many administrative and coordination functions. CAROT and CTMS automate transmission testing capabilities that rely on an extensive data base.

Throughout the design of these systems, an emphasis has been placed on enhancing the personnel interface with the machine and on minimizing the need of manually produced records. Routine and administrative functions have been automated.

100  
100  
100  
100

100  
100  
100  
100

100  
100

100  
100  
100  
100

100  
100

100  
100  
100  
100

100  
100

100  
100  
100  
100



## **No. 4 ESS:**

# **Program Administration, Test, and Evaluation**

By P. S. McCABE, J. B. OTTO, S. ROY, G. A. SELLERS, JR., and  
K. W. ZWEIFEL

(Manuscript received July 8, 1976)

*A number of hardware and software support systems are used in the design, testing, evaluation, and administration of the No. 4 ESS software. The Program Administration System (PAS) provides a management facility for the large data base that consists of the multitude of ESS programs. These programs must be changed, controlled, and manipulated in a uniform fashion to produce the various official releases of the No. 4 ESS software without jeopardizing the integrity of the data base. The IA Processor Utility System uses a minicomputer and a hardware interface with the No. 4 ESS as an operating system for the interpretation, initialization, execution, monitoring, and display of tests written in a high-level utility language. Thus it makes available a convenient means for testing and debugging No. 4 ESS software in a system laboratory environment. A directed-graph model, describing the sequential stimulus-action-transition behavior of a software process under test, acts as a reference and a starting point for the Automated Testing and Load Analysis System (ATLAS). Components of ATLAS then automatically derive tests and employ test directives embedded in the model to apply and monitor tests for the modeled process. Testing of the ESS software under high traffic loads is achieved in the laboratory by the Programmed Electronic Traffic Simulator (PETS). This system simulates the peripheral equipment and monitors the ESS responses for deviations from a norm.*

## **I. INTRODUCTION**

A large development undertaking like No. 4 ESS—one that spans years, involves several organizations, and is built by the combined efforts of many people—is seldom described solely in terms of the final mar-

keted product. Numerous supporting activities must exist to assist the project through its design, implementation, testing, evaluation, and production of official releases. A number of support tools that were developed to meet the demands of producing quality software in No. 4 ESS in a systematic and efficient fashion evolved to be systems of significant sizes themselves. A common objective behind these support systems was to provide their users, viz., the community of No. 4 ESS programmers, with powerful yet easy-to-use tools to assist in the development of a quality product.

Four such systems of diverse scope, content, and applicability but linked by the common objective above are described herein. The Program Administration System (PAS) provides management facilities for the sources, object modules, listings, and other data entities associated with the many programs constituting the No. 4 ESS software package. PAS is described in Section II. In Section III, the 1A Processor Utility System that enables program testing and debugging to be done in the controlled environment of the System Laboratories is described. A novel testing concept and its application methodology and component tools, collectively known as the Automated Testing and Load Analysis System (ATLAS), is described in Section IV. While ATLAS primarily concentrates on testing the complex logic of the programs and their interfaces, engineering aspects of the software and dynamic performance are evaluated in the face of various levels of system loading by the Programmed Electronic Traffic Simulator (PETS). PETS is described in Section V. Each section is essentially complete in itself. The objectives, components, features and usage experience related to a system are covered in its associated section.

## **II. PROGRAM ADMINISTRATION SYSTEM**

### **2.1 General**

It was anticipated that the object code of the No. 4 ESS software system might reach 1,000,000 words in size, that the number of separately assembled programs (PIDENTs) would be 1000 or more, that over 200 program designers would be involved, and that the volume of textual documentation associated with the software would exceed 100,000 pages. It was clear that no system of manual procedures, no matter how vigorously enforced, would suffice to keep order and accountability in the creation and administration of the No. 4 ESS software. An all-inclusive system of facilities for both programmers and administrators was needed.

Further, the No. 4 ESS software would continue to evolve. New features and program changes would be incorporated into new generic programs (GENERICs) while older GENERICs would simultaneously be supported.

The No. 4 ESS software system consists of two major components: the No. 4 ESS application software and the 1A Processor software. The 1A Processor software is used in both the No. 4 ESS and the No. 1A ESS (a local switching system utilizing the 1A Processor). These two pieces of software were developed concurrently but in two different organizations. Hence there was a need to administer them as two separate but closely coordinated communities. A further administrative division came from the need to provide special facilities for the diagnostic maintenance programs, which were used in the generic programs, and for the Installation Test System used by Western Electric Company engineers in the installation of No. 4 ESS offices. Finally, the support computing system, the IBM 360/Time Sharing System known as TSS, was anticipated to have insufficient capacity to accommodate the entire development in a single system. Hence the administration of the software development was to be carried out in several separate TSS systems. Within a single support computation system, it would be useful to allow several communities of users to be recognized and to be able to divide the data base along administrative boundaries. The Program Administration System (PAS) was designed to address the problems of administering the large software system of the No. 4 ESS.

## **2.2 Objectives**

The primary objective of the Program Administration System is to provide for the creation, modification, preservation, and administration of the No. 4 ESS and 1A Processor software and their associated data bases. This implies an orderly structure for all data (both temporary and permanent) and a set of facilities which enable the creation of data, the copying and erasing of data, the naming and renaming of data sets, and the combining of data sets to form new data sets. These facilities must allow for moving data between various types of storage and must include reporting which is necessary to keep the users up to date on the state of their data.

The second objective is to allow the user and the administrator\* easy and efficient access to the data base and to the support programs which are used in the development and testing of each PIDENT. Access must be concise to eliminate ambiguity over what is being accessed. Access must also be simple to minimize errors in the user's specification. In the case of data sets, a structured and functional naming convention is needed. This allows the user to readily determine the name of each data set through knowledge of the function of the data set. In the case of support programs, the calling sequences and parameters must be uni-

---

\* The administrator is responsible for the production of the GENERIC from the multitude of component parts known as PIDENTS.

form so the user can easily remember them. There is also a need to provide values (termed "defaults") for parameters when the user does not supply values. The use of defaults is a powerful administrative tool which allows the system to produce correct calling sequences even when the user does not know the correct values.

The third objective is to provide the administrator with the facility to manipulate the data base on a large scale to accomplish backup protection against both system failure and data mutilation due to inadvertent destructive actions on the part of users or administrators. It is important that these facilities be comprehensive and foolproof.

The fourth objective is to make maximum use of the basic TSS facilities in the implementation of the Program Administrative System. This approach minimizes the need to create new basic facilities and assures a robust design in the face of changes in TSS. An additional benefit is the consistency in appearance this produces between the Program Administration System and its host, TSS.

The fifth objective is to make efficient use of computing resources. Since the Program Administration System has a large number of users and has a heavy influence on how they use the computing facilities, it is important that its usage be efficient. In a similar vein, the administrative environment should discourage duplication of storage (for example, users making their own backups to protect their data) and processing.

The final objective is to implement the system in a modular and extensible structure. The Program Administration System functions within an evolving environment and frequently must accommodate change so that users are not impacted. It is important that the system be designed to be readily modified and extended to handle changes.

### **2.3 User facilities**

The user facilities provided by the Program Administration System fall into three categories: facilities which support the basic conversational and batch processes that are associated with PIDENT development, facilities which allow the user to create and save or execute batch jobs while engaged in conversational processing, and facilities which provide administrative and general-purpose computing capabilities.

The system provides the facility to edit PIDENT source data sets, cause assembly of a PIDENT (inputting the PIDENT source data set to the assembler), load the Output Program Module (OPM), simulate the execution of the loaded PIDENT OPM, or execute any meaningful combination of those processes by issuing a single terse command supplied with a single parameter: the PIDENT name. The proper data sets will be accessed, modified, or created in the Program Administration System data

base by each process without further attention from the user. For the assembly process, No. 4 ESS and 1A Processor software systems utilize a set of libraries called COMPOOLS which contain data definitions, data structures, and macro definitions. Each PIDENT utilizing any of these named values or structures obtains the correct definitions at assembly time via access to the correct COMPOOL. The Program Administration System assures that the proper COMPOOL is made available to each assembly.

A further extension supported by the Program Administration System is the concept of a PROGRAM UNIT, or a part of a PIDENT which may be assembled separately. The system allows the user to create an arbitrary collection of PROGRAM units, and to operate on each of them as though they were PIDENTS. The user can combine any group of PROGRAM UNITS into a single PROGRAM UNIT or into a PIDENT at will. This facility allows the user to develop a PIDENT by building it from smaller functionally oriented pieces.

A full set of batch facilities, all of which can be invoked while the user is in the conversational mode, are provided. The user can build data sets which consist of batch jobs (which may include both PAS and non-PAS commands and processes) and can execute these jobs concurrently with the conversational jobs. The user can create a series of such jobs and be guaranteed of the sequence of execution. The user will be informed of the batch job beginning and completion via messages sent from the batch job to the user's conversational job. If the user attempts to exit from the Program Administration System without specifying the disposition of any batch job created, he will be prompted for the proper disposition (execute, save, or erase).

The user is provided with access facilities other than the standard processes previously described. For instance, the user can create "private" versions of PIDENTS or PROGRAM UNITS and can maintain them separately from the official versions. The Program Administration System provides a facility which indicates the elements to be included in the official version. This facility is accessed by the administrator when assemblies are done for system loads.

A user may issue TSS commands while using the Program Administration System and can temporarily leave and return without invoking the overhead that is associated with a user's initial access. No TSS facilities are denied the user under the Program Administration System except those which actually conflict. More details on these issues will be given in Section 2.5.

#### **2.4 Administrator facilities**

The administrator facilities are primarily directed to the task of organizing and processing the PIDENT-associated data in accordance with

the needs of the project as a whole. In a more subtle vein, the administrator facilities must provide the ability to redirect or change the results of user processes and to change the details of the facilities available to the users. Finally, the administrator must have tools to create and permute the administrative data itself and to use that data to control user processes and administrator processes.

The Program Administration System provides a data base which is owned by the administrator and made available to the user on a shared basis. The administrator can permit access on a read-only basis, a read/write basis, or an unlimited basis (the user can create or destroy the data set); or the administrator can deny all access. The permission can be established at a data set level or over groups of data sets. The system supports many administrative communities, allowing each administrator autonomous control over a community, but allowing the transfer of data between administrative communities at the administrator level. Hence the No. 4 ESS administrator can gain access to the 1A Processor community in order to prepare a generic load for No. 4 ESS.

The administrator has the facilities to build the structure of the data and processes necessary to add a new user to the Program Administration System and to build the administrative data which would allow the new user to access all necessary PIDENT data. The administrator can reestablish all of these structures from data saved in the administrative data base if some system disaster should strike. The entire data base (including the administrative data) is protected by a powerful data base backup facility executed by the administrator. This facility allows the administrator to select the criteria for data set backup and the frequency of backup. The data base resides on a different media than the backup, which makes it unlikely that the backup can be damaged by the same mechanism which damages the data base.

The administrator has the same facilities as any user. Hence the administrator can perform tasks on behalf of the user, or even in spite of the user when necessary. In addition, the administrator has free access to all PIDENT data and can execute processes which are denied to any user. This is a heavily used facility when the software under development is under strict change control (termed "frozen"). Under these circumstances, only the administrator can modify a PIDENT, and users may make modifications only through the change control process.

The administrator also has the facility to modify the way in which a user accesses the support programs and to determine which version of a support program is made available to users. This facility further allows the administrator to introduce changes into the user environment while the user is using the Program Administration System.

In addition to the facilities for controlling user access to the data base,

which are described earlier in this section, the administrator has the overall capability to control the allowable syntax and semantics in that community and to control exactly which facilities are invoked with each user or administrator command. The definition of most processes and the parameters passed to each process (including default values) can be modified by the administrator.

When changes are introduced that require user notification, the administrator can issue messages to all conversational users and can send messages to be included in all batch jobs. An announcement can be sent to each user as they log into the system. There is a "news" facility which the user can use to obtain general information supplied by the administrator.

For changes that require that the system not be in use, the administrator can invoke controls that prevent further conversational and batch jobs from beginning. The administrator can also query to determine the identity of any users logged in. The administrator can then send messages to the conversational users requesting them to leave the system (or even issue the command which logs the user off), and can wait until the executing batch jobs finish before introducing the changes into the quiescent system.

## **2.5 Implementation**

The Program Administration System is implemented on and executes under the IBM 360/67 Time Sharing System, which is a large general-purpose conversational computing system. The principle characteristics of this operating system are:

(i) A very large addressing space (16 million bytes) available to each user, called "virtual" memory.

(ii) A time-shared operating system which gives seemingly continuous service to reasonable numbers of concurrent conversational users.

(iii) A powerful and flexible command system which includes the capability to build, preserve, or delete individual user commands.

(iv) A "default" facility, which allows prespecified values to be supplied to parameters of commands when the command has not supplied values explicitly.

(v) A hierarchical catalog which allows each user to create, access, and erase data sets by name, and allows users to share each other's data sets. The sharing permission is granted by the owner and can be read-only, read/write, or unlimited.

(vi) A group of data-set access methods which deal with the complexities of concurrent access of a data set by several users and control the interwrite problem using a system of access locks.

(vii) A complete set of batch-processing facilities and a capability which allows users to initiate batch jobs while in the conversational mode.

(viii) A dynamic loader which automatically loads programs invoked by any command. The TSS and the user provide libraries to the loader which are searched for programs in response to commands or to resolve references in programs being loaded.

The user of TSS is supplied with an environment of commands, defaults, and facilities (programs) when initially joined to the system. The user may subsequently add to that environment or change the values of the defaults or the function of the commands and may elect to save these changes for future use or let them disappear at the conclusion of the current session. An address space and identity known to TSS as a task is created for each user when he logs onto TSS either conversationally or as a batch job. At a given time, a user may have in existence one conversational task plus as many batch tasks as he has concurrently executing. When the user logs off TSS or his batch job completes, the associated task disappears. Changes to the user's environment during the execution of the task are normally lost when the task ends, but can be preserved by explicit command of the user.

The Program Administration System operates as a subsystem under TSS, and a major element in the design was the attempt to keep a consistent appearance to the user. In many cases, TSS did not have facilities to deal with the concept of a subsystem and much of the significant design work involved the creation of these facilities (most of which were then incorporated into TSS for the use of other subsystems).

When a user first accesses the Program Administration System (a process called "joining"), an environment is built for him. In subsequent accesses, the user will have this environment merged into the TSS environment, and any conflicts (commands or defaults with identical names) will be resolved in favor of the Program Administration System environment. The Program Administration System will interpret and execute all commands issued by the user. The user may execute any TSS command by prefixing the command with an underscore, which will cause the system to pass the command to TSS for interpretation and execution. Similarly, the user may execute a command of his own creation by prefixing the command with a dollar sign, which will cause the Program Administration System to restore the user's original environment and then pass the command to TSS for interpretation and execution. The user may also exit from the Program Administration System on a temporary basis to do work in TSS and then return without causing the reestablishment of environments.

After command interpretation by the Program Administration Sys-



tem, the result is usually a call to invoke a program and a stream of parameter values to be passed to that program. As described earlier in this section, the TSS dynamic loader resolves the program call by searching a list of libraries (called the JOBLIB chain) which has been established by TSS and the user. When the program is first found (the program may appear more than once in the chain of libraries), it is loaded and its parameters are passed to it before execution. If there are parameters expected by the program for which no values have been passed, TSS uses the default from the task environment for the value (if one exists). The Program Administration System utilizes these facilities by supplying a library or series of libraries to be put at the *top* of the JOBLIB chain for the user's task and by supplying a set of default values to be merged into the task environment. The administrator can change the libraries and default values freely, and the user is given access to a portion of the defaults to allow the tailoring of his environment. Further, the Program Administration System rarely makes a direct call to a support program; instead it issues a command. These commands are built through the TSS command-building facility and are part of the environment which is merged into the user's task. These commands can be easily altered by the administrator and provide power and flexibility in the mechanism which invokes support programs. Since these commands are used to pass parameter values to the support programs, it is possible to override the parameter values that the user has supplied or those that are default values.

Some programs expect to receive their input from the user's terminal or from a data set which will be identified in the terminal input. TSS uses a mechanism called the "source list" to serve as a buffer between the terminal data and the program which receives it. The Program Administration System manipulates the source list when it is necessary to provide input to such a program before the user begins his input or when it is necessary for the Program Administration System to "get in the last word."

The conversational user may use the "attention" key on his terminal to gain control over program execution, that is, to interrupt the execution of a program. When the interruption is accomplished, TSS will allow the user to carry out some other activity and then resume (if the user desires) the interrupted execution. TSS provides the facility to allow the interrupted program to recognize the attention, and provide its own response (called "fielding the interrupt"). The Program Administration System utilizes this facility extensively to allow or deny each support program the fielding of the interrupts during its execution. Further, the system may also field other types of interrupts such as those caused by input/output or system routines. The objective of this approach is to provide the user with a reasonable response to an attention at any phase of

process execution, and to shield the user from confusing system responses. Finally, the Program Administration System makes extensive use of the TSS interrupt-handling facilities to provide task-to-task communications. These capabilities include administrator announcements to all users, administrator-user messages, user-user messages, notification messages to a conversational user regarding batch job activity, and statistics gathering.

The data base for the Program Administration System is implemented entirely in direct-access disk storage. The majority of this storage is on private disk packs, although most storage in TSS is implemented in a common pool of direct-access disk storage called "public storage." The use of private disk packs provides the freedom to administer the storage independently between communities and independently of the rest of the TSS users. It also allows the user the option to keep his general storage separate from the project data base, and allows a backup-and-restore facility which can adopt administrative boundaries. Backup of this data base is done to magnetic tape on a regular basis by the administrator, and restoral of a single data set or the entire data base can be done by the administrator. Backup is currently done daily on all data sets changed that day, and the entire data base is backed up weekly. All project-oriented data generated under the Program Administration System and some project-oriented data generated within our support systems is maintained in this data base and protected through the backup procedures.

The Program Administration System command analysis is performed in a program which executes the lexical and syntactic analysis from a data table produced by a compiler-compiler. The lexical and syntactic rules were compiled into a data table, and this table is used to drive the analyzer. The result of this analysis is a data structure (a job stack) and a table of values (hash table), which can be used by the interpreter in the execution of the job stack. The interpreter program operates on the job stacks to produce calls to programs or commands in TSS, together with the parameter values to be passed to those processes. The lexical and syntactic rules and the values of constants needed for the hash table are stored in an administrative data set for each administrative community and are used to build the user and the administrator environment when each accesses the Program Administration System.

## **2.6. Commentary**

For several years the Program Administration System has been used continuously by several projects on two separate IBM TSS systems.

The objectives set for it were met, and there is no doubt that chaos would have reigned without such a system. Several observations should be made on the design approaches and the objectives themselves.

The decision to implement the system as a subsystem of TSS was a good one. Much design effort was saved by minimizing the reinvention of many facilities already available in TSS, and in many cases the extensions made for the Program Administration System were useful to other TSS users. TSS evolved through many software releases and through conversion to the IBM 370 series of hardware during the development period, and the impact of this on the Program Administration System was minimal. There was virtually no impact on the Program Administration System users from these TSS changes. The users of TSS and of the Program Administration System benefited from the consistency of appearance and function.

Although much was done to make the structure of the Program Administration System flexible and very general, not enough generality was achieved. It is now clear that all of the implications of generality may still not yet be known. A measure of generality which would allow an administration system to create subsystems within itself, each with all of the facilities of its progenitor, seems to be a good start on this problem. The Program Administration System was able to meet all of its original requirements, and was able to adapt to many new and unexpected ones, but eventually the necessity to use a "kludge" to solve an immediate problem produced a less and less modifiable system.

Finally, it is clear the entire software development process, not just the program-writing phase, is in need of administration. Hence there is a need for a development support system which would provide facilities for all phases of software development.

### III. 1A PROCESSOR UTILITY SYSTEM

#### 3.1 General

Convenient and efficient debugging tools were required for the development of the large body of software that would be resident in the 1A Processor. As mentioned earlier, the 1A Processor itself was being developed for application in two major Electronic Switching Systems:

- (i) The 1A ESS designed to serve customers at the local level.
- (ii) No. 4 ESS designed to serve the national network as a toll switch.

Because of this dual application, it was necessary to design a utility system which would serve both the common and individual needs of the two systems.

Since the ESS laboratories represented millions of dollars of invested capital, laboratory time was a precious development resource. To meet schedule goals, a total of four system laboratories utilizing the 1A Processor were initially constructed. Two of these were No. 4 ESS labora-

tories and two were 1A ESS laboratories. Because there was neither floor space nor money for a fifth laboratory, efficient utilization of the existing four laboratories was a necessity. Efficiency here means the time spent in test execution relative to the total laboratory time devoted to that test. Total time includes the overhead of setting up the test and of outputting test results.

To enhance user productivity, tools had to be provided that would give the most useful results in the form of run-time data with the least amount of personnel effort and time. These objectives led directly to development of the 1A Processor Utility System which would allow a batch mode of operation wherein the user could leave his test for an operator to execute.

The input language for the utility system user needed to be intuitively straightforward, and symbolic representation had to be permitted to the greatest possible extent. Output data needed to be conveniently formatted, with symbolic representation employed wherever possible.

Since data collected from a real-time multiprogram environment can have a high degree of time dependency, there are circumstances where the act of data collection should not interrupt the operation of the machine under test. Autonomous collection of data from key processor locations was therefore an important feature that needed to be provided by the 1A Processor Utility System.

To meet these needs, the 1A Processor Utility System employs an SEL 810B minicomputer with a vendor-supplied real-time monitor system. It is programmed to provide both user and administrative functions which are described herein. There is a complex hardware interface known as the Utility Test Console (UTC), and finally there is a body of utility code in the 1A Processor itself.

### **3.2 Hardware configuration**

The hardware configuration of the 1A Utility System is explained through use of Fig. 1.

The upper part of the figure shows the main elements of the 1A Processor. The central control is the central processing unit, and it is fully duplicated. There are four bus systems. Generally speaking, the call stores on the call store bus contain time-variant data, whereas the program stores on the program store bus contain invariant data including the program itself. Both of these stores are constructed of magnetic cores. The Auxiliary Unit (AU) bus serves magnetic disk units via the disk file controller. Major peripheral units of ESS, including the man-machine interface known as the Master Control Center, are served by the peripheral unit bus.

The lower part of Fig. 1 shows the major units of the utility system.

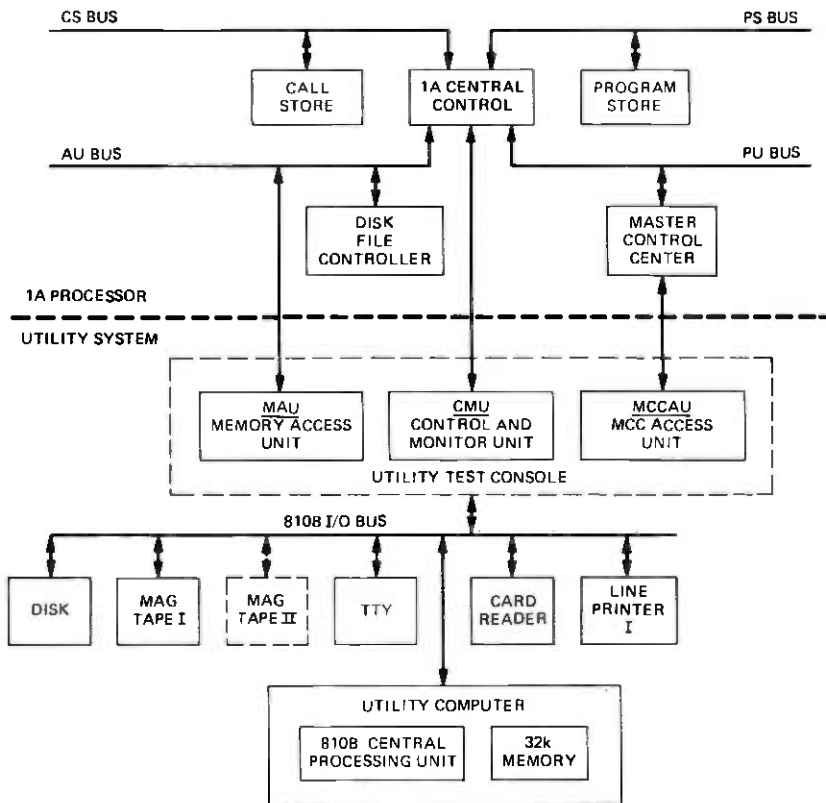


Fig. 1—1A Processor utility system hardware configuration.

The UTC serves as an interface between the 1A Processor and the SEL 810B computer.

The UTC consists of three functionally independent units: the Memory Access Unit (MAU), the Control and Monitor Unit (CMU), and the Maintenance Control Center Access Unit (MCCAU).

It is through the MAU on the auxiliary unit bus of the 1A Processor that the Utility Computer (UC) is able to read and write ESS memory locations.

The CMU contains registers which shadow a large number of central control locations. These registers in turn provide an input to matchers that can trigger a utility action when a predetermined condition exists. There is also a monitor store with capacity for 512 entries resulting from snapshots of the contents of processor locations. A single snapshot entry consists of a total of 528 bits of data. The data gathered on each snap consists of a fixed part and a variable part. The variable part is a set of up to four quantities chosen by the user from a list of 21 possibilities and specified in the job deck at the start of the run.

The MCCAU allows interrogation of processor status indicators and the setting of major controls on the MCC.

The utility computer has access to the subunits of the UTC via its I/O bus, which is time-shared with its own peripheral units. As shown in Fig. 1, the utility system peripherals consist of two disks (one with 2-megabyte capacity and the other with 6-megabyte capacity), a magnetic tape unit with an optional spare, a card reader, a line printer, and a typewriter.

The user input to the utility system is usually via the card reader, and normally the test results are printed on the line printer.

All utility system run-time actions occur as a result of a user-specified trigger. The trigger may either be a hardware trigger in the UTC or a software trigger resulting from planting a branch instruction at a specified program address which will interrupt the normal program flow and transfer control to the 1A Processor resident utility program. Following a hardware trigger, there may be autonomous actions such as a snapshot of processor locations, or, as in the case of the software trigger, there may be embedded actions requiring the execution of utility code in the 1A Processor. An example of an embedded action is the memory dump which moves a user-specified block of memory to space reserved for the utility system in call store. At the conclusion of the test, the contents of the utility call store, together with data in the monitor store in the CMU, are formatted and printed for the user.

As mentioned previously, run-time actions may be caused by a hardware trigger within the UTC. A variety of matchers have been provided which allow great flexibility in the choice of conditions that will initiate utility action.

The hardware matchers are provided in the CMU portion of the UTC. The matcher inputs are on the one hand a user specified quantity and on the other hand an address or data input from the central control.

There are 16 single address matchers. Since the matchers are independent of each other, independent utility actions may be specified for up to 16 individual addresses in a single test run. Twelve matchers may be set to match on either an instruction or data address residing in either the current program store address register or the data address register, respectively, of the central control. The other four may be set to match on a core address used in transfers to and from a unit on the AU bus (generally a file store). When a data address has been specified, the user has the additional option of qualifying the match condition to produce a trigger on a read, a write, or either a read or write operation.

Eight block address matchers have been provided. Six are for matching against instruction or data addresses obtained from the processor using the same register sources as the single address matchers. The other two are used to match against core addresses used in transfers to or from a

unit on the AU bus (generally a file store). As implied by the name, a block address matcher requires inputs to specify both the start and end address of the block. There are two selectable matching modes: the interface mode and the block mode. In the interface mode, a trigger is produced the first time the source address lies within the user-defined block or, optionally, the first time the address lies outside the specified block. A third option produces a trigger when either entering or exiting the address block. In the block mode, the user has the additional option of requesting whether the addresses of interest are inside or outside the block. In this mode, a trigger is generated and associated utility action results each time an address inside (or, optionally, outside) the specified address block is referenced.

Four 24-bit data matchers have been provided. These may be used to match against the contents of any of 16 central control registers. The match may be against all of the bits or against any combination of bits as specified by a masking quantity.

There are two counter matchers, each of which may count one of six user-specified count sources. The user specifies the source (e.g., processor cycles) and the value of the count which will produce a trigger.

An interrupt matcher is provided that produces a trigger each time a selected processor interrupt level occurs. Any combination of levels may be specified.

Finally, there are four multiple-condition matchers which will produce a trigger when a specified combination of any of these conditions occurs. The trigger and associated utility action will occur only when the specified combination occurs in the same instruction cycle.

### **3.3 Utility language**

Through the use of the utility language, the user has the ability to initialize the test environment, to control the execution of programs, and to specify the collection of test data. The statement is the basic component of the language, and there are two basic types:

- (i) Job control statements, which delimit the job deck and establish the operating environment
- (ii) Utility language statements, which determine the utility actions that are to occur during the test run.

All statements have a command field, except comment-only statements. In the case of the utility language statement, the command field is further subdivided into a trigger clause and an action clause. The trigger clause specifies the conditions that will initiate utility actions, and the action clause specifies the utility actions that are to occur. Conditional action

clauses based on run-time data are permitted. So that run-time decisions may be made to enable or disable a statement by name, a name label may accompany the utility language statement.

The basic purpose of job control commands is to establish the environment in which the test will be executed. The user may choose to accept the system in the state left by the previous job or may specify with considerable exactness the hardware and software configuration desired. Overwrites may be installed prior to the start of the test, and the user has the option of expressing them either in mnemonic terms or in octal.

Trigger commands specify the conditions that will initiate utility action. As explained previously, the trigger may be either a hardware trigger resulting from a specified condition detected by a hardware matcher or a software trigger resulting from a branch instruction planted at a specified program address. There are subtle but important differences between these two trigger types. The software trigger operates on the machine state existing just prior to the branch instruction and interferes with the program flow. Hardware triggers are completely autonomous in themselves, although the resulting utility action may not be. Autonomous action that may accompany a hardware trigger consists of either snapping key central control locations or initiating or terminating a trace action. The trace terminates either by command or when the trace counter is satisfied. A useful feature is the trace-last option whereby only a user-specified number of the last instructions (or branches) of the trace are output.

The action clause specifies the utility actions that are to follow a trigger. A variety of utility verbs may be specified in the action clause. Generally speaking, any combination of verbs may accompany any trigger. Specified values may be set into registers or memory locations. The contents of registers or memory locations may be saved for output at the end of the job. Other utility statements may be enabled or disabled. Program execution may be transferred to a user-specified address. Conditional action based on the current machine state is also possible. Instruction or transfer traces may be initiated or terminated. Messages may be input to ESS or sent to the system operator. This last action is useful when it is necessary to ask for manual action during the test.

A particularly useful capability available to the user is that of writing new utility functions and then invoking them during the test using any of the available triggers. This feature, known as the DOIT function, has enabled the users to meet unique needs which are not satisfied by the general utility verbs.

Pseudo operations do not directly generate any utility functions, but are a necessary part of the language to improve its convenience. Pseudo operations may be used to define symbols, to change default values, to



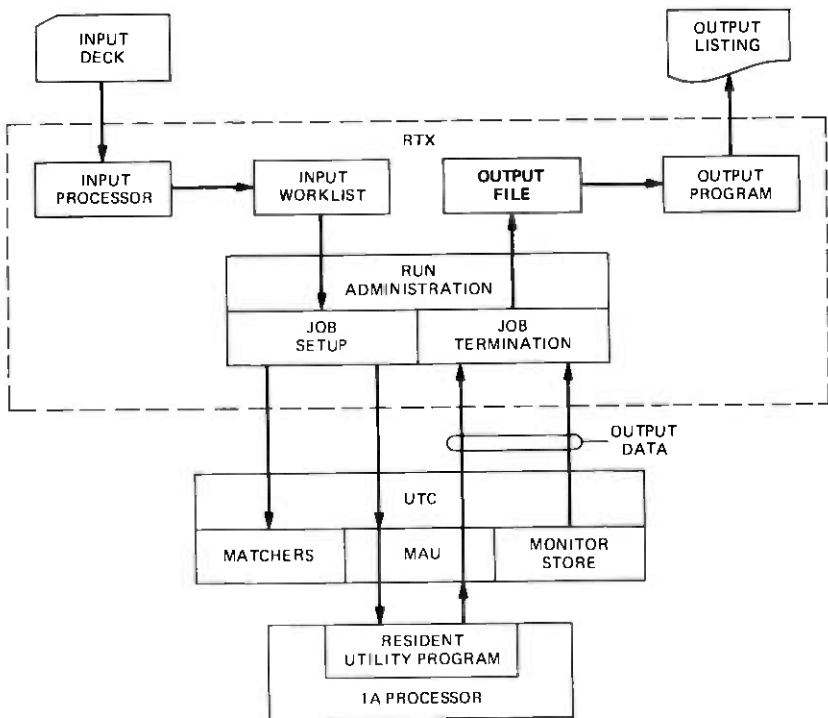


Fig. 2—1A Processor utility system software organization.

specify the start address and range of a set of overwrites, or to specify the use of a prestored set of utility statements.

### 3.4 Personal mode

Another mode of operation available to users is known as the personal mode. This mode gives the user the benefits of interactive operation; and though the personal mode commands are limited, they permit direct manual control. Typically, a user will input a set of commands for a conventional batch job and then invoke the personal mode. Through use of personal mode commands, the utility system may output any data immediately after it is collected instead of at the end of the test. Optionally, data collected may be erased, specified memory locations may be dumped, register or memory locations may be set to a specified value, or program execution may be transferred to a specified address or the test may be terminated.

### 3.5 Software organization

The software organization of the 1A Processor Utility System is explained through use of Fig. 2. The Real Time Executive (RTX) is a ven-

dor-supplied software monitor controlling the utility programs. The utility system software consists of four major programs: input processor, run administration, 1A Processor resident utility program, and output program. The input processor interprets the input and generates a worklist. Run administration sets up the job according to commands in the worklist, controls the job during execution, and collects the output data after the job terminates. During execution of the job, the 1A Processor resident utility program performs nonautonomous user-specified utility actions. The output program formats the data, outputs the listing and saves statistical data about the job.

The input processor program reads and interprets the input, generates a worklist corresponding to the input statements, and records the input as part of the information contained in the output file. Due to memory limitations, the input processor program is divided into three tasks which are executed sequentially. The first task reads the input and performs a syntactic scan of all statements. The second task performs a semantic check and builds a worklist for all job control statements. The third task performs a semantic check and builds a worklist for utility language statements. Run administration, the next utility system task, is activated by the input processor program.

Run administration is responsible for controlling the job during setup and execution, and collecting the data after the job terminates. Run Administration sets up the job according to commands stored in the worklist. Typical job setup functions include configuration and initialization of ESS, loading the 1A Processor system from magnetic tape, installing overwrites, initializing the UTC hardware matchers, installing branch instructions in ESS programs to implement the software triggers, and passing tables to the 1A Processor resident utility program for execution-time processing. During job execution, run administration times the job, responds to execution-time requests for activation or deactivation of hardware matchers, responds to job termination requests, and responds to user commands from the operator's console if the job is a personal-mode job. After the job terminates, data generated during the job must be collected. Autonomous traces and snaps are saved in the monitor store of the UTC. Data generated from nonautonomous action is saved in a call store dedicated to the utility system. This data is collected and merged in a time-sequential output file. Lastly, overwrites and software triggers are removed, and the original instructions are restored. Run administration is now completed and the Output Program is activated.

The 1A Processor resident utility program is entered after a trigger to execute any nonautonomous action. The program is primarily table driven, using tables passed by Run Administration. The work table contains encoded data about the specific requests for each trigger. An

entry vector table contains a pointer to the work table which was constructed for each trigger. For each software trigger, the return table contains the original instruction and the return address where ESS program execution will be resumed.

The 1A Processor has programs which are paged into core and overlay a portion of memory. Since utility requests may apply to paged programs, the 1A Processor resident utility program monitors the programs as they are paged and activates or deactivates the necessary triggers.

The output program formats the raw data from the output file, outputs the job listing, and saves statistics accumulated for the job. The addresses of the triggers are converted to a symbolic name plus displacement. Finally, the output program activates the input processor for the next job.

### **3.6 Administrative and support features**

As in any programming system, the versatility of the system depends greatly on support programs. The utility system is no exception and has numerous support programs which may be grouped into five functional types: loader map administration for symbolic name representation, overwrite administration, statistic gathering, message handling, and operator programs.

Program designers prefer to specify a program address as a symbolic program name (base address) plus displacement, rather than as an absolute number. Similarly, faster evaluation of the output is possible if the addresses are printed as a symbolic program name plus displacement. The loader process produces additional files on the loader tape for the benefit of the utility system. These files, called loader map files, are stored on the utility system disk. The input processor program references one file for name-to-address conversion. This file contains all program and entry names. Another file specifies the addresses of all the core-resident programs and a third file contains the addresses for paged programs. The output program accesses these latter two files for address-to-name conversion.

Overwrites account for a major portion of utility system work. An off-line incremental assembler accepts changes in the source language and produces the necessary overwrite data. This procedure greatly reduces the number of errors that might otherwise result. The next step requires that the utility system input the overwrites, allocate space temporarily, and install them for testing. If the overwrites are to be permanent, the utility system allocates the permanent patch space, updates the loader map files, and adds the overwrites to the loader tape.

The output program gathers various types of statistical information about individual test runs. Periodically, through operator request, the

statistical files are output. The statistics include the total number of runs, the number of successful (nonaborted) runs, the accumulated execution time, the accumulated number of input statements, and the accumulated number of output pages.

The utility system employs a message-handling program, whose purpose is to format all messages, to generate a message catalog, to control entering messages and data in the output file, and to maintain a log. Consistency for the utility system output is achieved, since a single program controls all messages and data formats for the output file.

The operator programs provide the operator with the capability to maintain, monitor, and control the 1A Processor utility system. The main operator program, activated by a manual switch, activates any of the other operator programs on request. The operator programs are grouped into administrative and operational programs.

The standard administrative operator programs may be used to initialize the utility computer from the utility system disk, save the disks on tape, reinitialize the disks from tape, update specified files on disk from tape, copy tape to tape, set time and date, and control diagnostic programs for the utility system main frame and peripherals. Special administrative operator programs build or print the statistical files, copy the loader map files from tape to disk, update the loader map files, log permanent overwrites, log the loader tape, and control diagnostic programs for the UTC.

The operational operator programs may be used to activate the input processor program, to terminate the active job, and to cancel any utility system job in progress. A widely used operator program serves as a partial loader. This program temporarily loads any new version of a 1A Processor program from the assembled output module. Other operator programs can dump 1A Processor memory locations or compare the 1A Processor memory with the loader tape. Lastly, since the 1A Processor resident utility program is relocatable, a parameter table defines the addresses of the relocated symbols. This parameter table also contains the operational status of all hardware matchers. An operator program can print or change the value of any parameter in this table.

### **3.7 Commentary**

A convenient and efficient debugging tool has been provided by the 1A Processor Utility System. The batch mode of operation has resulted in a throughput of about 20 jobs per hour with jobs varying in duration from seconds to several minutes. A functional test language has provided the user with a convenient language by which to describe and control tests. Symbolic representation of loader process names has been made available to both input and output. Autonomous tracing and snap-

shotting has satisfied the noninterfering requirement. The overwrite process has been widely and successfully used.

In regard to deficiencies, the memory limitation of the minicomputer has hampered the design, implementation, and throughput of the utility system. The input processor requires three separate tasks to complete its function. Overlapping execution of the input processor, run administration, and output programs to increase throughput can not be realized because of the same memory limitations. Greater use of symbolic representation would be highly desirable; but to accomplish this, access to the common data pool (known as COMPOOL) would be required. Because of the size of COMPOOL (about 100,000 symbols), implementing a capability to reference COMPOOL is not practical in the minicomputer.

To overcome most of these deficiencies, implementation is underway to link the computation center facilities to the minicomputer by means of a data link. As a consequence, the input processor and output programs would execute on the full-scale computer facilities that are available. Only run administration would remain in the minicomputer. It would get the input worklist via the data link from the input processor and in a similar fashion would pass the output file to the output program. This linking to full-scale computation facilities will give the user access to COMPOOL, will allow application of the full power of a high-level language, will allow the loader to install permanent overwrites using source language statements, and finally will improve the test throughput of the ESS laboratories by a factor of two to three.

#### **IV. AN AUTOMATED MODEL-REFERENCED TESTING TOOL**

##### **4.1 *Model-referenced testing***

The first step in developing systematic testing is to determine the standards against which the test is being verified or the behavior model of the process that is being realized by the software. A model at a level more detailed than the rather broad system requirements and guidelines is required. Software testing should not only detect deviations from the modeled behavior but also pinpoint the component causing such deviation.

A fairly accurate and formal class of models is frequently available for software processes, since coding is seldom done directly from the broad system requirements. The operation of the software system in processing a set of external stimuli is typically modeled by a series of small steps. Each step in the model consists of a decision concerning the classification of some external stimuli and/or internal variables followed by a reaction of the software process to the stimuli, external or internal. This model can be easily formalized into a directed graph where nodes represent the decision or stimulus classification points and directed arcs,

representing the outcome of such decisions and the ensuing reaction of the process, join the nodes. A flowchart is a familiar example of such a model; the state diagrams used for No. 4 ESS call-processing functions constitute another set of examples. Further information may be imparted to the model by specifying logical constraints imposed on the process by the physics of the problem. For example, if two classes of stimuli are mutually exclusive in the physical world then the model should exhibit no decisions that classify them into one set. This information is not readily expressed by the adjacency relationships in a graph and needs to be supplied additionally.

The Automated Testing and Load Analysis System (ATLAS) was developed in response to the need to systematically test the complex software processes of the No. 4 ESS. ATLAS provides a means for testing software processes that may be modeled formally by such a directed graph with optional logical constraints imposed upon the order and sequence of arc traversals. Entry (exit) nodes in the graph model usually correspond to the software process getting (relinquishing) control or being initiated (terminated). Every path in the graph that starts at an entry node, terminates at an exit node, and is admissible according to the logical constraints on arc traversals, is composed of a sequence of arcs that may be interpreted as a sequence of steps. Each step describes a stimulus-decision-action-transition sequence. This interpretation of the path description states that if the sequence of stimuli contained in the path is applied to the process modeled by the graph, then the accompanying set of actions must take place. If the stimulus-action descriptions are written as directives to a test driver for generating the prescribed stimuli in the sequence enumerated and for verifying the resultant actions, a test for the software is generated from the model.

The two major functions of ATLAS are to identify the admissible test sequences from the model and to apply the identified tests to the software. These functions and their ancillaries are described in the next two sections. No. 4 ESS application experience of ATLAS as well as the merits and demerits of ATLAS are also given. The interconnection of the components of ATLAS is shown in Fig. 3.

#### **4.2 Test Identification**

A directed graph model of the software to be tested is the starting point for the ATLAS process. Some logical propositions are added to the description of the model to express the inherent logic of the physical process being realized by the software. Addition of test commands which cause test drivers to initiate and monitor test actions associated with each arc of the graph makes the model ready for processing by ATLAS.

The first active step in ATLAS processes the model for the identification of admissible tests that are not only consistent with the adjacency

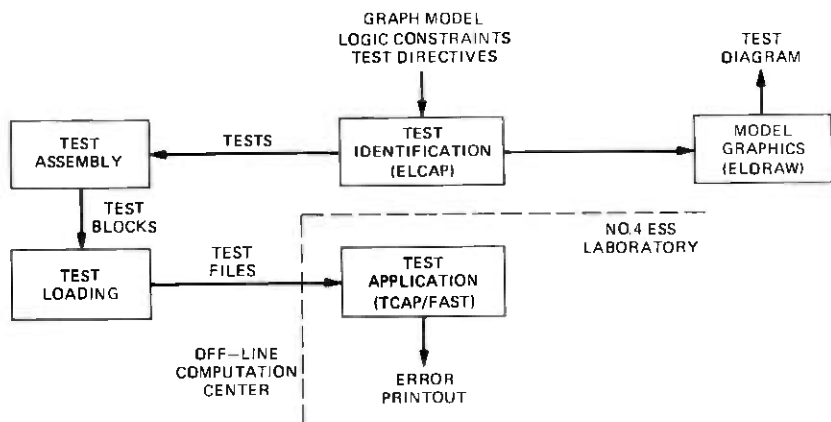


Fig. 3—Components of ATLAS.

of the graph but also satisfy the model logic. This is equivalent to finding paths in the graph that begin at the entry nodes and terminate at the exit nodes and satisfy the logical constraints of the model. The set of all such paths is an exhaustive set of tests for the software as stipulated by the model. This set may be quite large in size for models of even moderate complexity. A smaller set of tests, called the cover set of tests, may be selected from the paths identified. The cover set has the property that every arc of the model is traversed in at least one path of the set. This smaller set of tests is useful for checking the software under test against gross deviations from the model.

The Enumerator of Loops, Covers, and Admissible Paths (ELCAP) is the component of ATLAS charged with the identification of tests from the model. After a thorough syntactic check of the encoded model, ELCAP uses graph-theoretic algorithms to identify the set of all admissible paths in the model and subsequently selects a suitable cover set. Since the presence of loops in the model implies potentially infinite test sequences, ELCAP also identifies all such loops and warns the user about their presence. The traversal of such loops in any admissible path may then be constrained to be finite. The test commands associated with the arcs of each path are collected sequentially and provide a complete specification for the test identified by the path.

The ATLAS component that may be used to draw the encoded model is a computerized graphic process called ELDRAW.

#### 4.3 Test application

The tests derived from the model may be executed in many modes on the actual or simulated software depending upon the test facilities available. These facilities may be the 1A Processor, simulators, or test

drivers constructed specifically for the software under test. The Test Call Application Program (TCAP), a component of ATLAS, is one of the first such test drivers constructed for testing No. 4 ESS call-processing programs. Another component called Facility for the Application of Software Tests (FAST) extends the principles of TCAP to non-call-processing software of No. 4 ESS. Both are ESS resident programs that use the facilities of No. 4 ESS to apply and to monitor tests for the software in as natural a manner as possible.

TCAP has the capability of emulating various sources of stimuli that are recognized by the No. 4 ESS call-processing system. Software mechanisms enable TCAP to secure control before and after these stimuli are acted upon by the software under test so that TCAP may verify the accuracy of these actions against those specified in the test. Abnormal conditions, such as resource blocking, are also simulated by TCAP.

Commands associated with a test are presented to TCAP as a data block, called a test call block. Entries in this block generally contain command codes followed by command operands and are interpreted by the routines of TCAP. The execution of a test via TCAP may be described as follows. The initialization data provided in the test call block enables TCAP to secure the required trunks for the test to be executed. TCAP assumes control and generates stimuli on the remote ends of these trunks. Other necessary initialization is also done. Testing is then started by generating the first stimulus, generally a call origination prescribed in the test call block. When the stimulus is reported to the operating system of ESS, a built-in software mechanism transfers control to TCAP. TCAP checks the accuracy of the report, sets up linkages so that control may be returned to TCAP after the software under test has had a chance to process the stimulus, and then invokes the call-processing software. When control is regained by TCAP, verification commands in the test call block direct TCAP to monitor the accurate processing of the stimulus. The cycle is repeated for each stimulus until a test termination command is reached or the software is detected to behave differently from the pattern given in the test call block. The test is then terminated with the appropriate success or failure message. The system is subsequently purged of the test.

TCAP is capable of generating a large number of tests in parallel. Various other capabilities, such as timing and manipulation of special call-processing data and control structures, are also provided.

The Facility for the Application of Software Tests (FAST) extends the basic pattern set by TCAP to other No. 4 ESS software. The requirements are the same in both cases; generation of stimulus in a natural fashion, emulation of abnormal conditions, verification of response, and a smooth control interface between the operating system, the test driver, and the programs under test. Due to the wider scope of FAST, the control inter-



faces are not built into the system. They are generally realized via 1A Processor Utility System commands. FAST provides some general functions, e.g., stimuli generation, data initialization, timing, simulation of some systemwide abnormal conditions, and general data verification. It also provides for linkages to any special-purpose routines the user may generate so that the capabilities described above may be customized for the application at hand.

Two specialized command languages, one for each of the two test drivers described above, are incorporated in ATLAS to give the user the capability of writing test commands in a symbolic and easily readable format. The constructs in the languages are chosen to be particularly relevant to the application. The languages have declarative statements that permit the associated test drivers to initialize tests as well as executable statements that actually direct test actions. Since tests written in these languages consist of strictly predetermined sequences of stimulus-action pairs, conditional statement facilities are not provided. The conditional aspects of the tests are handled in the identification stages by ELCAP.

Assemblers, constructed with the macro handling facilities of SWAP, the assembler for ESS programs, are provided in ATLAS for resolution of symbols. These assemblers use the same symbol dictionaries (COM-POOLS) that are employed by the program assemblers so that ATLAS users may maintain uniform symbol definitions in both the coding and testing phases. The assemblers allow for initialization declarations to be placed in the model where deemed relevant by the user. They also bind tests to models via test and arc identifiers to facilitate the administration of tests and to resolve errors identified by the tests. A loader handles the movement of high volumes of test data generated in the computation center facilities (where parts of ATLAS are resident) to the No. 4 ESS Laboratories where TCAP and FAST are resident.

#### ***4.4 Usage data and summary***

The components of ATLAS have been used in various combinations to test over 40,000 instructions in No. 4 ESS software. In the majority of cases, ATLAS was used as a testing tool during the phase of integrating various component programs of a software function. The call-processing system has used ATLAS extensively. In all cases, the programs had been tested by using nonautomated methods before ATLAS testing methods were employed, though only a small set of cover tests were run rather than an exhaustive set of tests. The incompleteness and inconsistencies of the models constituted a large class of errors. However, about 300 problems were found in the pretested software during ATLAS testing, excluding the model errors. The errors ranged from hardware problems

that surfaced because of the intense mode of testing by the test drivers to software logic and implementation errors.

The primary benefit derived from ATLAS is the formalization of the testing process. Programmers are relieved from the very detailed, tiresome, and error-prone work of specifying how to test their software so that more thought may be given to the fundamental question of what needs to be tested. The repeatability of the tests, testing in a natural environment, documentation, and ease of test administration are other benefits that are directly attributable to the formalism imposed by ATLAS.

Since ATLAS is a model-referenced testing method, the excellence achievable by ATLAS testing is inherently dependent upon the quality of the model. If the model is incomplete in reflecting the requirements of the software, the set of tests derived from it will also be incomplete. Furthermore, since ATLAS cannot distinguish between a model error and a genuine program bug, model inconsistencies will generate spurious error data. These facts imply that the model should be carefully developed and checked for completeness and consistency for ATLAS testing to be fully effective. The use of a formal approach always needs more thoughtful planning than its ad hoc counterparts. Lead time to do this thinking and planning is needed. The direct computer charges associated with the use of an algorithmic testing approach such as ATLAS can be expected to be high. The software product of the No. 4 ESS project is designed to have high reliability and is expected to undergo considerable change over the years as new capabilities are added to the system. The benefits of formalism and discipline inherent in the use of ATLAS far outweigh its disadvantages.

## **V. PROGRAMMED ELECTRONIC TRAFFIC SIMULATOR**

### **5.1 System description and environment**

The stress testing and evaluation of large real-time software systems present unique problems. System performance under load is difficult to measure and is difficult to simulate. Stress testing of an Electronic Switching System (ESS) is no exception.

Previous ESS traffic simulation methods have employed electromechanical load boxes and software simulators with simulated calls applied to the network terminals. This approach required large quantities of per-terminal equipment and a network with associated peripherals large enough to process the high traffic loads. For large systems with high call capacities, terminal simulation processors cannot be economically justified because of both the dollar expenditure for all of the required equipment and the space limitations of laboratory test models. Thus, in order to apply a realistic traffic load to a No. 4 ESS in a laboratory

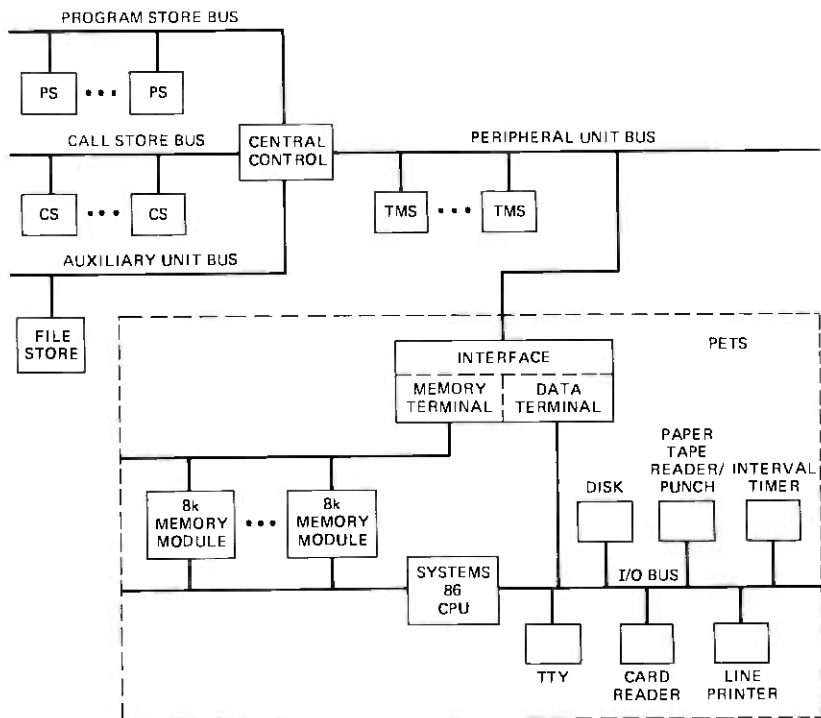


Fig. 4—PETS system.

environment, a new approach was necessary. The search for a new approach led to the development of PETS—the Programmed Electronic Traffic Simulator.

The basic objectives of the simulator system are the following:

- (i) Generation of traffic levels that exceed the designed or engineered capacity of the system.
- (ii) Interconnection with ESS via the Peripheral Unit (PU) bus.
- (iii) Generation of random traffic loads that are realistically distributed in time.
- (iv) Generation of traffic that is independent of the available ESS peripheral hardware.
- (v) Operation in real time.
- (vi) Operation without requiring any significant modification to the ESS program.
- (vii) Collection of sufficient meaningful data to allow evaluation of ESS performance.

To meet these objectives, the simulation system illustrated in Fig. 4 was developed. The PETS system consists of a commercial computer, a hardware interface which connects the simulator to the peripheral bus

system of the No. 4 ESS, and a memory and data terminal which enable the interface to communicate with the commercial computer. These components are discussed in more detail in subsequent sections of this paper.

PETS simulates most of the environment external to the ESS. To accomplish this, the peripheral equipment action that would result from either connecting office stimuli or the peripheral equipment action that would result from the ESS response must be generated. The outputs from the ESS processor to the peripheral system must be intercepted and analyzed. These outputs are compared with expected responses to analyze the performance of ESS.

PETS does not replace all of the peripheral equipment in a laboratory but supplements the existing laboratory peripheral system by making it appear to ESS that it is working with a large office.

To meet the stated objectives, the system described in the following paragraphs was developed. The type of peripheral equipment simulated includes the Signal Processor (SP), the Time-Slot Interchange (TSI), and the Terminal Access Circuit (TAC). Because the system laboratories are equipped with the necessary Time-Multiplexed Switch (TMS) configurations, this portion of the network does not require simulation.

To simulate the proper environment, both the PETS and the No. 4 ESS data bases must identify the peripheral equipment of a large office and must contain corresponding translation and parameter data. The real peripheral equipment in the laboratory may be defined in the ESS data base along with the simulated equipment. This allows real and simulated traffic to be simultaneously presented to the ESS processor.

The amount of memory required for the PETS system depends, in part, on the number of simulated circuits. Each simulated terminal, whether it is a trunk or service circuit, requires a dedicated block of storage in the PETS memory. To keep the memory required for PETS manageable, a relatively small number of terminals is simulated instead of a full-size network. In addition, the simulated terminals are evenly spread over the simulated equipment.

To achieve the large traffic volume necessary to adequately test the ESS, the per-terminal origination rate must be higher than the per-terminal origination rate of a working office. Thus, to provide a high calling rate using a reduced number of terminals, shorter holding times or "talking" intervals are necessary. However, realistic distributions of other call timing intervals are maintained.

PETS has the capability of simulating a maximum of eight SPs having both universal and miscellaneous scan and signal distributor points and eight additional SPs having only miscellaneous points. The additional miscellaneous points are required for the MF transmitters and receivers for maximum MF traffic loads.

The PETS hardware simulates a maximum of five TSI frames and four TACs with 16 terminals each. Only the TAC low-priority and high-priority message buffers are simulated.

Communication between the No. 4 ESS processor and the PETS system is via the peripheral unit bus system. This system consists of the enable address bus, the write bus, the reply bus, and the control bus. In addition, there is a communication link between the utility test console and the PETS system for start/stop control in either direction.

A hardware interface unit provides for the connection to the ESS peripheral unit bus system and the transfer of data into and out of simulator memory.

The computer used in the PETS system is the SEL Systems 86. In addition to the normal peripheral configuration, the Systems 86 is equipped with a special memory terminal and data terminal. The memory terminal is utilized by the interface to gain access to the Systems 86 core memory. The data terminal provides for the transfer of status information from the interface to the PETS software system.

To provide for the transfer of data between the interface and the PETS programs, two-port memory is utilized. This allows the interface to retrieve data directly from dedicated areas of memory and to store data necessary for processing by the software. The programs, in turn, can update status information utilized by the hardware interface. The area of core accessed by both the interface and the programs is referred to as "shared memory."

Control and status information is exchanged between the interface and the programs via the data terminal. The control functions of starting and stopping the systems are provided through the use of the data terminal.

PETS must provide external stimuli to ESS such as originations, dialed digits, answer, and disconnect signals. To accomplish this, the simulation program enters data into the SP buffers and updates simulated T-bits and scan points in shared memory. When ESS transmits an order to read a buffer, T-bit data, or scan points, the interface intercepts the order, retrieves the data from the proper locations of shared memory, and returns the data to ESS.

In response to the stimuli, the ESS processor sends orders to the peripheral equipment to connect and disconnect network paths, output pulse digits, etc. The interface intercepts these orders, provides the proper response to ESS and stores appropriate data in a common buffer area of shared memory. The simulation program periodically processes the data in the common buffer. The simulated portion of any equipment affected by the order is updated in the proper time sequence and appropriate additional stimuli are provided to ensure the progress of the simulated call.

The simulation program provides a random distribution of originations that approximate those encountered in a working office. In addition, realistic timing variations are provided for the various interoffice signaling stimuli and responses.

Call characteristics, such as permanent signal, partial dial, call abandon, and originator on-hook, are provided on a random basis based upon the parameters specified for a simulation run.

The simulation program keeps records of the traffic presented to ESS and the disposition of that traffic. Records are kept of all originations, terminations, anomalies and unexpected events. In addition, histogram data is maintained for the time delays between the various stages of call setup. The data are used to evaluate the performance of the ESS under load.

The number of terminals in the simulated office is dependent upon the desired load, the average call-holding time, the amount of available ESS call store, the amount of available PETS memory, and the number of service circuits required for the traffic load. The ESS call store requirement is dependent upon the maximum load and the number of simulated terminals. Based, in part, on these considerations, the system laboratory is engineered and the Office Data Assembler (ODA) is utilized to provide the ESS translation and parameter data required for the simulated peripheral equipment.

A data assembler produces the translation data base needed by the PETS system. The input to the data assembler will be derived from the same input used by the ESS ODA.

The translation and parameter data must be supplemented with additional data describing the call types to be included in the simulation run, the desired load characteristics, the various call anomalies to be simulated, etc. Much of the data is processed and built off-line. Thus, very little run-time processing is necessary.

## **5.2 Hardware configuration**

The hardware interface provides the communications path between ESS and the PETS system. The functions of the interface are to:

(i) Recognize when ESS sends a peripheral order to the simulated equipment, extract the appropriate data, and gate the data to the software system.

(ii) Process, via hardware, those orders requiring an immediate response to the ESS.

(iii) Provide the necessary timing and control functions to properly transfer or exchange data between the ESS PU bus and the Systems 86 Central Processing Unit (CPU).

(iv) Provide logic level shifting and data formatting for efficient

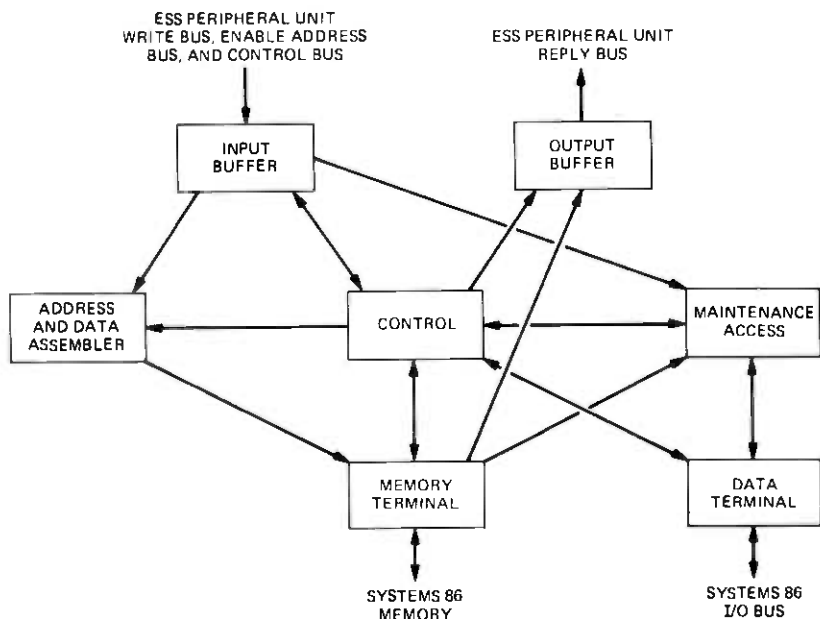


Fig. 5—PETS interface.

transfer of information between the ESS PU bus and the Systems 86 CPU.

(v) Generate, upon demand, signals to control both the ESS processor and the PETS system clock.

(vi) Provide a visual display of the interface status and report all anomalies in operation to the software system.

A block diagram of the interface is shown in Fig. 5.

The PETS interface with ESS is at the PU bus. Modified ESS cable drivers and receivers are employed. The modifications provide the necessary interface between the +1-volt logic levels of the ESS circuits and the +5-volt logic levels of the TTL circuits employed in the interface.

The interface processes those peripheral orders accompanied by the address of a simulated peripheral frame. When the address match is detected, the information on the PU write bus and enable address bus is processed. The interface also responds to the pulses on the PU control bus. These pulses are stored in the control register until the interface has appropriately responded to the ESS processor.

Responses from PETS to ESS are transmitted over the PU reply bus. The interface generates a response which is identical to replies from the real peripheral frames. An all-seems-well pulse and a parity bit are also transmitted as part of each response.

The exchange of information between the interface and the Systems 86 CPU is accomplished via two individual channels of communication. The memory terminal enables the interface to access any part of the Systems 86 memory and provides the capability to read or write as many as 64 bits (double-word) of data. In addition to double-word transfers, the memory terminal provides the capability of word, halfword, byte, and bit transfers.

The interface must provide the memory terminal with an address compatible with the Systems 86 memory address format. This address is derived from three sources:

- (i) Information received by the interface from the peripheral unit bus.
- (ii) Hardware address counters in the interface.
- (iii) Information obtained from a previous memory-read operation.

Approximately 24K words of Systems 86 memory are utilized as shared memory. The interface accesses this memory to obtain the data associated with the simulated equipment and to obtain address pointers and indices necessary to locate the data in shared memory.

Since PETS is designed as a load simulator and not as a maintenance tool, no faults or other hardware anomalies are simulated. Thus, the interface simulates responses to most of the operational orders and a small subset of maintenance orders associated with the simulated equipment.

The simulation of peripheral orders may require as many as five separate accesses of shared memory. For example, on a "read SP buffer" order, the interface must first read the dedicated location associated with the particular SP and buffer type to obtain the pointer to the next buffer entry to be unloaded. These data are then used to read the particular buffer entry into a register in the interface. The interface must then zero the memory location where the buffer entry was previously stored. Next, the interface must update the address pointer and write the new pointer into the dedicated address. Finally, the buffer entry is gated onto the reply bus for transmission to ESS.

The PETS interface operates asynchronously from both the ESS CC and the Systems 86 CPU. The timing and control functions are derived in the interface from data provided by the ESS central control and the Systems 86 CPU. The basic clock signal used by the interface is derived from a clock signal transmitted from the Systems 86 CPU via the memory terminal.

The interface detects certain types of failures during processing of peripheral orders and generates an interrupt through the data terminal



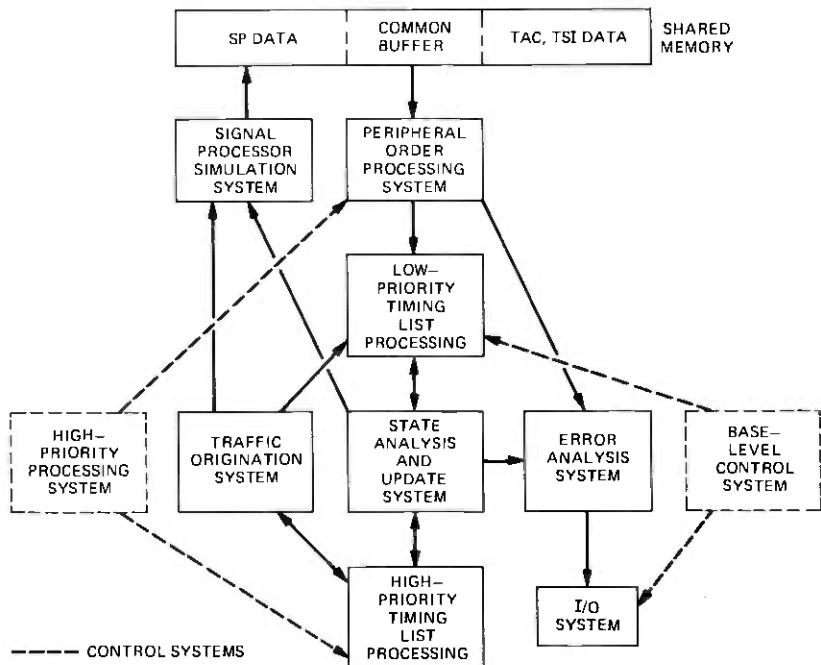


Fig. 6—Functional diagram of PETS software.

to notify the software system of the problem. The types of failures detected by the interface are:

- (i) An attempt to address nonpresent memory.
- (ii) A parity error occurring during a read operation of shared memory.
- (iii) An attempt to address a protected area of memory.

The cause of the failure is determined by the software utilizing test instructions.

### 5.3 Software description

The PETS software consists of many systems—each performing a generalized function. Figure 6 is a functional diagram of the software illustrating the various systems, with the dashed lines indicating the control programs. The following paragraphs describe the systems shown in the figure.

Data associated with the receipt of ESS peripheral orders are stored by the interface in a common buffer in shared memory. The peripheral order processing system periodically unloads this common buffer and processes each entry. Most orders are processed immediately. However,

in the case of network orders, processing is deferred until a complete ESS sequence (e.g., connecting an originating MF trunk to a receiver) has been received.

The signal processor simulation system administers the four buffers associated with each SP. This system provides the loading of these buffers with the various reports required during the simulation of a call and provides the ability to temporarily store a report if the buffer does not presently contain an empty slot.

The PETS software system is organized around two levels of processing—high-priority and base-level. The high-priority processing system utilizes a hardware timer that provides an interrupt every 5 ms. This interrupt causes the system to be entered for unloading the common buffer and the processing of peripheral orders. All high-priority tasks requiring attention are also processed during this interval. At the conclusion of high-priority processing, control is returned to base-level processing at the point of interrupt.

The base-level control system provides the basic control for the continuous cycling of the base-level tasks including base-level timing list processing and I/O control. To ensure that PETS is providing proper timing resolution, the length of the base-level cycle is continually monitored to ensure that it does not exceed specified limits.

The timing list processing system provides the real-time breaks between segments of a call simulation and provides the ability to measure various events occurring during a simulation run. Use of both high-priority and base-level timing allows great flexibility in the amount of time requested and the precision assigned to each request.

The state analysis and update system analyzes the various ESS responses or lack of response, determines the next state of the call simulation process, and provides ESS with any necessary stimuli.

The traffic origination system determines the time interval between call originations and generates Poisson traffic based upon data supplied by the user. This system determines the originating trunk and makes various call simulation decisions (e.g., abandon, permanent signal, etc.) associated with the call and presents the stimuli to ESS to accomplish the origination.

All error messages, status information, and user communications are processed by the I/O system. This system has the ability to process input commands and to supply the necessary data and communications without stopping or interfering with the simulation run.

All unexpected ESS responses are processed by the error analysis system. Data is recorded for output and all necessary action is taken to recover from the situation.

The philosophy for the software design is based upon three major concepts: the call register, timing list processing, and state analysis.

Keeping in mind the previously described systems, a general description of the PETS software follows, beginning with the call register.

Every simulated call in the system has a 6-word block of memory associated with it for the duration of the call. The call register contains the up-to-date status of the simulated call including the terminals involved, decision information to guide the progress of the call, and an indication of all ESS actions or responses received since the call register was last processed. Call registers are seized from an idle list during the origination and released after the call has been completely processed. During actual simulation, the call register will be linked to a timing list waiting for appropriate ESS actions or for a time-out to occur before presenting the next stimuli to ESS.

Each call register has an associated state stored as an item in the call register. The state value determines which program will process the call register after either the last requested time delay has elapsed or when ESS has taken some sort of action that involves the call. If an ESS action is involved, the processing of the call register will take place during base level; if a time-out occurs, the processing will take place in either high-priority or base level depending upon the timing precision required.

From an examination of the data contained in the call register, it can be determined what terminals are involved, how far the call has progressed, what ESS action has been received, what state analysis program to use to process the call register, and what anomalies, if any, are to be simulated as part of this call.

Numerous real-time breaks and timing measurements are required for each simulated call. Two timing lists are utilized. Base-level timing lists provide a wide range of timing applicable for most situations; high-priority timing lists provide greater precision for those events requiring critical timing but at the expense of being processed during the high-priority interrupt cycle.

Each timing list processing system consists of four timing list tables. The first table of the high-priority timing list is examined every 5 ms during high-priority processing. That is, every 5 ms the next slot of the first table is examined and all call registers linked to that slot are analyzed and processed.

Every time the processing of the first table is recycled (i.e., slot 0 is processed), the next slot of the second table is processed. Similarly, when the second and third tables are recycled, the next slot of the third and fourth tables, respectively, are processed. Thus, with four tables with 16 slots each, up to 5.461 minutes of high-priority timing in 5 ms increments can be provided.

Base-level timing differs from high-priority timing in that the processing is done during base level, the increment of time between slots on the first list is 20 ms instead of 5 ms, and the first list is 64 slots in

length instead of 16. The base-level timing list tables provide up to 1.456 hours of timing in 20-ms increments.

State analysis is a set of programs that analyze a call register to determine how far the call has progressed and what action is necessary next. Each call register has an associated state, and it is this state value that determines which program to use to process the call register. A state analysis program is entered as a result of processing or examining a call register and determines what stimulus, if any, is required for ESS or what time delay is necessary before taking any further action.

Functions other than those associated with the processing of a call can be provided by using special registers with states corresponding to the appropriate function.

Traffic originations are the result of processing special registers. When one of these traffic origination registers is processed, a traffic origination program is entered to generate a call and to make a decision, based upon the specified calling rate, as to when the next origination should occur.

Each of the traffic origination registers contains a calling rate that is currently in effect. The processing of another set of special registers causes the calling rate stored in the traffic origination registers to be periodically updated. Thus, a given calling rate curve can be approximated by a table containing 256 equal time intervals, each of which has a constant calling rate corresponding to the average over the interval. At the completion of each time interval, the next calling rate is retrieved from the table and stored in the traffic origination register. All traffic originations, selection of call characteristics, and trunk selections are based upon random number generators. Thus, the PETS system approximates user-specified values for the various call characteristics on a "random" basis.

A monitor system provides user communications prior to the actual start of the simulation run, provides the initial linkage of all call registers to the idle list, controls the PETS-ESS start/stop communications, processes all interrupts resulting from abnormal conditions (e.g., parity failure, nonpresent memory, etc.), and controls the periodic (5 ms) interrupt. The monitor system also controls the starting of the simulation run and the duration of the run.

#### **5.4 Data accumulation**

During a simulation run, the system records various events for analysis of ESS performance. The type of data collected is described in the following paragraphs.

Originations are recorded on an originating-trunk-type basis and on a trunk-subgroup basis. An origination is recorded when the initial SP buffer entry is made. Terminations are recorded at both the path res-

ervation stage and at the path setup stage of a call and are recorded according to the type of call and on a trunk-subgroup basis.

Counters are incremented as the simulated calls progress through the various stages and paths of call processing. In addition to recording completed calls, counters are incremented as each call anomaly (e.g., abandon and permanent signal) is simulated.

In order to evaluate the performance of ESS, PETS records those events related to the resources available to ESS. Counts are kept for all connections to tones and announcements, for events that indicate no available MF receiver or ESS call register, for failure to connect to a tone or announcement when one is required, for overflow of SP simulated buffers, and for similar items reflecting on ESS performance and the engineering of the ESS system.

Evaluating the performance of ESS requires measurements such as incoming delay, seizure time, address time, response time, answer time, and disconnect time. These six critical timing measurements are vital indicators of ESS performance. In order to record useful data and to display the data in meaningful form, these critical timing measurements are accumulated in histogram form. Each histogram consists of 16 elements of equal interval.

The PETS software system is in itself a complex real-time operating system with data that must be engineered to correspond to the simulated office environment. The system monitors and records its own base-level and high-priority cycle times, the number of times no trunk or no call register was available for an origination and the number of times the system was not able to cause ESS to properly idle a trunk.

PETS provides a periodic traffic summary in matrix form indicating the number of originations on a trunk-type basis and the disposition of those originations (e.g., completions on a trunk-type basis, abandons, and connections to tones and announcements). This traffic summary is output every minute unless the user specifies a different interval, and each summary presents the data for the last time period.

Because of the large volume of data collected during a simulation run, it is neither desirable nor practical to output a hard copy of this data during the run. The PETS system provides the ability to periodically dump the data to a disk file for analysis at the conclusion of a simulation run. A comparative analysis is available to print the data associated with each period and to compare one period with another. The data are presented in a well formatted dump indicating the counter description, its internal mnemonic, and the associated value.

### **5.5 Application results**

Providing for the testing and evaluation of ESS performance is a major goal of the PETS system. In addition, the simulation system is designed

to be used as a tool for supplying background loads for the debugging, integrating, and testing of ESS programs and systems. Some of the many applications and results are discussed in the following paragraphs.

In June, 1975, PETS was used to experimentally verify that No. 4 ESS could process its design objective peak busy-hour call load of 385,000 switched calls (431,000 total call attempts) while meeting all speed-of-service criteria. During the experiment, all normally scheduled maintenance tasks were performed as well as traffic administration and network management functions. Service circuits and transient call memory were engineered for the expected load. The results of this experiment demonstrated overwhelmingly that No. 4 ESS met and generally exceeded the advertised busy-hour capacity.

As a result of the initial experiment, further experiments were made that verified a new advertised peak capacity of No. 4 ESS of 550,000 switched calls (616,000 total call attempts) per busy-hour.

In addition to providing capacity verification results, PETS is a valuable tool for providing background loads for system testing. It has provided background loads exceeding 630,000 call attempts per hour while testing and evaluating the recent change system, the network management system, the traffic measurement system, and trunk maintenance procedures.

With background loads, more exhaustive testing of No. 4 ESS was possible, and means were available to compare internal ESS data with corresponding data in the PETS system.

PETS has been used to provide various loads to No. 4 ESS while measuring different aspects of performance. Many performance measurements such as base-level cycle times were taken and many logic and programming errors were uncovered during this load testing.

The present real-time capacity of the simulator is in the range of 680,000 call attempts per hour. It is quite possible that this capacity could be significantly increased by eliminating much of the internal data collection and by streamlining some of the more frequently used programs.

The PETS concept of load simulation has been successfully applied to the development of No. 4 ESS. The same concept with its associated rewards can be applied to other real-time software systems.

## **VI. CONCLUSION**

Large systems such as No. 4 ESS, which evolve as new capabilities are added over a long period of years, must continue to have support systems like those described in this paper. In fact, the support systems themselves must continue to evolve in order to meet the ever-increasing challenge arising from the need to produce a high-quality product under more and more difficult circumstances. Difficulties arise from the need

to support multiple generics which are inherent to the evolution process, the attendant growth in size, and the need to provide support to the field.

As we look to the future in No. 4 ESS support systems development, we see a three-pronged approach: (i) The scope of support must be widened to include all aspects of software development from the determination of requirements to the administration of code in the field. (ii) The individual support systems must be integrated into a network of systems which allows each process access to the necessary data of the other processes and which provides the user with a single, uniform interface. (iii) Such a network of systems must facilitate the introduction of new processes and allow the modification of those that already exist.

In summary, the four support systems described herein have significantly assisted the development process which produced the initial versions of No. 4 ESS. Together with No. 4 ESS, these support systems themselves will continue to evolve in order to provide new capabilities and to meet the demand for greater development efficiency.





## **No. 4 ESS:**

# **System Integration and Early Office Experience**

by H. B. COMPTON, P. K. GILOTH, M. B. HAVERTY,  
and B. G. NIEDFELDT

(Manuscript received July 12, 1976)

*This article describes the special equipment and test procedures developed for system-testing the No. 4 ESS toll system. It discusses novel methods developed for the coordination of hardware and software changes and testing in various system laboratory configurations. Planning, engineering, installation, and testing of early offices are described. Early field experience with the first operational No. 4 ESS is presented. This office, a selective routing tandem located in Chicago, went into service on January 17, 1976.*

## **I. INTRODUCTION**

A No. 4 Electronic Switching System was installed in Chicago to provide toll service between metropolitan Chicago and selected area codes in the United States. The new system was cut into service on January 17, 1976. During a three-month period, 150 local Chicago switching offices were connected through the new Chicago toll office to 57 toll offices in California, Illinois, Florida, and Ohio. This cutover was the culmination of seven years of designing, manufacturing, and testing the new No. 4 ESS toll switching system. A second No. 4 ESS office in Kansas City, Missouri, was cut into service on July 3, 1976. Two more offices, Jacksonville, Florida, and Dallas, Texas, went into service in December 1976, and nine more offices are scheduled for service in 1977.

The development and testing of a toll system of this complexity was a sizable undertaking. The framework for integrating such a system was laid down in the initial planning stages of the project, and was supported throughout by the existence of design standards, documentation re-

quirements, and control procedures. Some of the important aspects, in this regard, in the No. 4 ESS development were:

(i) Design standards for both hardware and software were developed and documented early in the project, and almost without exception, followed throughout.

(ii) A procedure for the overall design of the system, as well as for each hardware and software unit, was specified. The requirements for No. 4 ESS are based upon the requirements for a Bell System toll office. For each unit developed, two levels of design specifications were written, reviewed, and approved. The top level was known as a requirements design specification, and the second level as an implementation design specification. Each approved document became the vehicle for the next development step.

(iii) Procedures for documenting both hardware and software changes were developed very early in the project. These documents provided the vehicle for the institution of control and coordination procedures as required.

(iv) The development of test facilities was begun at the same time as the development of the system and was subjected to the same standards and development procedure requirements.

(v) Early in the development, it was decided that there would be no field trial and that the development would progress directly from a laboratory test to the first commercial installation. By January 1970, Chicago had been identified as the site of the first installation and a first office planning committee had been formed.

The following sections of this article describe how the overall laboratory and field testing was implemented. The excellent performance of the overall system at cutover resulted not only from good design but from the thorough planning, scheduling, and testing of the system before cutover.

## II. TESTING THE SYSTEM

The testing and design verification of No. 4 ESS was accomplished in three distinct phases. In the first phase, the many units making up the system were tested individually. The definition of a "unit" was rather broad. For example, hardware unit testing included verification that individual circuit packs met their design requirements and also that all of the frame types used in No. 4 ESS worked together in a laboratory environment. Software unit testing encompassed everything from simulating small software modules to verification that, for example, the data administration system correctly handled all specified input messages.

The second phase of the job was called functional testing. In this phase, the objective was to verify that not only was each system function cor-

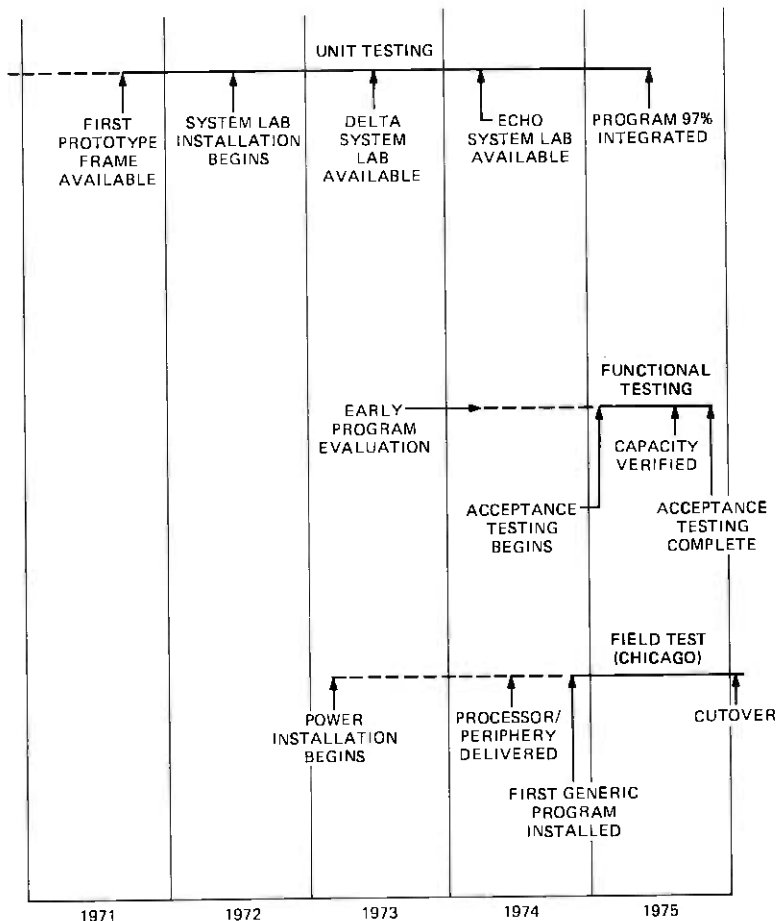


Fig. 1—Testing of No. 4 ESS.

rectly performed, but that the system remained stable during the performance of any combination of instructions.

The third phase was the evaluation and testing of the No. 4 ESS system under field conditions. In this phase, unit and functional tests similar to those run in the laboratory were repeated in order to provide a baseline from which to test the system in its operating environment.

The testing of No. 4 ESS was a continuing effort for a period of five years, as shown on Fig. 1. While the objectives of each testing phase were distinct, the phases overlapped in time. This overlap was accomplished by beginning functional testing as soon as unit tests on a cohesive set of features were completed, and beginning field testing as soon as usable stability was achieved. Also shown on Fig. 1 are several major benchmarks referred to in the next sections.

### III. UNIT TESTING

#### 3.1 Hardware—early testing

Initial verification of the hardware design was done using a logic simulator called LAMP,<sup>1,2</sup> running on a general-purpose computer. The LAMP simulator was capable of handling units varying in size from circuit packs of several hundred gates to frames of several thousand gates. In addition to verifying the hardware design, the test sequences used to detect and locate faults in the circuits were verified using LAMP.

Once designed and LAMP-tested, circuit pack models were constructed in Bell Laboratories' model shops and tested by the designers. Some skeletonized frames were constructed at Bell Labs to verify concepts and check timing. However, the general reliance was on Western Electric for production of prototype equipment. The first of a kind for each frame was built by Western Electric and tested by the designers at Bell Labs using a minicomputer-controlled test set. Since the Bell Labs and Western Electric test sets were compatible, once the test data base was designed and verified at Bell Labs it could be transferred to Western Electric and all future frames tested in the factory. Exactly the same test data forms the basis for the Installation Test System described in Section 5.2 and the generic diagnostic program.<sup>3</sup> As individual frame types were verified, groups of prototype frames were brought together in hardware laboratories where frame intercommunication and further operational checks were made.

#### 3.2 Software—early testing

Just as the early hardware testing was done with a simulator (LAMP), early software was first verified on an ESS program simulator running on a general-purpose computer. While all software could not be completely tested on the simulator, it proved to be a very efficient way of eliminating a large percentage of bugs early in the development. In all, 90 percent of the generic code was tested on the simulator. As was mentioned previously, most of the diagnostic tests (also approximately 90 percent) were verified in frame test and in the Installation Test System.

Before an operational 1A Processor was available, further testing of the software was accomplished using an available No. 1 ESS Processor. To do this, the No. 1 ESS Processor was slightly modified and connected via a No. 4 ESS peripheral unit bus branching frame to a signal processor, time-slot interchange, and time-multiplexed switch. The program, written in a combination of 1A Processor assembly language and ESS Programming Language (EPL), was translated so as to run on the No. 1 ESS Processor. Sufficient progress was made in checkout of the operating system and call-handling programs to allow a call to be completed

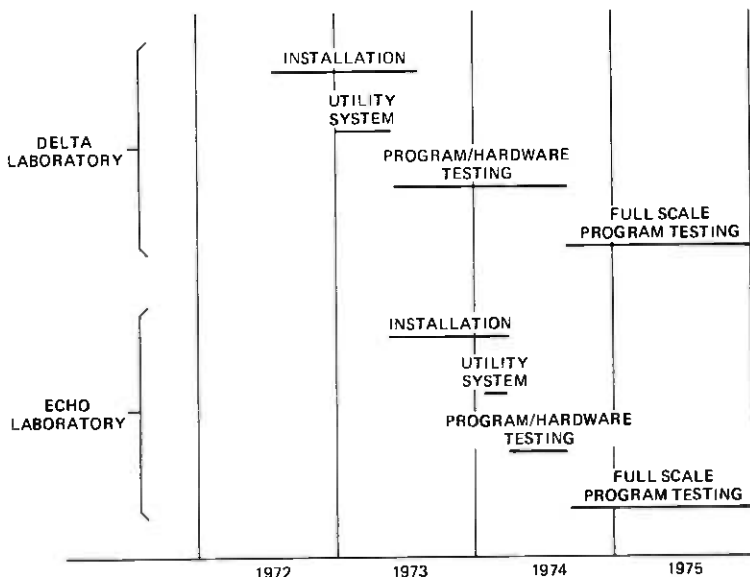


Fig. 2—No. 4 ESS system laboratory activity.

on November 2, 1972, seven months before a 1A Processor was available for general-program debugging.

### 3.3 System laboratories

Two system laboratories were devoted to testing No. 4 ESS. A system laboratory consists of a 1A Processor, a minimum set of No. 4 ESS peripheral hardware, and transmission switching interface and toll terminal equipment. An additional system laboratory containing only a 1A Processor was also used, mainly for testing 1A Processor common programs.<sup>4,5</sup> There were two reasons for having two complete system laboratories. First, the initial laboratory, which we called Delta, contained prototype equipment, while the second laboratory, which we called Echo, was made up of production equipment. Having Echo guaranteed that testing, especially the testing of changes, was carried out in a realistic environment. The second reason for two system laboratories was to double the available program testing time in order to meet the cutover schedule.

As shown on Fig. 2, activity in each system laboratory began with installation and testing of the equipment and utility system.<sup>6</sup> Program checkout began on June 1, 1973, in Delta and on April 1, 1974, in Echo. However, in both system laboratories, the initial program debugging began in an environment of continuing hardware debugging and hardware change activity. Many of the hardware changes, involving both

wiring and circuit pack change, came about as a result of problems found when operating with the generic program. One measure of change activity is shown in Fig. 3. The curve shows the cumulative number of wires changed in the two system laboratories during 1974 and 1975. Change activity consumed a great deal of laboratory time in early 1974, but began to level off by August when basic operating stability was achieved. It then increased slightly and continued at a relatively uniform rate throughout most of 1975. However, in this later period the changes generally were such that they did not cause extensive interference with program debugging. The approximately 40,000 backplane wires changed in the system laboratories in 1974 and 1975 were about 1.5 percent of the total number in the laboratories.

Beginning August 1, 1974, both system laboratories were devoted to full-scale program checkout. Full-scale program checkout was defined to be the point where 16 hours a day were scheduled for program debugging and where at least 80 percent of the scheduled time was actually productive. Unproductive time was caused by such things as hardware failures or change activities not being completed on schedule.

From the time program debugging began, the system laboratories were operated in two different modes. One was a closed shop, batch-type mode where programmers submitted test decks, which were run by operators, and printouts returned. Of the total system laboratory time available

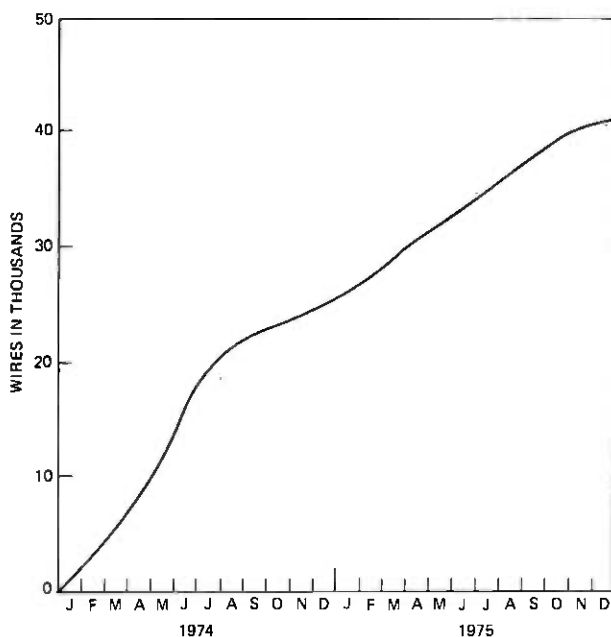


Fig. 3—Cumulative backplane wiring changes in Delta and Echo laboratories.

for program debugging during the unit testing interval, approximately 55 percent was devoted to batch operation. The availability of a powerful utility system and automated testing aids<sup>6</sup> made this mode of operation viable.

The remaining system laboratory program debugging time was devoted to hands-on operation. Hands-on time was assigned when required by specific problems, such as those whose solution required the interplay of a programmer with debugging tools and a hardware designer with oscilloscope, and also where problems existed which were impeding future debugging progress in other areas. In this second situation, a small group of programmers was assigned blocks of time. Much of the time was used running "instant turn-around" batch-type test jobs for these programmers.

Using both batch and hands-on techniques, a program debugging rate of about 20,000 instructions a month was sustained from September 1973 through February 1975. The fast startup of program debugging, shown in Fig. 4, was a result of the extensive pretesting of both the hardware and software.

#### IV. FUNCTIONAL TESTING

##### 4.1 Laboratory system testing

An independent system test group was chartered to provide a systemic testing viewpoint. A set of system-level test sequences was generated

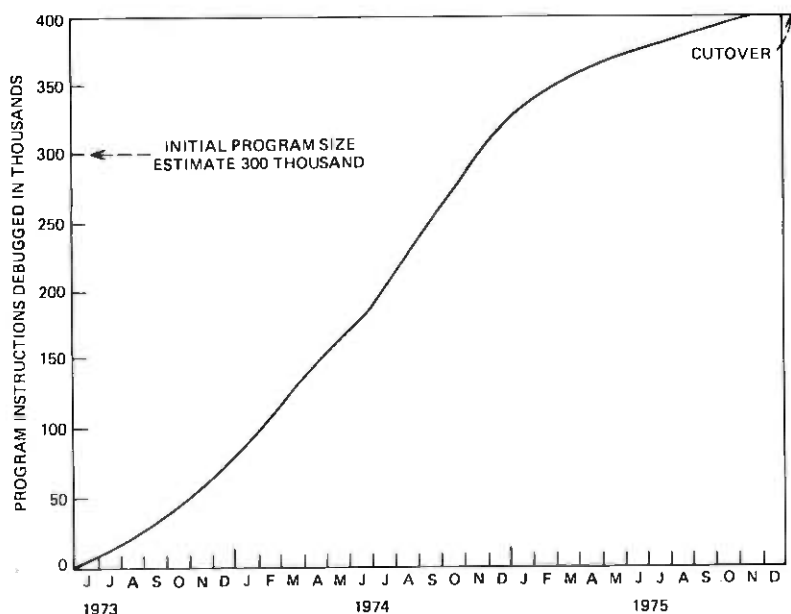


Fig. 4—No. 4 ESS program integration.

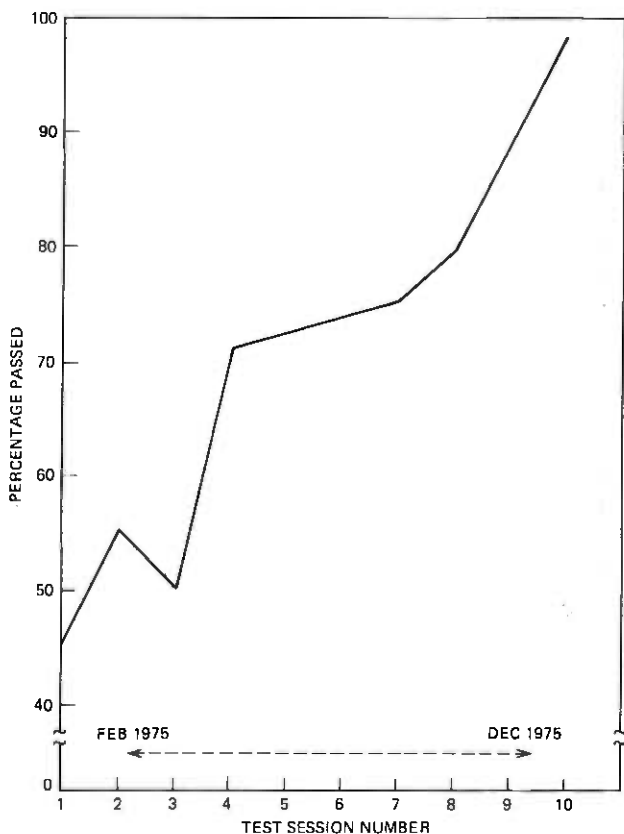


Fig. 5—Laboratory acceptance testing.

and executed in a controlled, highly interactive environment. A subset of these test sequences was used as a generic system laboratory acceptance test. This acceptance test was executed periodically during the functional testing interval, and the statistical performance results, as a function of time, were used to indicate the rate of achieving system stability. The results of the acceptance tests are presented in Figs. 5 and 6. Figure 5 shows the percentage of tests in an acceptance test that executed as expected. Figure 6 shows the mean time between unscheduled major recovery events. Both are graphed as functions of time. Absolute numbers were not used as prime indicators, rather the slopes were used and, as can be seen, both figures indicate that system stability was being achieved.

The highest-priority goal of the functional testing interval was to achieve system stability in the presence of system faults. This was focused on with the firm conviction that this characteristic was funda-



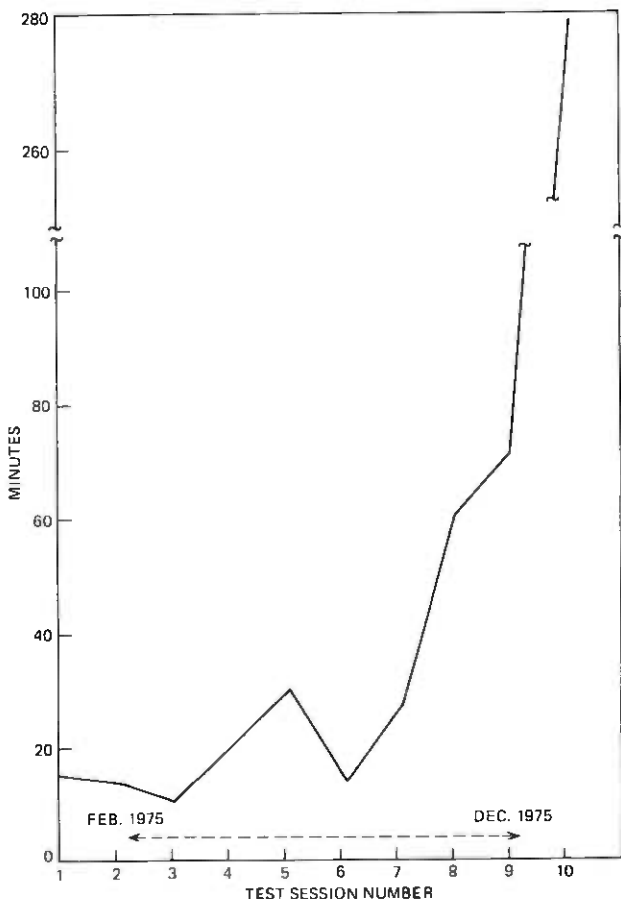


Fig. 6—Mean time between major recovery events.

mental to our ability to resolve problems in an operational office. This important goal was achieved.

The key point was that this was an independent effort. The people were responsible for the system, not for any individual component of it. They ensured that the system was ready for release, using the system laboratory as the testing vehicle. This was achieved by the system test group being the focal point of change control and also the primary interface with the field test group.

#### 4.2 Laboratory System Evaluation

A great deal of effort was spent to ensure that No. 4 ESS would meet all of its real-time objectives. Early in the project's life, numerous real-time simulations were developed and executed to establish confidence

that these objectives would be met. These simulators were designed such that after the system laboratories became operational, the same experiments could be run there to verify the simulator results. These experiments were done in three areas: (i) system response versus call attempts per hour; (ii) system overload control responses as a function of call load; (iii) system engineering requirements versus real-time capacity. In all three cases, the simulator results were verified, and in the case of real-time capacity, allowed us to increase the advertised capacity from 350,000 to 550,000 engineered call attempts an hour.

In summary, we were able to verify our simulated system performance characteristics, and thus, to achieve all engineering requirement system objectives.

## **V. ENGINEERING AND PLANNING EARLY OFFICES**

### **5.1 Selection of Early Offices**

There were many locations at which there was need for a large toll switching machine. Chicago was chosen jointly by AT&T, Bell Labs, and Illinois Bell as the site for the first No. 4 ESS for a number of reasons. It is conveniently close to both Bell Labs at Indian Hill, the development location, and to Western Electric at Lisle, the principal manufacturing location. This proximity facilitated support during the installation, testing, and initial service periods. In addition, the office, called Chicago 7, was planned to be a new-start office rather than a replacement. This avoided many problems that would have been associated with the cutover of in-service facilities from one office to another. It permitted the development of a simple phased cutover and contingency plan.

Kansas City was chosen as the site of the second office, called Kansas City 2, in part, in order to face the problems associated with replacing an in-service 4A crossbar office with a No. 4 ESS. These problems include the reuse of existing toll terminal equipment and the development of flash cutover procedures.

The selection of subsequent offices was done by Long Lines and AT&T and was based on need for additional switching capacity and the buildup in Western Electric production rate.

For both Chicago 7 and Kansas City 2, office planning committees were organized with representatives from AT&T Headquarters, Long Lines Headquarters, local Long Lines Operations and Engineering, Western Electric Product Engineering Control Center and Installation, and Bell Labs. These committees were distinct from and in addition to the normal operating company cutover committees. The planning committees established schedules for office engineering, manufacturing, installation, and testing. Progress was reviewed regularly and problems were solved as they arose. Because many decisions were required before

standard methods had been developed and documented, Bell Labs engineers did much of the engineering of the processor and network for Chicago 7. Western Electric's Product Engineering Control Center participated in this effort and developed the standard engineering questionnaire as a result. The transmission facilities for Chicago 7 were all new and posed no special problems. They were engineered by Long Lines and installed by Western Electric. Because Kansas City 2 was to be a replacement for an existing 4A crossbar office, the reuse of an existing set of transmission facilities raised many questions about methods of connection to No. 4 ESS, trunk circuit compatibility, and segregation of private line and switched circuits that were mixed in existing carrier systems. The economics of reusing existing older equipment compared with replacement by new toll terminal equipment was also considered. Bell Labs System Engineering worked closely with operating company engineers to answer these questions.

As part of the system development, standard floor plans were developed for the major elements of the system, i.e., 1A Processor, time-division network, signal processor-voiceband interface cores, and toll terminal equipment. Therefore, the preparation of floor plans for Chicago 7 and Kansas City 2 was simplified to arranging these large blocks, rather than individual frames, in the available space.

### **5.2 Installation Test System**

Concurrent with the development of the No. 4 ESS, Western Electric engineers developed factory frame tests and installation tests based on the common tests described earlier. A key element in the design of No. 4 ESS is that a single set of tests be developed and used for all testing of a particular type of unit, i.e., single unit testing in the factory, system testing during installation, and diagnostic testing in an in-service office. Installation Test System (ITS) is the name of the program used by the Western Electric installation testers to control the execution of unit and subsystem tests. ITS is characterized by permitting tests to be run simultaneously on several different pieces of equipment. A prerequisite for the beginning of system testing at Chicago 7 was that all ITS unit and subsystem tests pass successfully. The testing activity at Chicago 7 spanned a period of 1½ years starting in mid-1974 when ITS tests were begun. Figure 7 shows the sequence of tests through the phased cutover in early 1976.

### **5.3 Field Test Plan**

A field test plan consisting of functional tests, environmental tests, and an overall acceptance test was written and used as a final check that the system met its functional objectives at specified environmental

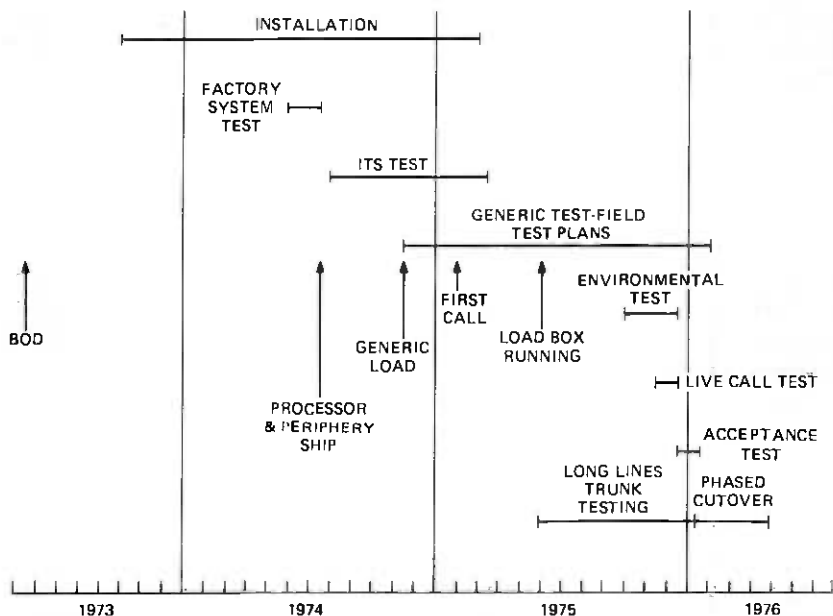


Fig. 7—Chicago 7 testing activity.

limits. About eleven thousand individual functional tests were written and most were executed several times during the system test interval at Chicago 7. The environmental tests verified system performance at the limits of high temperature and low voltage. A program-generated call load of 80,000 calls per hour was applied in order to verify the adequacy of the office engineering. In addition, six persons at real telephones placed several thousand calls in order to get a subjective measure of the system performance. Final acceptance of the system as ready for commercial service was based on tests during which the system was operated for several days by the Long Lines craft while it was switching test traffic.

This functional testing interval began on one shift a day in November 1974, increased to two shifts a day in January 1975, and went to three shifts a day in March of 1975. During this period, many tasks competed for system time. In addition to the execution of field test plans, time was required for the installation of hardware design changes, program corrections, ordinary maintenance troubleshooting, and testing and alignment of the connecting trunks by Long Lines. Although several tasks might be executed at the same time, careful scheduling was required to avoid the many incompatible combinations of system use. From the earliest stages of planning the system integration and testing, it was agreed that there would be no scheduled program debugging at Chicago

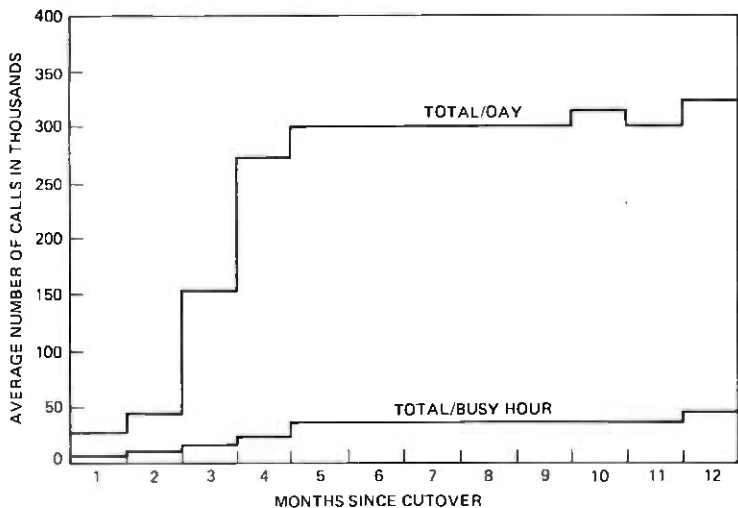


Fig. 8—Chicago 7 traffic.

7 in order to avoid yet another demand for system time. Despite thorough verification in the system laboratories, there were occasions when program tests were run at Chicago 7 to identify problems that were unique to the office because of its size, hardware configuration or data base. Despite these occasional lapses, the normal procedure was for system performance problems to flow from the field test group to the system test group and for integrated programs and tested changes to flow to Chicago 7.

## VI. CHICAGO 7 PERFORMANCE EVALUATION

The cutover of Chicago 7 was accomplished on the schedule set in 1972. The cutover was of the phased type, covering the period January 17 to April 24, 1976, during which about 4500 toll connecting trunks from 150 local offices in the 312 NPA and a nearly equal number of intertoll trunks to 57 toll offices in foreign NPAs were put in service. Traffic data for Chicago 7 are shown in Fig. 8. Traffic built up as trunks were added and reached a busy-day high of 350,000 and a busy-hour of 31,000 calls at the completion of the phased cutover. Additional trunks to the NPAs originally served were added in the summer of 1976 and trunks to New York City were added starting in the fourth quarter of 1976. The trunking configuration for Chicago 7 at the start of 1977 is shown in Fig. 9.

There are several criteria for judging the performance of an electronic switching system. These include the number of ineffective attempts, as a measure of the grade of service received by the customer, and measures of the required maintenance effort on the part of the operating company.

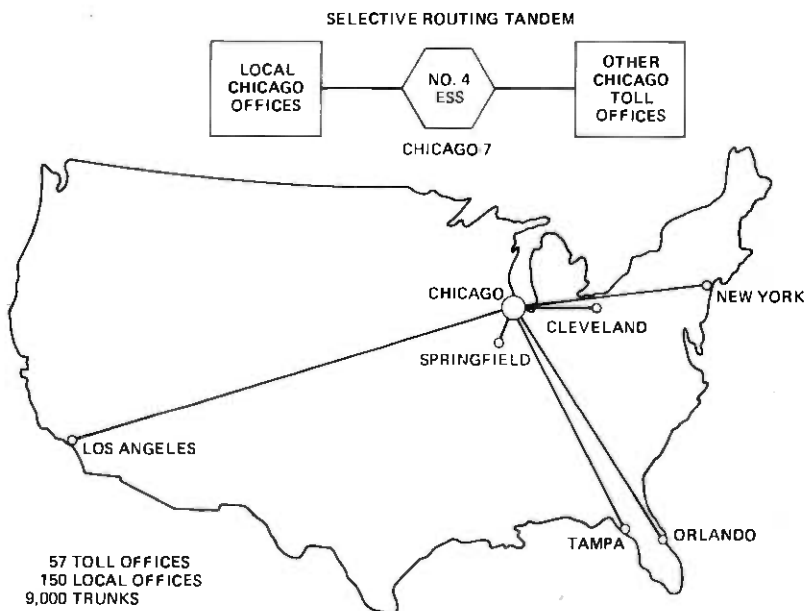


Fig. 9—Chicago 7 trunking.

Prior to cutover, two criteria were set for each of a number of measurements, Table I. These were a desirable objective and second a level at which the advisability of putting the office into service would be questioned.

Ineffective attempts result from sources external to the switching machine as well as from internal causes. Figure 10 shows the history of ineffective attempts at Chicago 7 for the first year of service. The most recent data show total ineffective attempts averaging less than 1 percent.

The replacement rate for plug-in apparatus is one measure of the maintenance effort required. Figure 11 shows the monthly average of the daily replacement rate through the first year of service. During the first 122 days of service, there were 413 plug-ins of all kinds replaced,

Table I — Cutover performance criteria

|                      | Objective | Concern threshold |
|----------------------|-----------|-------------------|
| Ineffective attempts | 1.25%     | 3.0%              |
| Plug-in replacements | <2/day    | 7/day             |
| Interrupts           | <50/day   | 200/day           |
| Phases (2 or higher) | 1/2 month | 1/2 day           |

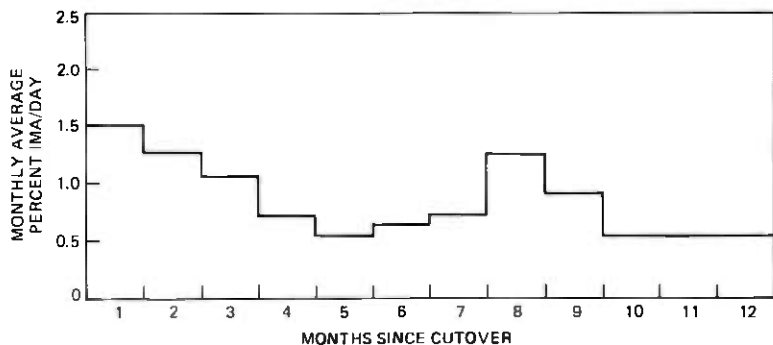


Fig. 10—Chicago 7 ineffective machine attempts.

for an average replacement rate of 3.4 a day. The present average replacement rate is between 2.5 and 3.0 a day.

A maintenance interrupt occurs whenever the system experiences an event of sufficient importance to interrupt the normal flow of program control in order to take immediate action. Maintenance interrupt actions require only a few milliseconds and have no impact on customer service. They are, however, a measure of required maintenance activity. Another measure of system performance is the number of occurrences for which the system response was a memory reinitialization phase<sup>3</sup> of sufficient severity to have an impact on call processing. Neither a phase 2 nor a phase 3, either of which can be run automatically, will disturb calls in the talking state, but these phases will prevent the completion of calls in the process of being established. A phase 4, the most severe reinitialization, can be induced only by manual action. A phase 4, in addition to preventing completion of transient calls, will disconnect stable calls in the talking state. One observation is that there is a tendency for one

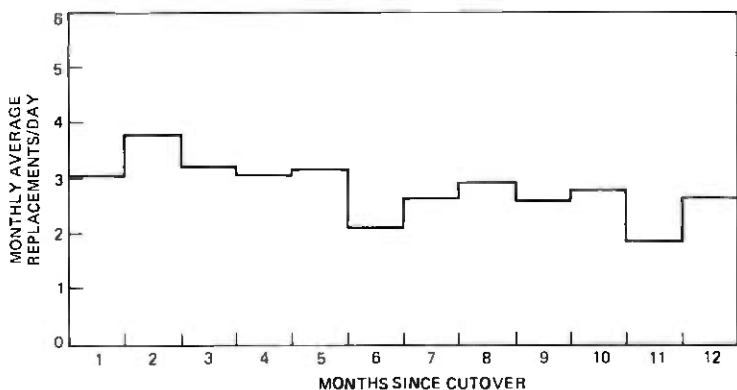


Fig. 11—Chicago 7 plug-in replacements.

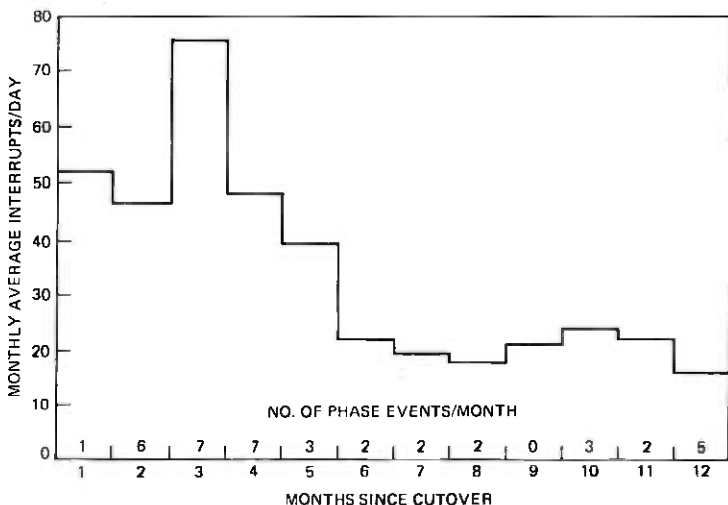


Fig. 12—Chicago 7 interrupts and phase events.

stimulus, e.g., a hardware failure, to trigger a number of phases in rapid succession. Each such incident has been studied in detail and these studies have resulted in a number of improvements in the maintenance program strategy. Transient hardware errors resulting from marginal equipment failures are one of the most difficult classes of problem for the system to cope with. Figure 12 shows both the monthly average of the number of interrupts a day and the occurrences of maintenance incidents that required one or more reinitialization phases.

## VII. SUMMARY

The successful introduction of No. 4 ESS into commercial service in Chicago 7 resulted not only from good design but also from a large effort directed toward system testing and integration. Initiated early in the development process, these efforts included careful planning for measuring system performance and coordination of the introduction of hardware and program changes. This ensured that the process converged toward a stable and satisfactory system design.

The overall performance of the Chicago 7 system has been excellent. No fundamental system problems have been encountered, although some software and hardware improvements have been incorporated as a result of operational experience. It is expected that all performance objectives will be reached in the near future as a result of additional craft training and recently introduced system improvements.



## REFERENCES

1. H. Y. Chang, G. W. Smith, Jr., and R. B. Walford, "LAMP: System Description," *B.S.T.J.*, 53, No. 8 (October 1974), pp. 1431-1449.
2. T. T. Butler, T. G. Hallin, J. J. Kulzer, and K. W. Johnson, "LAMP: Application to Switching System Development," *B.S.T.J.* 53, No. 8 (October 1974), pp. 1535-1555.
3. M. N. Meyers, W. A. Routt, and K. W. Yoder, "Maintenance Software," *B.S.T.J.*, this issue, pp. 1139-1168.
4. G. F. Clement, P. F. Fuss, R. J. Griffith, R. C. Lee, and R. D. Royer, "Control, Administrative, and Utility Software," *B.S.T.J.*, special issue on 1A Processor, 56, No. 2 (February 1977), pp. 237-254.
5. P. W. Bowman, M. R. Dubman, F. M. Goetz, R. F. Kranzmann, E. H. Stredde, and R. J. Watters, "Maintenance Software," *B.S.T.J.*, special issue on 1A Processor, 56, No. 2 (February 1977), pp. 255-289.
6. P. S. McCabe, J. B. Otto, S. Roy, G. A. Sellers, and K. W. Zweifel, "Program Administration, Test, and Evaluation," *B.S.T.J.*, this issue, pp. 1239-1278.



## **No. 4 ESS:**

# **The Switched Digital Network Plan**

By J. E. ABATE, L. H. BRANDENBURG, J. C. LAWSON,  
and W. L. ROSS

(Manuscript received August 2, 1976)

*A plan for the introduction of time-division switching in the intertoll network is outlined. The plan uses the No. 4 ESS as the switching vehicle. The plan defines a Switched Digital Network (SDN) which can evolve compatibly with the existing Switched Analog Network (SAN). The plan introduces 64 kilobits per second time-division multiplexed PCM as a standard switching signal format in the intertoll network. Facilities utilization, trunk design, timekeeping, and maintenance plans which are required for the new format are presented. Effects on intertoll network evolution, particularly voice performance, are assessed.*

## **I. INTRODUCTION**

The selection of a digital format for the No. 4 ESS switch represents a significant step in the continuing evolution of the intertoll network in the United States. This selection was dictated by the economics of advancing technology and by the requirements for a large toll switch which could economically satisfy current and projected volumes of intertoll service.

Economic considerations, however, must be bounded by the requirements of a viable technical plan for the installation and operation of a digital toll switch in an evolving intertoll network which is now analog and which will remain substantially analog for many years in the future. This paper outlines the basic characteristics of the network plan for introducing digital switching into the intertoll plant.

The plan proposes direct interconnection of digital transmission facilities, digital switches, and analog-to-digital converters to form a Switched Digital Network (SDN).

Facilities in the SDN will be the No. 4 ESS switch; a set of digital transmission facilities which carry digital bit streams on paired wire, coaxial, radio, and millimeter waveguide at rates from 1.544 megabits per second to 274 megabits per second; a set of analog-to-digital converters which include D-type channel banks using the  $\mu = 255$  D2 coding law and the Voiceband Interface Frame (VIF); and a digital interface, the Digroup Terminal (DT), which permits direct digital interconnection between the No. 4 ESS switch and digital transmission facilities. This facilities set will permit a digital connection to be established from a local (class 5) switch through the intertoll network to a distant local switch. Additionally, the plan provides for compatible interconnection with the existing switched analog network (SAN) so that the intertoll DDD network can include both the SDN and the SAN as its evolution continues.

This plan introduces two new types of trunks into the telephone plant. In addition to the existing 4-kHz analog trunk, digital trunks and combination trunks will be required. Digital trunks transmit and receive a 64 kilobit per second PCM format. They interface with No. 4 ESS at both ends through the digroup terminal (DT). Combination trunks interface with 4-kHz analog at one end and 64 kilobits per second at the other. The analog end terminates in a D-type channel bank and the digital end in the DT.

The introduction of these two new types of trunks changes long-established viewpoints of network engineering, transmission objectives, and maintenance. Digital trunks, in particular, require new objectives and maintenance procedures. They also require the operation of a nationwide timekeeping plan.

Digital switching offers a new opportunity for integration of switching and transmission equipments. From the origin of telephony to the present, switching and transmission systems have been integrated into a telecommunication network at a 4-kHz baseband interface. Progress in both switching and transmission has been constrained by the traditional requirements of the interface which affect both transmission and signaling. The introduction of a 64 kilobit per second PCM signal in a word-organized time-division multiplex format offers significant opportunities for improved performance, and reduced capital and operational costs.

Section II provides a basic overview of the network plan for the SDN, an overview of the major differences between the SDN and the SAN, and the expected mode of evolution of the SDN.

Section III provides some details of the SDN plan in the areas of signal formats, facilities utilization, trunk design (loss and level designs), timekeeping, and maintenance. Space does not permit a complete description. Details such as idle channel code selection and facilities provisioning are included. These details were critical elements requiring

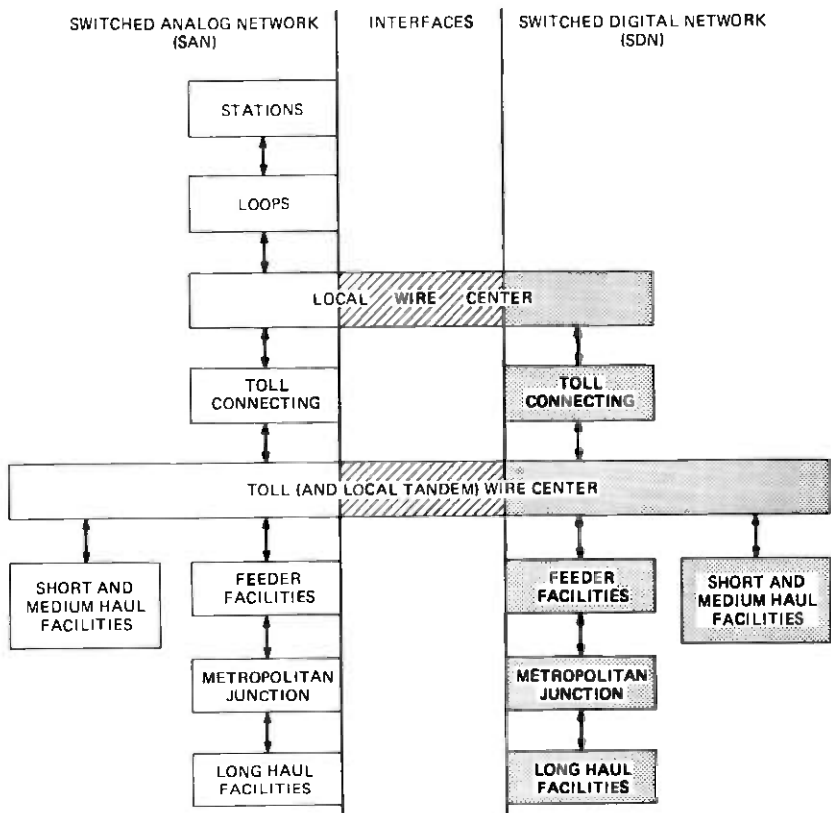


Fig. 1—Functional organization of the SAN, SDN, and interfaces.

specification so that a viable technical plan could be realized.

Section IV assesses expectations of voice performance of the network as the SDN evolves. Section V concludes with a summary of trends which begins as the SDN is introduced and notes some currently perceived issues which are to be resolved as future technology develops.

## II. SDN PLAN IMPACT ON DDD NETWORK EVOLUTION

The introduction of No. 4 ESS and with it, the SDN plan, will have impact on the DDD network from the viewpoints of new engineering and operational methods and economics. This section gives an overview of these viewpoints as perceived at the present time.

### 2.1 Compatibility of the SDN with the SAN

Compatibility of the SDN with the SAN is the dominant constraint for the introduction of No. 4 ESS and the SDN into the current DDD network. Figure 1 broadly illustrates how the SDN will interface. The SDN is

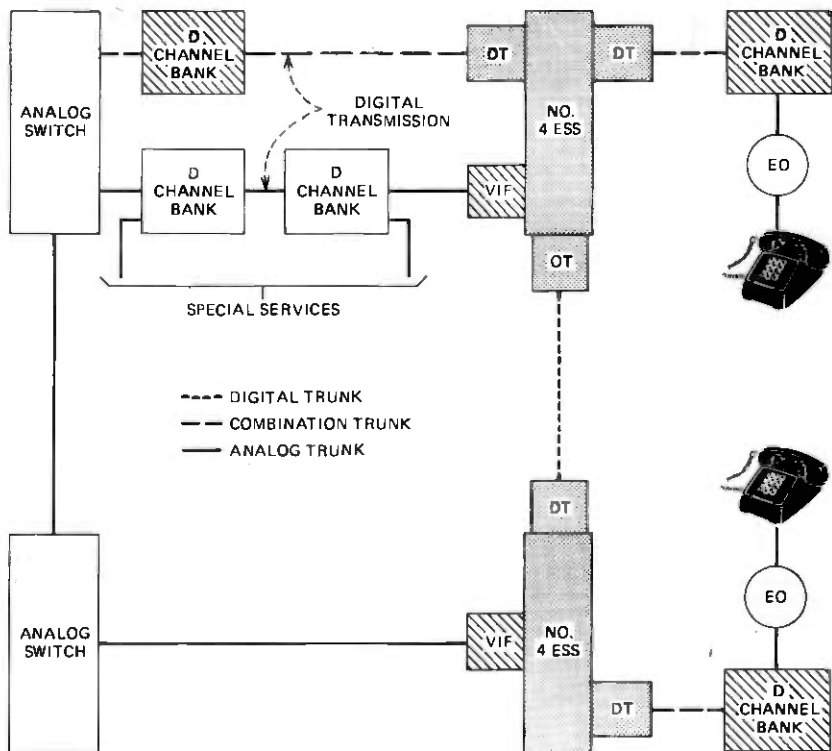


Fig. 2—Simplified configuration of analog, digital, and interface facilities as parts of the SAN and SDN.

constructed exclusively with digital transmission and switching facilities. In contrast, the SAN is constructed with analog switches, and a mixture of digital and analog transmission systems. These two facilities networks are distinct. Interconnection will occur at toll and local wire centers where analog-to-digital conversion occurs.

Traditionally, transmission and switching designs in the DDD network have adopted a standard 4-kHz interface which permits route selection by the switch to be made independently of transmission facility type, and to be dependent only on route destination and traffic engineering and design considerations. The SDN introduces a second interface standard which is a  $\mu = 255$  PCM word-organized digital format<sup>1</sup> in a way which preserves the traditional method of route selection by the switching machine. Figure 2 illustrates, in more detail, the component parts of the SDN, and the way 4-kHz and 64 kilobit per second interfaces coexist in the combined SAN SDN network that will be the DDD network of the future. As shown in Fig. 2, traditional switching functions are maintained by the introduction of two new trunk types, digital and

combination, which were defined in the previous section. As a result of these choices, it is expected that the basic functional and service characteristics of the intertoll network will be unchanged by the introduction of the SDN.

## **2.2 Impact of SDN plan on the DDD network**

The SDN plan impacts the engineering and operation of the DDD network in four areas. These are trunk design, facilities utilization, maintenance, and synchronization.

The selection of a signal format, and loss and level plan, represents a significant change from traditional standards in the network. These choices, which are detailed in Sections 3.1 and 3.3, simplify maintenance, level administration, and trunk design and minimize the need for digital processing. The significant features of this plan are:

(i) The  $\mu = 255$  PCM format described in the D2 channel bank specification<sup>1</sup> becomes a network standard.

(ii) Loss and level administration merge into a unified plan rather than being treated independently as in the past.

(iii) A digital milliwatt test signal becomes a standard at all test points in the SDN; furthermore, this test signal is compatible with the analog milliwatt test signal in the SAN.

(iv) The preservation of bit integrity through the network offers opportunities for simplified and enhanced maintenance, and the promise of enhanced voice performance.

Direct digital interconnection of digital and combination trunks on No. 4 ESS changes facilities utilization in three significant ways:

(i) Significant economies have been demonstrated by directly terminating the digroup, which is a 24-channel TDM-PCM signal carried on 1.544 megabit per second facilities. These economies are sufficient to eliminate the use of voice-frequency cable in the toll connecting trunk plant and to stimulate the use of T1 carrier significantly.

(ii) The 24-channel digroup becomes the basic unit in facilities provisioning instead of a single circuit. Past practices of intermixing message channels and private-line channels on transmission facilities requires modification. Section 3.2 describes a plan called modified segregation in which the majority of message trunks and private line channels are carried on segregated facilities.

(iii) Channel numbering and trunk numbering in switching machines are strongly interrelated because of the sequential nature of the 24-channel word (time slot) organized TDM-PCM digroup. The SDN plan uses the straightforward one-to-one correspondence between channel,

time slot, and trunk (traffic) number with the expectation of simpler and more straightforward data base and record administration.

Maintenance of the SDN will be simplified because of the opportunities for automatic error monitoring in digital switches and digital facilities. Significant economies will occur in the maintenance of digital trunks, since a successful test of one digital trunk in a digroup will assure the performance of the remaining 23 trunks.

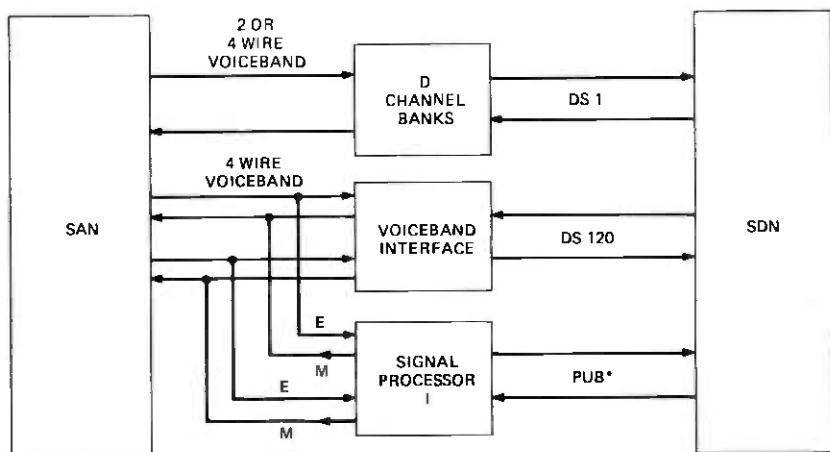
Finally, the introduction of the SDN into the DDD network requires the introduction of a synchronization or timekeeping plan. A timekeeping plan, described in Section 3.4, is designed to maintain synchrony between parts of the SDN to about one part in  $10^9$  when the network is stressed because of equipment failures. Normally, timekeeping will be maintained to accuracy attainable with atomic clocks. The timekeeping plan will provide a level of performance which makes timekeeping impairments substantially smaller than impairments caused by hits, outages, and equipment failures.

### ***2.3 The Evolution of the Switched Digital Network***

The SDN will be formed as No. 4 ESS machines are installed and are interconnected via short, medium, and long-haul digital transmission systems as dictated by economics. Thus, the manner in which the SDN grows and evolves is a function of the economical sequence in which No. 4 ESS's are installed throughout the Bell System for both growth and modernization, and the existence of digital transmission facilities where they prove-in over their analog counterparts. This section describes how these economic considerations will dictate a particular mode of evolution of the SDN, called "islands," and discusses some of the operational characteristics that result from such a mode.

Beginning with the introduction of four No. 4 ESS machines in service within the Bell System at the end of 1976, and a projection of several score more by the end of the decade, the SDN will emerge. It should be emphasized again that the timing and location of this installation sequence is determined by the economics of each installation on a case-by-case basis and not solely on the pre-existence of digital carrier systems. During this same period, as in the past, the Bell System will be installing short-haul digital carrier and also medium-haul digital systems when the case-by-case economics so dictate. Economic studies have shown that the lower cost of terminating digital carrier on the No. 4 ESS does substantially reduce the prove-in distance of T1-carrier, resulting in considerably greater use of T1, but has a limited ability to prove-in additional long-haul digital systems over those which would be economical with conventional analog terminations. Thus, it is expected that the SDN will materialize in metropolitan clusters of "islands" intercon-





\*PUB = PERIPHERAL UNIT BUS

Fig. 3—Interface terminals between SDN and other communication networks.

nected by digital transmission systems that are short enough that the termination savings possible with the digroup terminal will have a measurable effect on total system economics. It is only later, when long-haul digital systems prove-in on their own merit, that a nationwide SDN will result.

In summary, then, the early years of the SDN will be marked by three characteristics: clusters of No. 4 ESS machines in "islands," a large number of combination toll connecting trunks and a smaller number of short digital trunks within islands, and a limited digital interconnection of islands with long-haul digital facilities.

These characteristics of SDN evolution have dictated the form of the timekeeping plan (Section 3.4). This plan, as will be seen below, capitalizes on this mode of evolution. It will provide an orderly transition from a group of SDN islands to a nationwide, fully interconnected SDN.

### III. NETWORK PLAN

#### 3.1 Signals Within the SDN

In this section, the idle channel code signal and the two interface signals between digital transmission and switching equipment within the SDN are briefly described. The two interface signals are referred to as the DS-1 and the DS-120 signals. Typical appearances of these signals are illustrated in Fig. 3. The idle channel code is the digital signal transmitted by the No. 4 ESS on digital and combination trunks when such trunks are in the idle (on-hook) state.

### **3.1.1 The DS-1 signal**

The DS-1 signal carries one "digroup" (i.e., twenty-four 64 kilobit per second channels). The digroup is the lowest or first multiplex level in the SDN. It is a 1.544 megabits per second signal, organized into frames of 193 bits repeated at a rate of 8000 frames per second. Twelve such frames constitute a superframe of 2316 bits. The frame consists of twenty-four 8-bit words, called time slots, plus a 193d bit, which alternates in function between framing and signaling subframe.

Frame integrity is preserved in the SDN by aligning incoming DS-1 frames at the No. 4 ESS. Such alignment requires one-frame storage in the DT. However, superframe integrity is not retained in the SDN, since superframe alignment at each No. 4 ESS would cause a signal delay approaching 2 milliseconds at each switch. Such delays would cause greatly increased echo suppression costs. Thus, when channels on two distinct digroups are connected through the No. 4 ESS, the signaling bit transmitted from the No. 4 ESS will most likely occupy a bit position formerly occupied by PCM information on the incoming digroup. The impact of signaling frame realignment (called digit robbing) on SDN transmission performance is not expected to degrade service. Further, the planned transition to Common Channel Interoffice Signaling (CCIS) will eliminate the need for digit robbing.

The lowest level cross-connect in the SDN is at the DS-1 digital speed, or in groups of 24 voice channels. This characteristic of the SDN represents one of the most fundamental changes from current practice. In effect, it preserves digroup integrity throughout the network and legislates against per-circuit access for cross-connecting on a digital basis. This characteristic of preserving digroup integrity permits maintenance opportunities and efficiencies heretofore not achievable in the current analog network, as discussed in Section 3.5.

### **3.1.2 The DS-120 signal**

The No. 4 ESS may receive inputs from either of two interface terminals: the DT or VIF. The signal passing between these two interfaces and the Time-Slot Interchange (TSI) is referred to as the DS-120 signal. This signal is identical whether it originates from the DT or VIF. Thus, the switch need not keep track of which of these facilities the DS-120 signal originates from. The DS-120 bitstream can accommodate the 120 VF trunks processed by a VIF or five digroups processed by a DT. It is organized into a frame of 2048 bits with a frame rate of 8 kiloframes per second. The DS-120 signal is described in detail in another paper in this issue.

### **3.1.3 Idle channel code signal**

To ensure that all statistical and syntactical constraints on digital bitstreams within the SDN are met, No. 4 ESS will transmit an "idle

channel code" in all time slots corresponding to idle switch terminations. This code will prevent transmission impairments either on the digital facility or at A/D converters at the SDN boundary. The idle channel code will be generated within the Time-Slot Interchange (TSI) unit of No. 4 ESS. The format of the idle channel code is repetitive transmission of the code word 01111111.

The following describes the rationale for selecting the idle channel code and its point of generation in the SDN.

The reason for generating the idle channel code at the TSI is that it must be generated at a location where the busy/idle state of every trunk is easily accessible. This information is an integral part of No. 4 ESS operation and is readily available at the TSI; it will not be readily available to transmission equipment in a CCIS environment.

The statistical constraints on the minimum number of "ones" that a PCM word contains derives from the need for a T1 bitstream to provide sufficient timing energy in the signal to accurately time the regenerators of a T1 repeatered line. The requirement is that each 8-bit PCM word must contain at least one "one."

Repetitive transmission of a single 8-bit word is desirable because the TSI output buffer memory, where idle channel code generation seems most appropriate, is capable of storing at most one 8-bit word per channel per frame. It is relatively simple to read one predetermined word into the output time-slot buffer memory when a time slot becomes idle. There does not appear to be sufficient need for a more complicated code.

The idle channel code must satisfy syntactical constraints in order not to imitate any of the patterns used for terminal framing, signal framing, and alarm functions.

The requirements on the idle channel code are as follows:

(i) The idle channel code is to be a sequence formed by repetitive transmission of a single code word read toward the line at the No. 4 ESS TSI during periods when the switch termination is idle.

(ii) The word must contain at least one "one".

(iii) Digit 2 of the word must be a "one" because a repetitive zero would be interpreted as an alarm.

(iv) The word must not decode to a significant analog value at the SDN boundary.

The four constraints above are satisfied by repetitive transmission of code words of the  $\mu = 255$  inverted binary coding algorithm, near to the center of the coder characteristic. The word 01111111 is chosen for two reasons. Its slightly lower ones density results in less crosstalk impairment of crosstalk-limited digital transmission facilities, notably T1, and because, if a T1 line fails in such a way so as to produce all "one's," the

failure condition is more easily distinguished from the normal idle channel code sequence.

### **3.2 Facility utilization plan**

In the current analog network, digroups terminating on channel banks are often filled with a mixture of different types of message trunks and special service circuits. All circuits are returned to voiceband at each switch. Thus, by cross-connecting at voiceband, message trunks terminate at the switch and special service circuits can be routed as required. However, use of the digroup terminal to terminate a digroup on the No. 4 ESS digital switch excludes the possibility of trunk appearances on a distribution frame, resulting in the loss of individual circuit access. Therefore, special plans are needed to accommodate the presence of special service circuits.

The plan to be followed in the SDN can be described as a modified segregation plan. Strict segregation requires that digroups entering a digroup terminal be composed solely of message trunks terminating on the No. 4 ESS.

The imposition of a strict segregation rule would decrease the efficiency of utilization of transmission facilities. In some cases, additional digroups beyond those needed with current practice will be required. A decrease in digroup fill of about 5 percent has been estimated. Since the digroup fill will now be applied to No. 4 ESS switching equipment, such as signal processors and time-slot interchange units, the same reduction in fill occurs on switching equipment.

In addition, a strict segregation policy would require more effort for circuit layout. This, however, may not be a serious problem. In return for this extra effort, segregation would help to reduce the number of distribution frame appearances and consequently, their associated problems of space, installation, and data-base records. By fostering facility provision for No. 4 ESS on a digroup basis rather than one circuit at a time, segregation should also help to simplify future rehomings and rearrangements. However, the plant is not now segregated, and thus must be groomed for future segregation for No. 4 ESS.

Because of the likely reductions in fill as described above, total segregation may not provide as much savings as might appear at first, and a modification of the segregation policy which maintains facility fill is the recommended plan. The plan for facility utilization with No. 4 ESS is as follows: A policy of segregation will be followed where economic, but where segregation would cause unacceptable transmission cost penalties, a digroup may contain a mixture of switched and other circuits. About 97 percent of the digroups can remain in a segregated format, without a reduction in fill. Thus, the digroup terminal can still be used in most cases, but circuits on "mixed" digroups will be demultiplexed

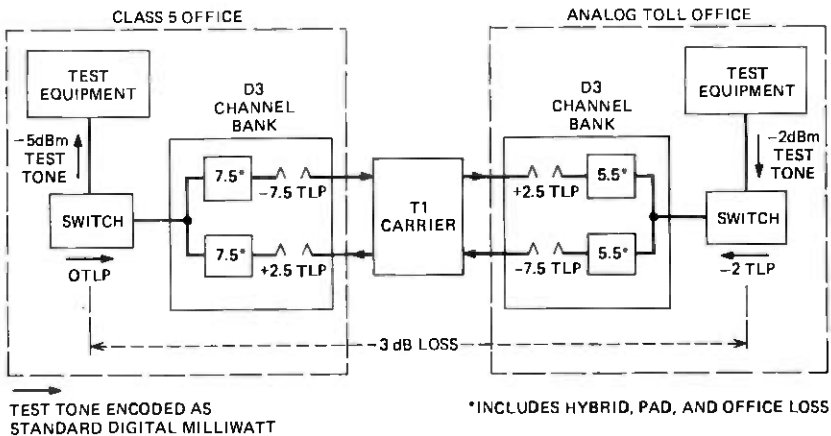


Fig. 4—Method of providing 3-dB toll connecting trunk loss in the SAN.

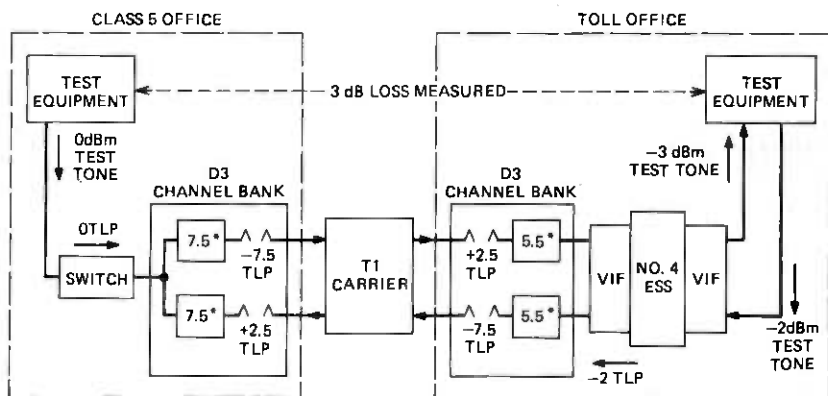
and decoded by a D channel bank, and must enter the No. 4 ESS through the voiceband interface. Circuits in these “mixed” digroups could also be permanently switched (“nailed-up”) rather than demultiplexed, if this capability is available for No. 4 ESS.

### 3.3 Loss and Level Plan

The effect of echo is currently controlled by the judicious use of trunk loss and echo suppressors. If these same techniques were used within the SDN, digital loss would be required. Since provision of digital loss would introduce additional cost and transmission impairments as well as require more administration, a study<sup>2</sup> was conducted to investigate whether loss is needed in digital trunks. This study showed that zero-loss digital intertoll trunks would suffice if the following conditions were met:

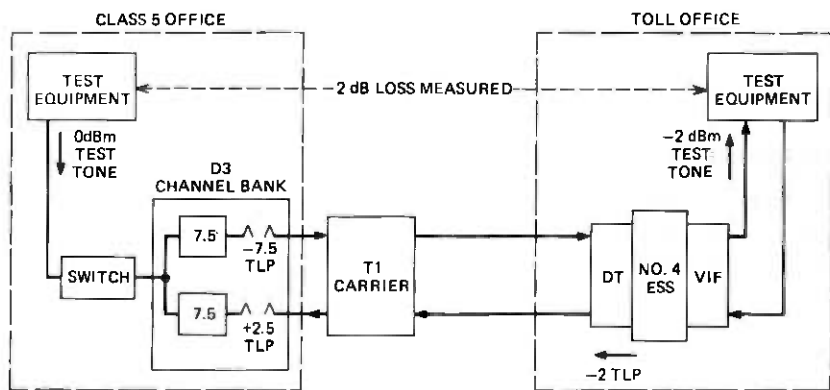
- (i) Toll connecting trunk loss was fixed at 3 dB.
- (ii) Digital echo suppressors were used on long digital trunks.
- (iii) All toll connecting trunks should achieve or exceed a terminal balance requirement of 22 dB on the average, and none less than 16 dB.

The provision of 3 dB toll connecting trunks in the SDN requires a change in level administration as shown in the following. The provision of 3 dB toll connecting trunks in the current analog network using T1 with D3 banks is shown in Fig. 4. (The receive test build-out pad is not included.) Note that in this case the receive level at the toll switch is 3 dB below the reference level, 0 TLP at the class 5 office; and the receive level at the class 5 office is 3 dB below the toll switch reference level, -2 TLP (0 TLP is often used on the toll connect side of toll switches instead of -2 TLP, but



\*HYBRID, PAD, AND OFFICE LOSS IN dB

(a)



TEST TONE ENCODED AS  
STANDARD DIGITAL MILLIWATT

(b)

Fig. 5—(a) Provision of 3-dB toll connecting trunk with No. 4 ESS defined as  $-2$  TLP. (b) Loss for toll connecting trunks when a DT is used with the No. 4 ESS defined as  $-2$  TLP.

for our purposes  $-2$  TLP will suffice to demonstrate the problem). This same technique can be used with the No. 4 ESS and VIF as demonstrated in Fig. 5a. The VIF in the figure is designated as a  $-2$  TLP, meaning that the VIF would encode a  $-2$  dBm tone into the standard digital milliwatt and would decode a digital milliwatt into a  $-2$  dBm tone. When a DT replaces the D3 and VIF as shown in Fig. 5b, the technique does not work. Note that a 0 dBm tone transmitted from the class 5 office would appear

as a  $-2$  dBm tone at the No. 4 ESS test position, indicating a 2 dB trunk instead of a 3 dB trunk.

To resolve this problem, one of the following would suffice: (i) redesignate the class 5 office as a  $+1$  TLP; or (ii) redesignate all toll offices as  $-3$  TLP. The first solution has been ruled out primarily because of the cost involved in modifying all class 5 test equipment. The second solution has been partially ruled out because of the cost of converting existing toll offices. However, it has been decided that No. 4 ESS offices will be designated as a  $-3$  TLP. This solves the problem in the long run, of providing 3 dB combination toll connecting trunks, as demonstrated in Fig. 6, where the VIF is now designated as a  $-3$  TLP. However, this so-

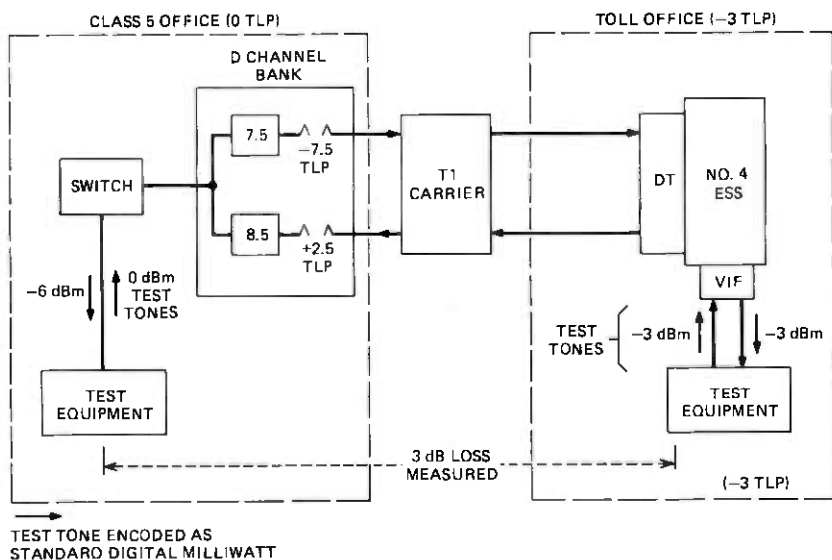


Fig. 6—Provision of 3-dB toll connecting trunks with DT and with the No. 4 ESS as  $-3$  TLP.

lution causes problems during the evolution of the SDN, because analog toll offices will still be a  $-2$  TLP. As a consequence, combination intertoll trunks will be designed to have 1 dB loss as illustrated in Fig. 7.

The loss plan for the SDN is designated the fixed-loss plan and specifies a fixed 3-dB loss for all toll connecting trunks, 0-dB loss for digital intertoll trunks, and 1 dB for combination intertoll trunks. Intertoll trunks utilizing analog facilities are designed to via net loss. The No. 4 ESS will operate at a  $-3$  TLP and existing analog switches will be at a  $-2$  TLP. This loss plan is summarized in Fig. 8. The plan for combination toll connecting trunks was shown in Fig. 6.

In conjunction with the above loss plan, digital echo suppressors will be used on digital intertoll trunks longer than 1850 miles.

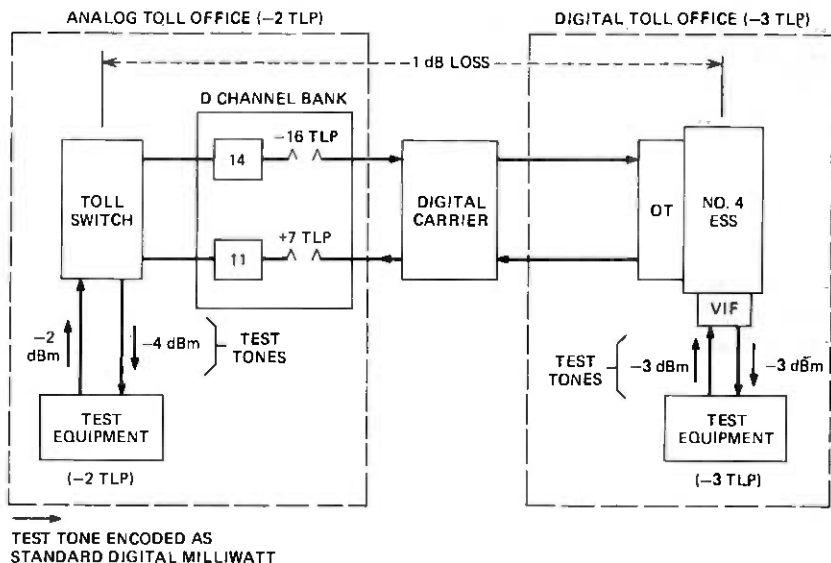


Fig. 7—Provision of intertoll combination trunks (1 dB only).

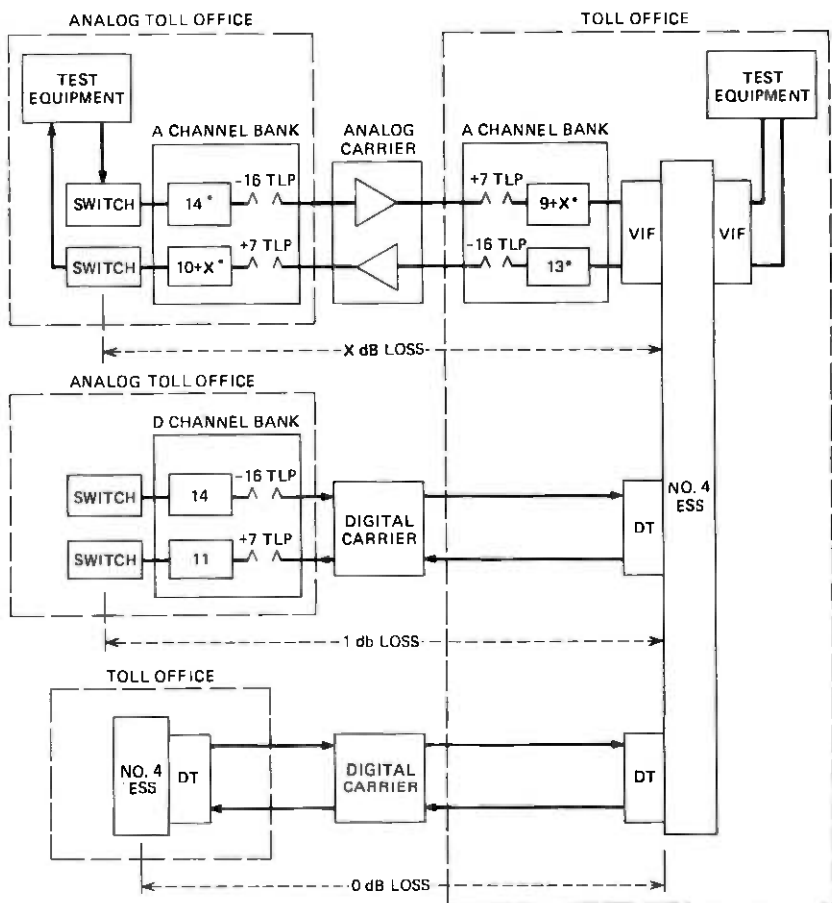
### 3.4 Timekeeping Plan

#### 3.4.1 A slip rate objective

Each No. 4 ESS switch will have a clock that controls its output frame rate as well as its internal timing. A signal to be transmitted over a digital trunk leaves the originating No. 4 ESS in digital form at a rate determined by that switch's clock. After transmission, the signal is read into a buffer in a digroup terminal at the terminating No. 4 ESS; the read-in rate is of course that determined by the originating switch. However, the signal is read out of the buffer at a rate controlled by the terminating switch's clock. It is clearly desirable that the read-in and read-out rates—or equivalently, the relative rate difference between No. 4 ESS clocks—be very nearly identical on the average. For example, if the read-out rate is too fast, then eventually the buffer will be scanned twice in succession while occupied by the same frame, i.e., the frame will be repeated. Conversely, if the read-out rate is too slow, then eventually a frame will be overwritten before being read out, i.e., the frame will be deleted. The phenomena just described, the repetition or deletion of an entire frame, are referred to as *slips*. Slips are one of the basic impairments to which signals in the SDN will be subjected. Like all other impairments, slips cannot be eliminated. But the objective of the time-keeping plan is to control slips to within tolerable limits.

Evidence with regard to the effect of slips on voice signals indicates that most slips are inaudible. However, the SDN will also carry voiceband





\* PAD AND OFFICE LOSS, X = DESIGNED INSERTED LOSS

Fig. 8—Provision of intertoll trunks within the SDN and between the SAN and SDN.

data signals and the effect of slips on these can be much more serious; a single slip can impair the operation of some data sets for several seconds. A specific objective that has been adopted for the SDN is based primarily on voiceband data requirements. The slip rate objective is at most one slip in 5 hours on an end-to-end connection. Based on a reference connection of two intertoll trunks, two toll connecting trunks (this is longer than for most intertoll calls), and accounting for the possibility of digital local offices, the slip objective is at most one slip in 20 hours per trunk. Since the duration of a frame is 125 microseconds, the slip rate objective leads immediately to a clock accuracy objective of 1.7 parts in  $10^9$  for the average relative rate difference between clocks. The specific goal of the timekeeping plan is to satisfy the clock objective. In fact, the

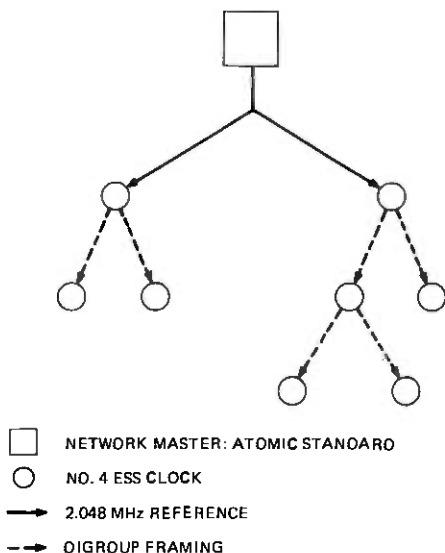


Fig. 9—Master-slave timing structure.

scheme to be described will do much better; under most conditions (including most carrier outages or failures) performance should be virtually slip free.

#### 3.4.2 Basic features of the timekeeping plan

The timekeeping plan provides for the following three basic features: (i) a master-slave hierarchical timing structure, (ii) stable local clocks at each No. 4 ESS, and (iii) the means to phase-lock a local clock to one of two different types of external reference timing signals.

The concept of master-slave synchronization of clocks is shown in Fig. 9. The arrows represent facilities over which timing information is carried. The direction of the arrows indicates the direction of timing flow so that a clock at the head of an arrow is "locked" to the clock at the tail. In the SDN, this locking can be described as a loose phase-locking. Care must be taken in setting up and administering the timing structure that a strict hierarchy is maintained; such anomalies as timing loops should not occur. This is an especially important consideration during a temporary reconfiguration of the hierarchy for maintenance purposes.

The clocks in Fig. 9, with the exception of the network master, represent the local No. 4 ESS clocks. These clocks have a measured frequency stability of better than one part in  $10^{10}$  per day. In the event of a failure on one of the timing links, the No. 4 ESS clock can "free-run" and in the free-run mode it would take in the worst case (linear drift with initial frequency offset) at least 3 days before a single slip occurred. On the other

hand, carrier failures typically last less than a few days. Thus, such failures could easily be bridged by free-running clocks.

The network master is not a No. 4 ESS clock; rather it is a standard reference obtained from the Bell System Reference Frequency Network (BSRFN). The BSRFN is being deployed to meet the stringent demands of the new high-capacity analog transmission systems such as L5. An atomic standard located in Hillsboro, Missouri near the geographical center of the country provides a precise reference frequency at 2.048 MHz.<sup>3,4</sup> This reference is transmitted without regeneration over analog cable and radio systems to regions throughout the country. Recent field trials have shown that reference frequency signals can be transmitted over cable and radio facilities for a thousand miles with a propagation error of less than one part in  $10^{11}$  over a 15-minute measuring interval.

The actual timing structure in the SDN will be based on a partition of the network into timing regions. The deployment of No. 4 ESS switches is expected to exhibit an initial clustering near metropolitan regions, with digital interconnection within each cluster but no digital interconnection between clusters. These clusters were referred to as SDN islands in Section 2.3. They are geographically separated areas of digitally interconnected No. 4 ESS switches and they form natural timing regions. One switch in each island is "elected" the master for that island, and the other switches in the island are slaved directly or indirectly to the master. The paths used for timekeeping purposes are a subset of paths in the communications network. Timing information appears on every path in the form of digroup framing bits. Thus, special equipment is needed to bridge onto a digital bit stream, detect the framing, and feed derived timing information to the clock.

With the introduction of medium- and long-haul digital facilities the previously established islands will become digitally interconnected. It will be expedient for administrative, maintenance, and performance reasons to retain separate timing regions. For example, such retention will place natural limits on the length of timing chains that can be formed. But there now has to be timekeeping between islands. For this purpose the master in each island will be slaved to the BSRF.

### **3.4.3 The clock and its interfaces**

Figure 10 shows in stylized form the No. 4 ESS clock and some of its interfaces. Note that the blocks shown as isolated units in the figure do not necessarily correspond to physically isolated hardware components.

The figure indicates three reference inputs; two are T1 lines and one is an analog system carrying the BSRF. A particular No. 4 ESS may have only two inputs. If the switch is a slave, the BSRF input would not appear.

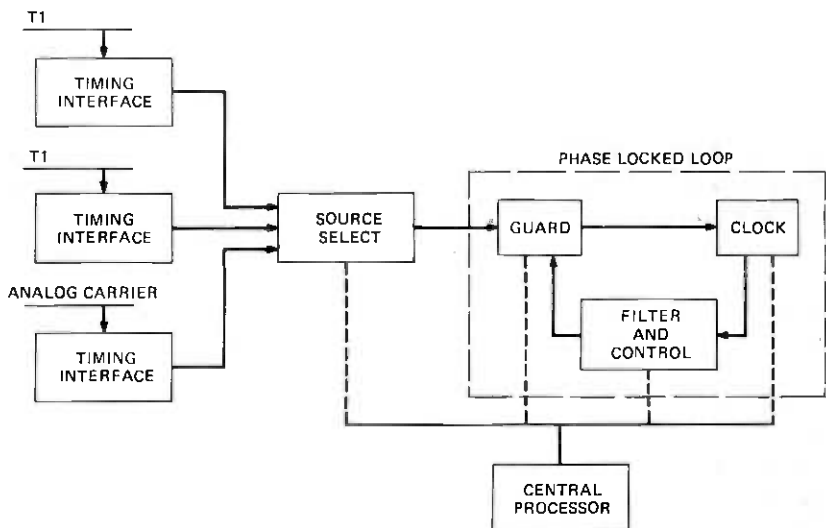


Fig. 10—No. 4 ESS clock interfaces.

But a slave should have two sources of timing via framing on digital communication paths, preferably over physically separated systems. One source would be used as a spare for maintenance purposes. If the switch is a master then the BSRF input is necessary. In addition, it would be desirable to have provision for a framing reference, again for maintenance purposes.

The blocks labeled *timing interface* accept as input an external reference: either the 2.048 MHz BSRF from analog carrier, or a digroup from, for example, T1 carrier. They produce as output an extracted timing signal at an integral subfrequency of the No. 4 ESS clock frequency. One of these extracted timing signals can be chosen by means of a source selector to provide the timing reference for the clock.

The next major subdivision of the figure is a digitally controlled phase-locked loop consisting of three blocks. The guard box performs several functions including detection of reference outage, initiation of the source select (via software), opening the phase-locked loop in order to make the clock free-run, and correction for sudden phase discontinuities due, e.g., to protection switches. The clock consists of four coupled crystal oscillators but can operate with only one working. Each oscillator has a frequency stability of better than one part in  $10^{10}$  per day. As mentioned previously, this degree of stability permits the clock to free-run and bridge most carrier outages or failures and yet produce virtually no slips. The final box labeled filter and control determines the dynamics of the loop.

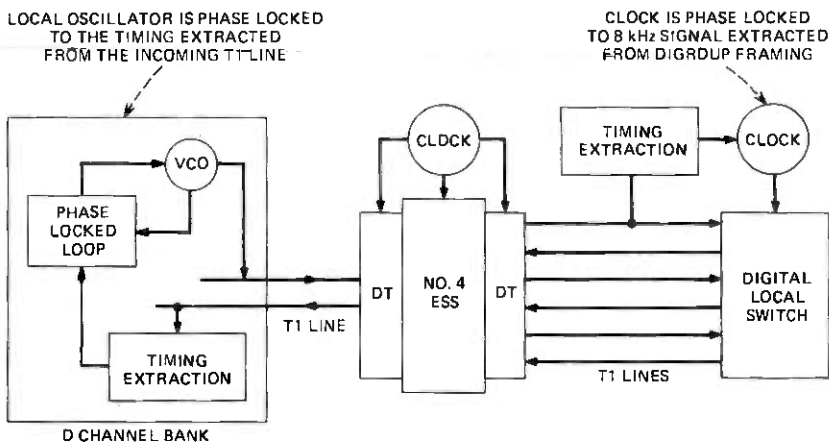


Fig. 11—Timing of channel banks and local digital switches when connected to the DT.

### 3.4.4 Channel banks and local switches

Other equipment will also interconnect with the No. 4 ESS as illustrated in Fig. 11 and will require accurate timing. D channel banks will be digitally connected to the No. 4 ESS. The special requirement for D channel banks is that they must have the means to be loop timed, i.e., to have the local oscillator phase-locked to the timing of the incoming signal and return this timing to the far-end No. 4 ESS. We must also account for the possible introduction of digital local switches and their digital interconnection with the No. 4 ESS. Recall that for the derivation of the slip rate objective for trunks, the toll connecting trunks in the reference connection were assumed to terminate on digital local switches. Thus, the introduction of digital local switches per se is consistent with the goal of the timekeeping plan, which is to control the end-to-end slip rate. A possible method for extending timekeeping to the exchange area is shown in Fig. 11.

## 3.5 Digital trunk maintenance plan

### 3.5.1 Special characteristics of digital trunks

The basic task of maintenance in any telecommunications network is to ensure adequate performance of the network. But maintenance activity in a digital network will be different than in an analog network for several reasons. One of the underlying reasons for this difference is that different performance parameters are measured on digital trunks. Transmission through an analog trunk is subject to noise, loss, echo, and various types of distortions. Transmission through a digital trunk is

subject to only three impairments: errors, misframes, and slips. An error is the erroneous substitution of a zero for a one or vice versa in the bitstream at any point in its transmission. Misframes can occur at a multiplex or digroup terminal, and are usually caused by the masking of framing bits by an error burst, or by the displacement or omission of framing bits due to protection switching in the upper levels of the digital hierarchy, or by propagation of misframes in other facilities in the hierarchy. During a misframe, the affected equipment searches the incoming bitstream for a bit with the right framing pattern; this search effectively produces a noise burst that can last tens of milliseconds. Slips, as defined in the previous section, cause an entire frame to be deleted or repeated. Transmission performance on digital trunks is thus characterized by (i) error rate, (ii) misframe rate, and (iii) slip rate. Jitter is not a separate problem since the digroup terminal buffer will absorb jitter. Excessive jitter can, however, result in slips, or, if the jitter is at a digital repeater, can result in errors. A digital trunk in an integrated digital network has two characteristics which can significantly simplify network maintenance:

(i) For voice circuits which reach a digital switch in multiplexed digital form, individual circuit access will not be available at the digroup terminal. However, per-circuit access is not necessary since a fault that affects a single circuit will almost always affect the digroup in which the circuit occurs. Thus, maintenance activity can be concentrated at the digroup level with a potential 24:1 increase in efficiency.

(ii) Digital systems can potentially monitor themselves. The predominant impairment will be errors, and it is possible to automatically obtain a measure of error rate while equipment is in service. Misframes and slips will be measured and reported by the digroup terminal. Thus emphasis can be placed on automatic surveillance of equipment as a means of controlling network performance.

### **3.5.2 Maintenance tasks**

Maintenance of a telecommunication network takes on different meanings depending upon the viewpoint. From a strictly operational viewpoint, a telecommunication network consists only of switches and trunks; and therefore, only two areas of maintenance—switch and trunk maintenance. However, from an equipment point of view, such a network consists of switches, transmission lines, multiplexers, channel banks, signaling gear, etc. Hence, from this viewpoint, the maintenance job involves the maintenance of this equipment when interconnected in a network. By combining both points of view, a telecommunication network has two basic areas of maintenance—trunk and equipment maintenance.

Up until now, these two areas of maintenance have been relatively

independent of each other. For the SDN, the plan is to merge trunk and equipment maintenance activity as much as it practicable.

Equipment maintenance consists of the following functions: trouble detection and sectionalization, removal of defective equipment from service, repair and return of equipment to service after repair. SDN maintenance of digital trunks will simplify these functions by requiring the partition of the network into maintenance spans. Each maintenance span will have a monitor at the end of the span so that unsatisfactory performance is detected. These monitors will provide an adequate measure of error rate and should provide a minor alarm when the error rate reaches a level which affects circuit performance and causes it to be degraded. This minor alarm signifies a need for corrective maintenance in order to restore good service within a reasonable time. A major alarm should also be provided when the error rate reaches a higher level which affects network operations. This alarm will cause the faulty system to be automatically removed from service. The major and minor alarm levels in the SDN are nominally  $10^{-3}$  and  $10^{-6}$  errors per bit, respectively.

Several additional techniques will be used to perform the maintenance functions. Automatic protection switching upon a major alarm substitutes a working set of equipment for the failed span. In the case of a multiple failure where a working spare is unavailable, the affected digroup must be removed from service. This will be accomplished by requiring at the failed span that a signal be inserted (such as the all-ones signal) that will inhibit maintenance alarms downstream, but will activate major alarms at the trunk ends. These major alarms will cause trunks to be automatically removed from service. Maintenance functions can also be aided by such centralized maintenance techniques as automatic remote monitoring and alarm pattern analysis because some equipment may not be monitored, and because of the possibility that some faults can produce ambiguous alarm indications.

In view of the above, the trunk maintenance activities of digital trunks will largely be concerned with network administration when faults occur. This administration is concerned with removal of malfunctioning circuits from service, verification of repair, and restoral to service. In addition, a digital test line that can measure error rates on individual digital trunks should be provided so that periodic end-to-end digital trunk testing can be performed as part of the trunk maintenance activity. Such testing will provide further assurance of proper performance and will detect the rare occurrence of single circuit failures.

#### **4. VOICE PERFORMANCE ON THE SDN**

##### **4.1 *Noise-loss grade of service***

In order to enter the SDN, an analog voice signal must be encoded into

digital form by a PCM interface terminal (D channel bank or VIF). The PCM encoding and subsequent decoding introduce noise and distortion into the voice channel. In a fully evolved digital toll plant, there will be only one such encoding and decoding on each connection. However, during the evolution of the SDN, when a connection may use a mixture of analog and digital facilities, a path may enter and leave the SDN several times, being subjected to further PCM noise and distortion each time. During the same period of evolution, the introduction of digital long-haul transmission facilities may be expected to reduce the accumulation of noise on long-haul connections, since digital transmission systems do not accumulate such impairments. The question addressed in this section is the evaluation of the net effect of these two competing tendencies, one tending to degrade the grade of service and other tending to improve it.

This section describes the results of computer simulation studies designed to predict the combined noise-loss performance of the DDD network as the SDN evolves. The performance measure is voice grade of service of the transmission quality of telephone connections. The simulation model that was used accounted for the deployment of No. 4 ESS, digital transmission facilities, PCM interfaces, and analog switching and transmission facilities including improved analog carrier such as L4, L5, and TD3.

The computer simulation model used for the DDD network is a modification of a model developed by T. C. Spang.<sup>2</sup> The approach used in the computer model was to duplicate within the computer the routing and transmission characteristics that could have occurred on a sample of actual calls in the network. From this information, estimates of the distributions of loss and noise were obtained. To these were added an estimate of an equivalent amount of additive noise contributed by a D channel bank or VIF. Estimates of customer opinion were obtained by subjective tests<sup>5</sup> in which customers rated a call having a given set of transmission parameters on a scale of "excellent," "good," "fair," "poor," and "unsatisfactory." By combining this information with the distributions of occurrences of the parameter, estimates were obtained of the expected percentage of customers who would rate a call "good or better" (good and excellent) or "poor or worse" (poor and unsatisfactory) if a large number of calls are made. These estimates are referred to as "grade of service."

#### **4.2 Results**

For the future, as the SDN evolves in the DDD network, the predicted noise-loss grade of service expressed in percent of subscribers rating transmission quality both "good or better" and "poor or worse" is illustrated in Fig. 12. The grade of service is a function of connection



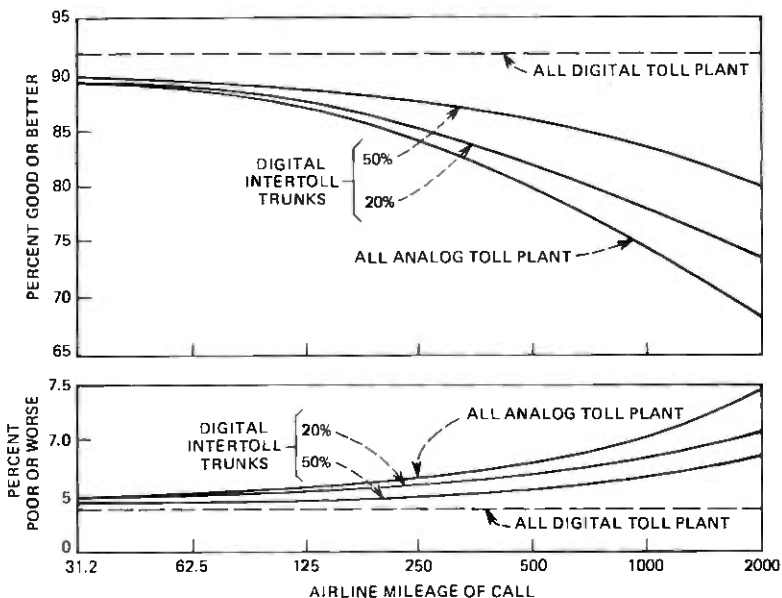


Fig. 12—Noise-loss grade of service of switched network with evolving digital facilities as a function of connection distance.

distance (airline mileage). The curves shown are smoothed versions of the computed results. The “all-digital toll plant” refers to a hypothetical situation when all voice circuits are carried from end office to end office in digital form without intermediate analog-to-digital conversion. The grade of service is predicted to improve as the amount (percentage) of digital connectivity in the toll plant increases. In Fig. 12, two examples (20 percent and 50 percent intertoll trunks are digital) are illustrated. The improvement is more substantial for connections greater than a few hundred miles than for shorter connections. The evolution of digital facilities improves grade of service and reduces the contrast between long and short connections; that is, the improvement is greatest where grade of service is least, at the long distances. For the all-digital toll plant case, no contrast is found between long and short calls.

## 5. SUMMARY AND FUTURE TRENDS

The introduction of No. 4 ESS and the initial implementation of the SDN mark a transition from existing practices in intertoll telephony to a new intertoll network which will continue to evolve. The plan outlined here has been subject to debate and challenge throughout the development cycle of the No. 4 ESS machine and has been maintained essentially unchanged. Projected economies of digital termination on No. 4 ESS switches are being realized in initial installation. Projected economies

in maintenance are being realized also. The difficult transition from current methods of facilities utilization and trunk provisioning to the new methods required by digroup engineering have been accomplished, and improvements in circuit record data base administration are reasonable expectations. Improvements in voice performance remain a reasonable expectation, but years of evolution will be required to demonstrate them.

One note of caution, however, must be mentioned. A number of planning opportunities or alternatives have been used up by decisions reached to formulate the plan. Such an example is the adoption of a unified loss and level plan which restricts independent evolution of loss planning and level planning. Future technological innovation will be constrained to some degree by existence of this plan for an integrated digital transmission and switching network.

#### ACKNOWLEDGMENTS

Many contributions have been made by people too numerous to mention. Special mention, however, must be made of the contributions by C. C. Case to timekeeping and maintenance, by M. B. Brilliant to digital transmission objectives, and by W. A. Stubblebine, who defined the idle code plan.

#### REFERENCES

1. C. L. Damman, L. D. McDaniel, and C. L. Maddox, "D2 Channel Bank: Multiplexing and Coding," *B.S.T.J.*, 51, No. 8, (October 1972), pp. 1675-1699.
2. T. C. Spang, "Loss-Noise Echo Study of the Direct Distance Dialing Network," *B.S.T.J.*, 55, No. 1 (January 1976), pp. 1-36.
3. R. E. Powers, "Reference Frequency Transmission Over Bell System Radio and Coaxial Facilities," *Proc. 28th Annual Symp. on Frequency Control*, pp. 373-379, May 29-31, 1974, U.S. Army Electronics Command, Fort Monmouth, New Jersey.
4. J. F. Oberst, "Keeping Bell System Frequencies on the Beam," *Bell Laboratories Record*, 52, No. 3 (March 1974), pp. 84-89.
5. J. L. Sullivan, "Is Transmission Satisfactory? Telephone Customers Help Us Decide," *Bell Laboratories Record*, 52, No. 3 (March 1974), pp. 90-98.

## Contributors to This Issue

**John E. Abate**, B.S.E.E., 1954, Newark College of Engineering; M.S.E.E., 1960, Stevens Institute of Technology; Sc.D.E.E., 1967, Newark College of Engineering; Bell Laboratories, 1963—. Mr. Abate has worked on problems in adaptive quantization and prediction of speech and video signals, digital transmission systems analysis, and digital network performance studies. Since 1970, he has been supervisor of a Digital Network Planning Group responsible for planning the evolution of digital switching and transmission facilities into the Bell System network. Member, IEEE, Sigma Xi, Tau Beta Pi, Eta Kappa Nu, and AAAS.

**J. F. Boyle**, B.S.E.E., 1963, Newark College of Engineering; M.S.E.E., 1965, Columbia University; Bell Laboratories, 1963—. Mr. Boyle has worked on physical design of utilized facility terminals, digroup terminals, and switched maintenance access systems and on voice-frequency applications. He was assistant engineering manager—transmission for AT&T for two years. Currently, he is supervisor of the voice-frequency equipment group at Bell Laboratories. Member, Tau Beta Pi, Eta Kappa Nu.

**Lane H. Brandenburg**, B.S.E.E., 1962, M.S.E.E., 1963, Ph.D., Columbia University; Bell Laboratories, 1963—. As a member of the Digital Network Planning Department, Mr. Brandenburg is concerned with the orderly introduction of digital systems into the nationwide Bell System telecommunications network, with emphasis on special requirements and compatibility issues.

**Thomas J. Cieslak**, B.S. (math) 1965, Illinois Institute of Technology; M.S.E.E., 1966, University of California (Berkeley); Bell Laboratories 1965—. Mr. Cieslak began at Bell Laboratories working on No. 1 ESS arranged for AUTOVON in the fault-recovery area. Since 1968 he has been engaged in software design on time-division network control and CCIS security arrangements for No. 4 ESS and is currently working on No. 4 ESS peripheral-fault recovery. Member of ACM.

**John R. Colton**, B.S.E.E., 1964, University of Pennsylvania; M.S.E.E., 1966, MIT; Bell Laboratories, 1964—. Mr. Colton has been engaged in the development of hardware and maintenance software for equipment that directly terminates digital carrier systems onto No. 4 ESS. In 1972 he became supervisor of a group responsible for system design, planning, and maintenance of the Digroup Terminal. He is currently supervisor of a group responsible for the development of remote access and test systems for special service circuits. Member, IEEE, Eta Kappa Nu, Tau Beta Pi, Sigma Xi.

**Harry B. Compton**, B.E.E., 1950, M.E.E., 1955, Polytechnic Institute of Brooklyn; Bell Laboratories, 1955—. His early work was in circuit design for radar systems including the DEW Line and Nike projects. He then worked on logic and system design on the UNICOM and AUTOVON communication systems. Most recently he supervised a group responsible for field testing the first No. 4 ESS in Chicago. Member, Tau Beta Pi, Eta Kappa Nu, IEEE; associate member, Sigma Xi.

**Lance M. Croxall**, B.S.E.E., 1965, M.S.E.E., 1966, Purdue University; Bell Laboratories 1966—. Mr. Croxall was initially responsible for the development of the network management program for the electronic translator system for 4A crossbar. He later supervised groups responsible for operational program development for No. 4 ESS, including input/output, traffic measurements, and CCIS. He currently supervises a group responsible for international call handling features for No. 4 ESS. Member, Tau Beta Pi, Eta Kappa Nu.

**Carl L. Dammann**, B.S.E.E., 1961, University of Maryland; M.S.E.E., 1963, New York University; Bell Laboratories, 1961—. Mr. Dammann has been involved in digital terminal design, including design of wide-band coding terminals and digital channel banks. Currently supervising digital terminal design. Member, IEEE.

**Paul K. Giloth**, B.A., 1942, Beloit College; B.S.E.E., 1947, Northwestern University; Illinois Bell Telephone Company, 1947-1950; Bell Laboratories, 1951—. Mr. Giloth worked initially on analog computer simulators for military applications. Following this he supervised development of TRADIC, a transistorized bombing and navigation system, and the guidance computers for the NIKE-ZEUS ABM system. In 1961 he was appointed head of the UNICOM Test Model Department and was

responsible for digital terminal equipment and the store-and-forward message portion of the UNICOM system. In 1963 he became head of the Data Switching Systems Department and was responsible for development of the No. 1 ESS ADF Data Switching System. From 1970 to 1972 he was responsible for AUTOVON development. As head of the No. 4 ESS Coordination and Evaluation Department, he is now responsible for developing software tools for system evaluation and the planning, engineering, and integration of early No. 4 ESS offices. Senior Member, IEEE; Member, Sigma Xi.

**John A. Giunta**, B.S.E.E., 1965, University of Dayton; M.S.E.E., 1967, Columbia University; Bell Laboratories, 1965—. Mr. Giunta initially worked on studies of military communications systems. In 1967 he became involved in the system and hardware design of trunk maintenance systems, including test positions and the Switched Maintenance Access System (SMAS). He currently supervises the group responsible for the development of the Circuit Maintenance System (CMS) 1A. Member, Tau Beta Pi.

**T. V. Greene**, B.S.E.E. 1957, Polytechnic Institute of Brooklyn; M.S.E.E. 1959, New York University; Bell Telephone Laboratories, 1957—. Mr. Greene's early work on automation of design processes for the manufacture of Nike-Zeus computer systems was followed by processor design for the UNICOM system and software development for AUTOVON switching offices. He supervised the software development of features for the AUTOVON and No. 4 ESS systems. He currently supervises the development of network management and traffic administration features for No. 4 ESS.

**D. G. Haenschke**, Diplom Vorpruefung 1951, Dipl. Ing. 1953, Technical University, Vienna; Institute for Telecommunications, Vienna University, 1953-1955; Bell Telephone Laboratories, 1955—. He was engaged in the planning of communications systems for air traffic control, and later in traffic studies applied to data communication systems and a long-range toll network planning. During 1968 Mr. Haenschke supervised a group responsible for performance appraisal of new services, including *PICTUREPHONE*® service. Since 1969 he has been responsible for the planning of traffic network management and overload control features for new switching systems such as No. 4 ESS. He is currently supervising a group engaged in the feature planning of computer-based centralized network management surveillance and control

systems. Senior member, IEEE; member, AAAS, Verband Deutscher Elektrotechniker.

**Michael B. Haverty**, B.S.E.E., 1960, University of North Dakota; M.S.E.E., 1962, New York University. Bell Laboratories, 1960—. Mr. Haverty has worked on many phases of electronic switching development including circuit and logic design, program design, and system testing. From 1965 to 1972, he was associated with the development of the No. 1 ESS used in the government AUTOVON communications network. Since 1972, he has been head of the department responsible for development and operation of No. 4 ESS test facilities. Member, IEEE, Sigma Tau, and Sigma Xi.

**S. F. Heath III**, B.S., 1965, University of Minnesota; M.S., 1966, Stanford University; Bell Laboratories, 1965 to 1976; AT&T, 1976—. Mr. Heath has been engaged in program development for No. 1 ESS, No. 2 ESS, and No. 4 ESS. In No. 4 ESS, he has supervised the Data Administration Group and later the Toll System Evaluation and Utility Group. Mr. Heath is currently the Assistant Engineering Manager—Local Switching, AT&T. Member, Phi Beta Kappa.

**Barbara H. Hornbach**, A.B. (mathematics) 1969, Vassar College; M.S. (information science) 1972, University of Chicago; Bell Laboratories, 1969—. Ms. Hornbach has been responsible for the development of operational software for the No. 4 ESS network management and traffic and plant measurement systems. Currently she is designing software for No. 4 ESS international trunk maintenance features.

**John H. Huttenhoff**, B.S.E.E., 1960, University of Illinois; MSEE, 1962, New York University; Bell Laboratories, 1960—. Mr. Huttenhoff worked on the design of new computer systems for military applications from 1960 to 1971, when he joined the No. 4 ESS project. He has worked on the network clock for No. 4 ESS and is presently supervisor of the Peripheral Recovery Program Group.

**John Janik, Jr.**, B.S.E.E., 1954, New York University; 1954–1957 Communications Development Training Program; 1958 Digital Techniques Laboratory Program; Bell Laboratories 1954—. Mr. Janik is head of the Peripheral Circuit and Physical Design Department on No. 4 ESS.

He has worked on several military projects, UNICOM, AUTOVON, and No. 4 ESS. His present responsibilities are for the network and peripheral unit designs in No. 4 ESS. He is a member of the IEEE, Eta Kappa Nu, and Tau Beta Pi.

**Carl E. Johnson**, B.S.C.E., 1957, Syracuse University; M.S.C.E., 1964, University of Southern California; Jet Propulsion Laboratory, 1959–1965; Bellcomm, Inc., 1965–1972; Bell Laboratories, 1972—. Mr. Johnson has worked on systems studies concerned with advanced manned missions to Mars in support of the Apollo lunar project. At Bell Laboratories, he has worked on systems tasks on a computerized maintenance and administration system. Most recently he has worked on the requirements for the No. 4 ESS engineering and administration reports.

**G. Douglas Johnson**, B.S.E.E., Texas A&M University; Bell Laboratories 1936—. Mr. Johnson was involved in the development of mobile communications systems, radar, analog-digital converters, and data communications. For a number of years he has been concerned with toll switching systems, particularly evaluating and selecting the time-division network, as well as other design activities related to No. 4 ESS. He is supervisor of the Special Studies and Planning Group in the AUTOSEVOCOM II Development Laboratory.

**Barry J. Karafin**, B.S.E.E., 1961, University of Pennsylvania; M.E.E., 1963, New York University; Ph.D., 1974, University of Pennsylvania; Bell Laboratories, 1961—. Mr. Karafin has worked in the areas of digital transmission systems, computer language and compiler design, and computer-aided circuit analysis. His recent work has been in the design of controllers for digital switching system peripherals. Presently, he heads a department developing business customer telephone systems.

**John C. Lawson**, B.S.E.E., 1962, Yale University; M.S.E.E., 1967, The George Washington University; Bell Laboratories, 1967—. Mr. Lawson has worked on systems engineering and economic problems concerned with toll and local digital switching. He is currently engaged in studies of provision of digital services.

**Henry J. Luer**, B.S.M.E., 1956, Newark College of Engineering; Bell

Telephone Laboratories, 1956—. Mr. Luer's work has included development of guided missile inertial instruments and guidance systems. He presently supervises a group engaged in the physical design of power equipment.

**H. Mann**, B.A., 1950, Brooklyn College; M.S.E.E., 1955, Columbia University; Bell Laboratories, 1954—. Mr. Mann's early work was in exploratory development of framers and coders for T1 carrier. Subsequently he worked on the comand decoder for Telstar, on development of signaling units and the 4A signaling test set, and on development of the Digroup Terminal. Currently he is engaged in development of the No. 4 ESS digital interface frame. Member, IEEE and Pi Mu Epsilon.

**Peter S. McCabe**, B.S. (engineering), 1956, Trinity College; B.S.E.E., 1957, Rensselaer Polytechnic Institute; Bell Laboratories, 1957—. Mr. McCabe has worked on memories for Nike-Zeus, digital circuits for the 101 EPBX, and software design for the UNICOM, AUTOVON, and four-wire No. 1 ESS projects. Since 1967, Mr. McCabe has worked on support software system design and program administration for No. 4 ESS. He currently supervises a group that administers the No. 4 ESS program and designs support software for the No. 4 ESS development.

**Michael N. Meyers**, B. A., 1967, University of Illinois; M. S., 1969, University of Chicago; Bell Laboratories, 1967—. Mr. Meyers first worked on program administration and the office parameter data assembler for No. 1 ESS ADF. From 1970 until recently he has been involved with many aspects of No. 4 ESS diagnostic design and development. Currently he is engaged in advanced fault-recovery studies. Member, IEEE, Pi Mu Epsilon.

**Billy G. Niedfeldt**, B.S.E.E., 1959, University of Maryland; M.S.E.E., 1961, New York University; Bell Laboratories, 1959-1962; Bellcomm, Inc., 1962-1970; Bell Laboratories, 1970—. Mr. Niedfeldt's initial assignment at Bell Laboratories was exploratory development of high-speed data terminals. His assignments at Bellcomm were guidance and control system analysis, Apollo lunar landing mission error analysis, and determination of an engineering model of the lunar gravitational potential field. After rejoining Bell Laboratories in 1970, his initial assignments were in the Safeguard project where he was responsible for the operations of the tactical system laboratories and for the engineering,



planning, and coordination of the Madison, N.J., building. Later he was responsible for the system test and integration of the tactical operating system. In 1974, Mr. Niedfeldt joined the No. 4 ESS project where his first assignment was the responsibility of the first generic software package system test. He currently supervises a group responsible for No. 4 ESS system laboratory operations and utility system development.

**Roman Ostapiak**, B.S.E.E., 1961, University of Pennsylvania; M.S.E.E., 1963, New York University; Bell Telephone Laboratories, 1961—. Since joining Bell Telephone Laboratories, Mr. Ostapiak has been engaged in the design of power-conditioning circuits. Presently he is a supervisor of a group responsible for the design of common power equipment.

**John B. Otto**, B.S. (mathematics), 1959, St. Norbert College; M.S., 1966, Stevens Institute of Technology; Western Electric Company, 1959–1963; Bell Laboratories, 1963—. Mr. Otto has been involved with acceptance testing for the SAGE project, programming for No. 1 ESS AUTOVON and design and development of utility, support, and test programs for No. 4 ESS.

**James T. Raleigh**, B.S. (engineering science), 1957, Pennsylvania State University; Bell Laboratories, 1957—. Mr. Raleigh initially worked on precision frequency standards and portable transmission test equipment. In 1962 he transferred to Bellcomm Inc., where he was involved with system engineering and analysis in support of the manned lunar landing. He later supervised a group concerned with spacecraft and ground communications. Upon his return to Bell Laboratories in 1972, Mr. Raleigh supervised a group involved with switched special services and later the transmission aspects of operator and attendant services. He currently supervises the group responsible for the development of the No. 4 ESS 51A test position and test positions for 4A crossbar and special services. Member, IEEE, Tau Beta Pi.

**A. E. Ritchie**, A.B. 1935, M.A. 1937 (physics), Dartmouth College; Bell Laboratories, 1937—. His initial work was in designing crossbar switching systems and in organizing courses, writing texts, and teaching in the fields of radar, switching circuits, and system design. Since 1951, his assignments have been in systems engineering and fundamental planning in the telephone switching field. He became director of

Switching Systems Engineering in 1958 and, when responsibilities were divided in 1965, director of Toll Switching Engineering. This position includes responsibility for planning studies leading to new toll switching systems such as No. 4 ESS, and planning for development and implementation of new signaling systems, notably Common Channel Inter-office Signaling. Senior member, IEEE.

**Joyce B. Roberts, B.S.**, 1957, Dickinson College; Bell Laboratories, 1961—. As a resident visitor at Bell Laboratories, she worked on data reduction and analysis of the SAGE air defense project. She joined Bell Laboratories to work on UNICOM, a military communication system, where her specific assignments included the code simulator, traffic and supervisory consoles, and hot-line feature. She later worked on AUTOVON call processing and office parameters. Since 1968, she has been involved with call processing for No. 4 ESS with prime responsibilities for the numbering plan, digit reception and analysis, and service evaluation. At present she is in the International Call Processing Design Group.

**William L. Ross, B.A. (chemistry)**, 1948, **M.A. (physics)**, 1950, **Ph.D. (physics)**, 1953, University of British Columbia; Bell Laboratories, 1955—. Mr. Ross conducted systems engineering studies on negative impedance loading, T1 carrier, and digital transmission on coaxial and waveguide systems. Since 1965, he has worked on network modeling algorithms for optimal transmission facilities deployment, on toll network planning, and on initial planning for No. 4 ESS in the evolving network.

**William A. Routt, B.S.E.E.**, 1966, Pennsylvania State University; **M.S.E.E.**, 1967, Carnegie Institute of Technology; Bell Laboratories, 1966—. Mr. Routt initially worked on requirements for No. 4 ESS and then worked on the design and development of the software integrity programs for No. 4 ESS. Currently, he is supervisor of the group that is responsible for international trunk maintenance software for No. 4 ESS. Member, Eta Kappa Nu, Sigma Tau.

**Sumit Roy, B.E.E.E.**, 1969, Calcutta University; **M.S.E.E.**, 1965, **Ph.D.**, 1970, Illinois Institute of Technology; Bell Laboratories, 1970—. Mr. Roy was involved with the development of automated software testing methods for No. 4 ESS software. He has also worked on trouble-

locating procedures, data-base management, and programming productivity issues related to No. 4 ESS. He is supervisor of the Advanced Software Design Group. Member, Tau Beta Pi, IEEE, Sigma Xi.

**Michael W. Saad**, B.S.E.E., 1964; M.S.E.E., 1965, University of Illinois; Bell Laboratories, 1966—. Mr. Saad first worked in the maintenance software area for the AUTOVON system. His No. 4 ESS experience includes MF and DP call-processing design, peripheral error-analysis strategies and design, and peripheral-unit-configuration program design. Mr. Saad is currently working on No. 4 ESS translations for international switching features. Member, Tau Beta Pi.

**Jack M. Scanlon**, B.A.S.C., 1964, University of Toronto; M.S., 1965, Cornell University; Bell Laboratories, 1965—. Mr. Scanlon initially worked on the development of the signal processor for No. 1 ESS and No. 1 ESS peripheral-fault recovery and diagnostics programs, then supervised groups responsible for the addition of new call features to No. 1 ESS and No. 4 ESS call-handling design. He currently heads a department responsible for the No. 4 ESS system design, design of No. 4 ESS call-handling software, program administration for No. 4 ESS, and exploration of new software development techniques. Member, IEEE.

**Walter R. Schleicher**, B.S.E.E., 1955, Newark College of Engineering (New Jersey Institute of Technology); Bell Laboratories 1941—. In the earlier part of his career, Mr. Schleicher worked on the design and development of telephone apparatus and transmission measuring equipment. In 1956, he was involved in the design and development of a naval air-to-ground data communications system. This was followed by UNICOM, a military electronic switching system. Since 1962, he has supervised physical design groups working on the design and development of electronic switching systems—AUTOVON and No. 4 ESS. Member, Tau Beta Pi.

**Gabe A. Sellers, Jr.**, B.S.E.E. Kansas State University, 1948; Bell Laboratories 1948—. Mr. Sellers has been associated with the development of military data communications, Nike-Zeus digital computers, UNICOM, AUTOVON, and four-wire No. 1 ESS. He has been associated with the No. 4 ESS project since 1967. He currently supervises a group responsible for No. 4 ESS system design and planning.

**Matthew F. Slana**, B.S.E.E., 1957, University of Notre Dame; M.S.E.E., 1958, University of Wisconsin; Bell Laboratories 1958—. Mr. Slana first worked in the Military Computer Research Laboratory, investigating high-speed logic circuits, cryotrons, and radiation effects on devices and circuits. He next worked in the Military Communications Engineering Laboratory, engineering communication links for Nike and SAFEGUARD missile systems, and was a member of a Minuteman Missile Safety Study team. Since 1964, he has been involved with the development of a variety of electronic switching systems, including responsibility for the design of the time-division network for No. 4 ESS. Currently, he supervises the System Design Group in the AUTOSEVOCOM II Development Laboratory. Senior member, IEEE, Eta Kappa Nu.

**Muerl T. Smith, Jr.**, B.S.E.E., 1961, Clarkson College of Technology; M.S.E.E., 1963, New York University; Bell Laboratories, 1961—. Mr. Smith initially worked in the development of operational software for the UNICOM and later the No. 1 ESS ADF systems. He then supervised the group responsible for the development of the No. 4 ESS trunk and service circuit maintenance software. Since 1976 he has been involved with No. 4 ESS call processing and system architecture requirements and design. Member, IEEE, Eta Kappa Nu, Tau Beta Pi.

**Albert E. Spencer, Jr.**, B.S.E.E., Drexel University, 1951; Bell Laboratories, 1951—. Mr. Spencer initially participated in the design of transmission systems, then had several development and systems engineering assignments related to military and government communications, and to central office and customer switching. In 1971, Mr. Spencer was appointed director of the Customer Switching Laboratory, Denver. He assumed his present position as executive director of the Toll Electronic Switching Division, Naperville, Illinois, in 1975.

**Frank H. Tendick, Jr.**, B.S.E.E., 1951, University of Michigan; Bell Laboratories, 1951—. Mr. Tendick was first engaged on exploratory transistor circuits for airborne military computers and submarine cable amplifiers. He has supervised groups on development of twistor stores for Nike Zeus, time-division switch units and control circuits for No. 101 ESS PBX, and the signal processors for No. 4 ESS. At the present, he is supervisor of the No. 4 ESS System Laboratory and Test Facilities Group. Member IEEE, Tau Beta Pi, Phi Kappa Phi, Eta Kappa Nu, and Sigma Xi.

**L. S. Tuomonoksa**, B.S.E.E. 1952, Worcester Polytechnic Institute; M.S.E.E. 1954, Massachusetts Institute of Technology; Bell Laboratories, 1954—. His first assignment was in the development of the Morris experimental electronic switching system. At the completion of the Morris trial, he was assigned to development of No. 1 ESS, responsible for the maintenance planning and design for that system. In 1966 he was appointed department head responsible first for maintenance planning and later for operational program design. In 1968, he became department head of the No. 4 ESS System Planning and Operational Program Design Department. In April of 1974 he was appointed assistant director of the No. 4 ESS Switching System Laboratory and in March 1976, the director of that laboratory. In May 1977, he became director of the No. 1 ESS Switching System Laboratory.

**Keith W. Yoder**, B.S.E.E., 1966, University of Cincinnati; M.S., 1968, Northwestern University; Bell Laboratories, 1966—. Mr. Yoder initially worked on system initialization and manual interface programs for No. 1 ESS arranged for data features. He also worked on No. 4 ESS peripheral fault recovery. He is currently working on international switching features for No. 4 ESS. Member, Tau Beta Pi and Eta Kappa Nu.

**Kenneth W. Zweifel**, B.S.E.E., 1962, Washington University (St. Louis); M.S.E.E., 1963, Carnegie Institute of Technology; Bell Laboratories, 1962—. Mr. Zweifel has worked on diagnostic program requirements for No. 1 ESS ADF and the development of a system for control and reporting of switching system projects. He had responsibility for the design of the software system and overall system design responsibility for the Programmed Electronic Traffic Simulator.



# Papers by Bell Laboratories Authors

## BIOLOGY

**A Quantitative Model of Actin-Myosin Interaction in Skeletal Muscle.** A. Gersho, M. Orentlicher, *Biophys. J.*, **18** (May 1977), pp. 141-159.

**The Binding of CO to Mutant Alpha Chains of Hemoglobin M. Iwate Evidence for Distal Imidazole Ligation.** J. Peisach, K. Gersonde, *Biochemistry*, **16** (1977), pp. 2539-2545.

**An Experimental Test of the Vibronically Coupled Tunneling Description of Biological Electron Transfer.** J. J. Hopfield, M. J. Potasek, *Proc. Nat. Acad. Sci.*, **74** (January 1977), p. 229-233.

**<sup>31</sup>P Nuclear Magnetic Resonance Studies o-Ehrlich Ascites Tumor Cells.** G. Navon, S. Ogawa, R. G. Shulman, T. Yamane, *Proc. Nat. Acad. Sci.*, **74** (January 1977), pp. 87-91.

## CHEMISTRY

**Classical Trajectory Study of the Dissociation of Hydrogen on Copper Single Crystals II. Cu(100) and Cu(110).** A. Gelb, M. J. Cardillo, *Surface Sci.*, **64** (April 1977), pp. 197-208.

**Photochemistry of the Sunday Effect.** T. E. Graedel, L. A. Farrow, T. A. Weber, *Environ. Sci. Technol.*, **11** (1977), pp. 690-694.

**Oxidation of Ni<sub>3</sub>Sn<sub>4</sub>, Metastable SnNi, Ni<sub>3</sub>Sn<sub>2</sub> and Ni<sub>3</sub>Sn.** H. G. Tompkins, J. E. Bennett, *J. Electrochem. Soc.*, **124**, No. 4 (April 1977), pp. 621-623.

**Picosecond Dynamics of Azulene.** E. P. Ippen, C. V. Shank, R. L. Woerner, *Chem. Phys. Lett.*, **46**, No. 1 (February 1977), pp. 20-23.

## COMPUTING

**LTX—A Minicomputer-Based System for Automated LSI Layout.** G. Persky, D. N. Deutsch, D. G. Schweikert, *Journal of Design Automation and Fault-Tolerant Computing*, **1**, No. 3 (May 1977), pp. 217-255.

**Unconventional Uses of On-Line Information Retrieval Systems: On-Line Bibliometric Studies.** D. T. Hawkins, *J. Amer. Soc. Inform. Sci.*, **28**, No. 1 (1977) pp. 13-18.

## ECONOMICS

**Portfolio Choice and Equilibrium in Capital Market with Safety-First Investors.** E. R. Arzac, V. S. Bawa, *Journal of Financial Economics*, **4** (May 1977), pp. 277-288.

## ELECTRICAL AND ELECTRONIC ENGINEERING

**IMPATT Diode Circuit Design for Parametric Stability.** J. Gonda, W. E. Schroeder, III, *Trans. Microw. Theory Tech.*, **MTT 25** (May 1977), pp. 343-352.

**Interpolative Picture Coding Using a Subjective Criterion.** A. N. Netravali, *IEEE Trans. Comm.*, **COM-25**, (May 1977), pp. 503-508.

**Adaptive Quantization of Picture Signals Using Spatial Masking.** A. N. Netravali, B. Prasada, *Proc. IEEE*, **65**, No. 4 (April 1977), pp. 536-548.

**Simultaneous Phase Tracking and Detection in Data Transmission Over Noisy Dispersive Channels.** F. R. Magee, Jr., *IEEE Trans. Comm.*, **COM-25** (July 1977), pp. 712-715.

**Rank Reduction of Ill-Conditioned Matrices in Waveguide Junction Problems.** D. N. Zuckerman, P. Diament, *IEEE Trans. Microw. Theory Tech.*, *MTT-25* (July 1977), pp. 613-619.

**Some Observations of Anomalous Depolarization on 19 and 12 GHz Earth-Space Propagation Paths.** D. C. Cox, H. W. Arnold, A. J. Rustako, Jr., *Radio Sci.*, *12* (May-June 1977), pp. 435-440.

**On Low-Sensitivity Realizations of Band-Elimination Active Filters.** R. N. Gadenz, *IEEE Trans. on Circuits and Systems*, *CAS-24*, No. 4 (April 1977), 175-183.

**Rain-Scatter Interference on an Earth-Space Path.** T. S. Chu, *IEEE Trans. Ant. Propag.*, *AP-25* (March 1977), pp. 287-288.

## GENERAL MATHEMATICS AND STATISTICS

**Applications of Group Actions on Finite Complexes to Hilbert Cube Manifolds.** L. S. Newman, Jr., *Proc. Amer. Math. Soc.*, *62*, No. 1 (January 1977), pp. 183-187.

**Charge Singularity at the Vertex of a Slender Cone of General Cross Section.** J. A. Morrison, *Siam J. Appl. Math.*, *33* (July 1977), pp. 127-132.

## MATERIALS SCIENCE

**Beryllium Doping and Diffusion in Molecular Beam Epitaxy of GaAs and Al<sub>x</sub>Ga<sub>1-x</sub>As.** M. Ilegems, *J. Appl. Phys.*, *48* (March 1977), pp. 1278-1287.

**Composition and Stress State of Thin Films Deposited by Ion Beam Sputtering.** R. N. Castellano, M. R. Notis, G. W. Simmons, *Vacuum*, *27*, No. 3 (June 1977), pp. 109-117.

**Computer Model of Metallic Spin Glasses.** L. R. Walker, R. E. Walstedt, *Phys. Rev. Lett.*, *38*, No. 9 (February 28, 1977), pp. 514-518.

**Concerning the Semimetallic Characters of TiS<sub>2</sub> and TiSe<sub>2</sub>.** J. A. Wilson, *Solid State Commun.*, *22* (1977), pp. 551-553.

**The Copolymerization of Styrene with Sulfur Dioxide. Determination of the Monomer Sequence Distribution by Carbon-13 NMR.** R. E. Cais, H. J. O'Donnell, F. A. Bovey, *Macromolecules*, *10*, No. 2 (March-April 1977), pp. 254-260.

**Electronic Structure of Tetragonal Tungsten Bronzes and Electrochromic Oxides.** G. K. Wertheim, M. Campagna, J. N. Chazalviel, D. N. E. Buchanan, H. R. Shanks, *Appl. Phys.*, *13* (1977), pp. 225-230.

**Glass Transition Temperatures of Regularly Alternating Acrylonitrile-Vinyl Acetate Copolymers.** A. E. Tonelli, *Macromolecules*, *10* (1977), pp. 716-717.

**Impurity Effects in the Nucleation of Alpha(bcc)-Tantalum or Beta-Tantalum Films.** N. Schwartz, E. D. Feit, *J. Electrochem. Soc.*, *124* (January 1977), pp. 123-131.

**Intramolecular Interactions as the Source of Sequence Distribution-Glass Transition Effects and Dilute Solution Properties of Styrene-Methyl Methacrylate Copolymers.** A. E. Tonelli, *Macromolecules*, *10* (1977), pp. 633-635.

**Large Elastic Softening in the One-Dimensional Magnetic Material CsNiF<sub>3</sub>.** M. Barmatz, L. R. Testardi, M. Eibschutz, H. J. Guggenheim, *Phys. Rev. B*, *15*, No. 9 (May 1, 1977), pp. 4370-4375.

**Localized Gap States in Amorphous Semiconducting Compounds.** J. J. Hauser, F. J. DiSalvo, R. S. Hutton, *Phil. Mag.*, *35* (June 1977), pp. 1557-1575.

**Many-Body Effects in X-ray Photoemission from the Noble Metals.** Y. Yafet, G. K. Wertheim, *J. Phys. F Metal Phys.*, *7* (1977), pp. 357-365.

**Phonons in a Cold Glass.** B. Golding, J. E. Braebner, *Bull. Amer. Phys. Soc.*, *22* (March 1977), p. 310.



**Phonon Self-Induced Transparency in Silica Glass.** B. Golding, J. E. Graebner, *Bull. Amer. Phys. Soc.*, **22** (March 1977), p. 310.

**Raman Spectrum of Vitreous BeF<sub>2</sub>.** G. E. Walrafen, R. H. Stolen, *Solid State Commun.*, **21** (1977), pp. 417-419.

**Relative Rates of Nickel Diffusion and Copper Diffusion Through Gold.** H. G. Tompkins, M. R. Pinnel, *J. Appl. Phys.*, **48** (July 1977), pp. 3144-3146.

**Spectroscopic Study of RF Oxygen Plasma Stripping of Negative Photoresists. II. Visible Spectrum.** J. E. Griffiths, E. O. Degenkolb, *Appl. Spectrosc.*, **31** (March 1977), pp. 134-137.

**Temperature Dependence of Positron Lifetimes in Linear Polyethylene.** S. Y. Chuang, S. J. Tao, T. T. Wang, *Macromolecules*, **10** (May-June 1977), pp. 713-715.

## PHYSICS

**Excited State Spectroscopy of Molecules Using Opto-Acoustic Detection.** C. K. N. Patel, R. J. Kerl, E. G. Burkhardt, *Phys. Rev. Lett.*, **38** (May 23, 1977), pp. 1204-1207.

**AC Stark Splitting of Two-Photon Spectra.** J. E. Bjorkholm, P. F. Liao, *Opt. Commun.*, **21** (April 1977), pp. 132-136.

**The Anomalous Behavior of TiSe<sub>2</sub> and the Excitonic Insulator Mechanism.** J. A. Wilson, S. Mahajan, *Commun. Phys.*, **2** (1977), pp. 23-29.

**Avalanche Punch-Through Erase (APTE) Mode in Dual-Dielectric Charge-Storage (DDC) Cells.** D. Kahng, J. R. Brews, W. J. Sundburg, *IEEE Trans. Electron Dev.*, **ED-24** (May 1977), pp. 531-535.

**A Bistable Fabry-Perot Resonator.** P. W. Smith, E. H. Turner, *Appl. Phys. Lett.*, **30**, No. 6 (March 15, 1977), pp. 280-281.

**Charge Density Waves in the 2H-TaSe<sub>2</sub> Family: Action on the Fermi Surface.** J. A. Wilson, *Phys. Rev.*, **B15** (June 15, 1977), pp. 5748-5757.

**Charge Transfer Complexes and Hyperpolarizabilities.** B. F. Levine, C. G. Bethea, *J. Chem. Phys.*, **66** (February 1, 1977), pp. 1070-1074.

**Cluster Model for the Electronic Structure of the Chevrel-Phase Compound PbMo<sub>6</sub>S<sub>8</sub>.** L. F. Mattheiss, C. Fong, *Phys. Rev.*, **15** (February 15, 1977), pp. 1760-1768.

**Direct Calculation of the Group Velocity.** R. N. Thurston, *IEEE Trans. Son. Ultrason.*, **SU-24** (1977), pp. 109-111.

**Dispersion of Electronic Surface Resonances and Crystal Surface Structure.** E. G. McRae, J. M. Landwehr, C. W. Caldwell, *Phys. Rev. Lett.*, **38** (June 13, 1977), pp. 1422-1425.

**Electron Energy Loss Spectroscopy of the Si(111)-Simple-Metal Interface.** J. E. Rowe, G. Margaritondo, S. B. Christman, *Phys. Rev. B*, **15** (February 15, 1977), pp. 2195-2201.

**Excited States of Excitons Bound to Nitrogen Pairs in GaP.** E. Cohen, M. D. Sturge, *Phys. Rev. B*, **15**, No. 2 (January 15, 1977), pp. 1039-1051.

**Laser Doppler Velocimetry of Electron-Hole Drops in Germanium.** J. Doehler, J. C. V. Mattos, J. M. Worlock, *Phys. Rev. Lett.* **38** (March 28, 1977), pp. 726-729.

**Laser-Doppler Velocimetry of Electron-Hole Drops in Germanium.** J. Doehler, J. C. V. Mattos, J. M. Worlock, *Proc. Amer. Phys. Soc.*, **22** (March 1977), p. 349.

**Lateral Current Confinement by Reverse Biased Junctions in GaAs-Al<sub>x</sub>Ga<sub>1-x</sub>As DH Lasers.** W. T. Tsang, R. A. Logan, *Appl. Phys. Lett.*, **30**, (May 15, 1977), pp. 538-540.

**Many-Body Effects in X-ray Photoemission from the Noble Metals.** Y. Yafet, A. K. Wertheim, *J. Phys. F. Metal Phys.*, 7, No. 2 (1977), pp. 357-366.

**Method of Forming Novel Curved-Line Gratings and Their Use as Reflectors and Resonators in Integrated Optics.** P. K. Tien, *Opt. Lett.*, 1, No. 2 (August 1977), pp. 64-66.

**Nonlinear Optical Response of Metal-Barrier-Metal Junctions.** B. Fan, S. M. Faris, T. K. Gustafson, T. J. Bridges, *Appl. Phys. Lett.*, 30, No. 4 (February 15, 1977), pp. 177-179.

**Optical Multiplexer for Multimode Fiber Transmission Systems.** W. J. Tomlinson, G. D. Aumiller, *Appl. Phys. Lett.*, 31 (August 1, 1977), pp. 169-171.

**Orientation Dependence of Free-Carrier Impact Ionization in Semiconductors: GaAs.** T. P. Pearsall, R. E. Nahory, J. R. Chelikowsky, *Phys. Rev. Lett.*, 39 (August 1, 1977), pp. 295-298.

**Relationship Between Plate and Surface Modes of a Tube.** R. L. Rosenberg, R. N. Thurston, *J. Acoust. Soc. Amer.*, 61 (June 1977), pp. 1499-1502.