MAINTENANCE PROGRAM ORGANIZATION

DESCRIPTION

NO. 1 ELECTRONIC SWITCHING SYSTEM

CONTENTS PAGE

1. GENERAL

1.01 This section describes the maintenance programs of the 2-wire No. 1 ESS and the

4-wire No. 1 ESS. The function of the maintenance programs is to keep the system operating in the presence of trouble, to localize trouble for maintenance personnel, and to guard against the accumulation of undetected troubles or of uncorrected errors that might affect the traffic handling capability of the system.

1.02 A telephone switching system must provide continuous and accurate service without unreasonable delays. This quality of service must be provided 24 hours a day throughout the design life of the system. Thus a successful system must be both dependable and maintainable. Dependability is defined as a measure of service continuity and accuracy; maintainability is defined as a measure of the ease with which the component failures can be detected, located, and repaired.

1.03 The use of high-speed data processing circuits in the No. 1 ESS has brought about an increase (as compared to earlier switching systems) in the time sharing of circuits and consequently in the centralization of the control functions. This centralization has many advantages, but it can make the system more vulnerable to component failures. For example, if no redundancy is provided, it would be possible for a single component failure in the central control to cause a complete system failure.

1.04 The No. 1 ESS is at least as dependable and easy to maintain as the existing electromechanical switching systems. The objectives for dependability are that the system down time should not exceed three minutes per year and that the calls handled incorrectly should not exceed 0.002 percent. The objectives for maintenance require that the troubles can be located and repaired easily and rapidly and that the system can be left unattended for extended periods of time.

1.05 The No. 1 ESS maintenance plan is implemented partly by circuits and partly by programs. Some maintenance functions can be performed by either the circuits or the programs. Trouble detection is a function that requires continuous attention and is, therefore, usually better suited to circuits than to programs so that system real time is not used for this function. There are some instances where programs can accomplish the detection function more effectively than circuits. For example, faults in the circuits that are used with duplicate units might be likely to go undetected until the circuit is needed. To avoid undetected faults, programs called routine exercises test for trouble in circuits that are not frequently used. These routine exercises may be initiated automatically by the system or they may be requested via the teletypewriter by maintenance personnel.

1.06 There are five classes of No. 1 ESS programs. In order of urgency, these classes are as follows.

(1) Programs that recover the system data processing ability. These programs operate at levels A through E interrupts.

(2) Programs that recover proper operation of the peripheral system. These programs operate at a level F interrupt.

(3) Programs that localize faults within a system unit that contains troubles but may or may not pass diagnosis. These programs operate at levels B and K interrupts.

(4) Programs that handle inputs and outputs. These programs operate at levels H and J interrupts.

(5) Programs that process calls within the system. These call processing programs operate at the base level (level L).

1.07 There are two basic types of maintenance programs: interrupt programs and noninterrupt programs.

(a) Interrupt maintenance programs are initiated by trouble detection circuits. These circuits alert the central control which interrupts the call processing programs. The interrupt initiates a fault recognition program which is nondeferrable and which must be completed before call processing can be resumed. The fault recognition program determines which system unit has failed, removes the faulty unit from service, switches to a duplicate unit, records the failure in the call store area reserved for subsystem status, and returns control to the call processing programs. Fault recognition programs are usually brief enough so that they do not interfere with normal call processing.

(b) Noninterrupt maintenance programs include diagnostic programs, routine exercise programs, audit programs, and deferrable fault recognition programs. These programs are performed under the control of the maintenance control program and the maintenance control register. The maintenance control program periodically checks the maintenance control register in the subsystem status area of the call store to determine whether a noninterrupt maintenance program is required. When the maintenance control program starts a maintenance program, that maintenance program is called the client program or the client of the maintenance control register.

1.09 Definitions of terms and abbreviations used in this section are as follows.

(a) ***Fault***—A recurring trouble (a malfunction that can be consistently reproduced).

(b) ***Error***—A nonrecurring trouble.

(c) ***Interrupt***—A seizure of control by the central control from a program being executed for the purpose of giving control to another program that is more urgent.

(d) ***Noninterrupt***—A low-priority request that central control acts on when time is available.

(e) ***Recovery***—Regaining the ability to process calls and to perform other normal functions in spite of a malfunction. An example is the substitution of standby equipment for active equipment that is not functioning properly.

(f) ***Routine Exercise***—Periodic programmed routines whereby the system tests its own ability to perform properly and detects faults before system operation is affected.

(g) ***Audit***—A check on the validity of the state of the system. In particular, a call store audit consists of a check on the correctness of information in some storage locations and of the correction of invalid information. The contents of the storage locations checked are compared with information in other call store locations or with information in the program store.

(h) ***Deferrable***—An operation that can be performed when system time is available. Deferrable operations do not interrupt operations in process.

(i) ***Nondeferrable***—An urgent operation that must be performed immediately to restore system operation in the event of a major failure. Operations in process are interrupted until a nondeferrable operation is completed.

(j) ***MAC***—The maintenance control program that controls and administers various deferrable maintenance programs specified in the maintenance control register (MACR). The specified maintenance program is called the client program.

(k) ***MACR***—The maintenance control register that stores requests for deferrable maintenance program operations. The MAC program interrogates the MACR to determine which maintenance program to initiate. The maintenance program then started under control of the MAC program is called the client of the MACR.

2. ACTIONS BY MAINTENANCE PERSONNEL

2.01 When a diagnostic program discovers a failure, the system leaves the equipment frame containing the fault out of service. In order to alert maintenance personnel to an existing fault, the system

(a) Sounds an audible alarm

(b) Lights the pilot lamp for the aisle containing the faulty frame

(c) Lights the out-of-service lamp on the control panel of the equipment frame containing the fault

(d) Lights the appropriate MAJOR or MINOR alarm lamp on the alarm, display, and control panel of the master control center

(e) Lights a lamp on the alarm, display, and control panel of the master control center to indicate which type of equipment frame contains the fault

(f) Prints the frame identification and the diagnostic results at the maintenance teletypewriter.

2.02 The relationship of maintenance programs, trouble detection circuits, and maintenance personnel is summarized in Fig. 1.

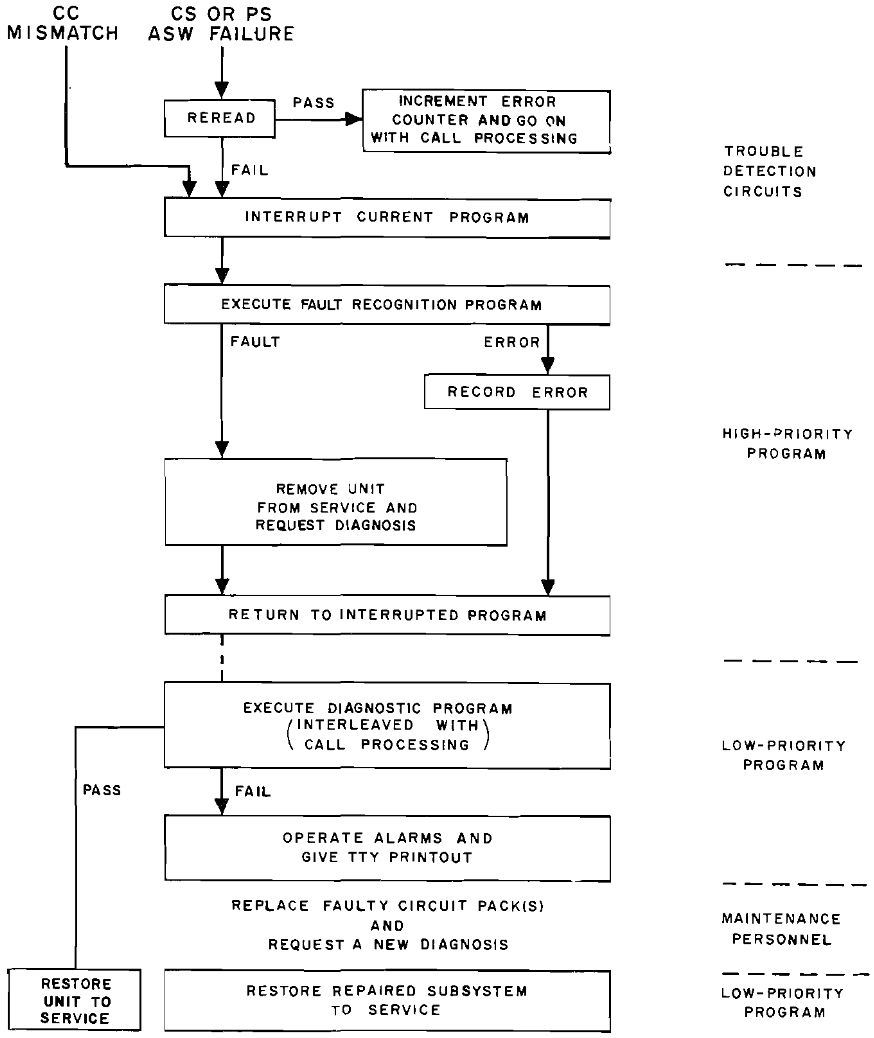

Fig. 1—Maintenance Program Reaction to Trouble

3. RECOVERY FROM TROUBLE

3.01 Circuit troubles are detected by fault detection circuits or, when a network controller or a trunk circuit produces an abnormal response, by call processing programs. When trouble is detected, program control is usually transferred immediately to programs that recover the call processing capability of the system. These programs are called fault recognition (FOR) programs.

4. FAULT RECOGNITION PROGRAMS

4.01 The purpose of the FOR programs is to re-establish the call processing ability of the system. This is accomplished by removing the faulty unit from service and by requesting a diagnosis of the unit. The FOR programs are usually nondeferrable and have the highest priority of the system programs. Also these FOR programs can be used as deferrable programs.

4.02 A typical FOR program carries out the following tasks:

(1) Determines whether the trouble detected was an error (a transient malfunction that can no longer be reproduced) or a fault (a reproducible malfunction that can be diagnosed)

(2) Identifies the faulty system unit and takes it out of service

(3) Records a request for a diagnosis of the faulty unit

(4) Records all pertinent error or fault information

(5) Returns the system to normal call processing.

4.03 Interrupt sources within the system are divided into ten levels that are designated A through K (excluding I) from highest to lowest priority (Table A). An interrupt source of a given level can override any interrupt source of a lower level. The priority level of the main program is called the base level or level L.

TABLE A

INTERRUPT LEVELS

LEVEL	MEANING
A	Master control center (manual control)
B	Emergency action
C	CC mismatch
D	CS reread failure
E	PS reread failure
F	Improper operation of the peripheral system
G	Error evaluation and special programs
H and J	5-millisecond interrupt
K	Signal processor (2-wire only)

4.04 The interrupt level provides a rough isolation of trouble by indicating what type of unit contains a fault. A separate FOR program is initiated by each of the levels B through E interrupts (Fig. 2). The level F interrupt has many sources; a filter program selects the proper FOR program.

4.05 When the FOR program has established a fault-free configuration, the FOR program transfers control to a restart program which determines whether the condition of the system justifies a return to call processing either at the point where it was interrupted or at a reference point that does not depend on past history. The return to a reference point is necessary in such cases as when a central control (CC) mismatch was caused by an error and the system has no way of determining which CC was wrong.

4.06 The relationship between programs shown in Fig. 2 is incomplete. For example, a CC mismatch may be due to a malfunction in a program store or a call store, in which case a transfer is made to the appropriate FOR program.

EMERGENCY ACTION

4.07 The emergency action program (PD-1A010) is activated by certain checks on programs (for example, by failure to complete E-to-E base level cycles in a given amount of time) or by an excessive rate of maintenance interrupts. The emergency action program is divided into phases which are executed until an operational memory is obtained (Part 12). With each phase more and more call store memory is initialized until, with phase VI (initiated manually only), even the recent change memory is initialized. The emergency action is a backup for the normal recovery facilities and should occur very infrequently in normal system operation.

CENTRAL CONTROL FAULT RECOGNITION

4.08 During system operation two CCs are running in step: one designated active and the other standby (synchronously executing the same instruction). Cross-connection circuits continuously match the data in these two CCs running in step. A malfunction in either CC causes a mismatch which initiates a level C interrupt and a transfer of program control to the start of the central control-fault recognition (CC-FOR) program. The primary objective of this program is to determine whether the mismatch was caused by a permanent fault in one of the CCs or by a transient error. If a permanent fault is detected, the CC is removed from service and a diagnostic is requested.

4.09 The CC-FOR program (PD-1A015) functions are as follows:

(a) To qualify the CC trouble indication (mismatch) as transient malfunction (error) or reproducible malfunction (fault)

(b) To isolate the fault (if any) to the active or to the standby CC

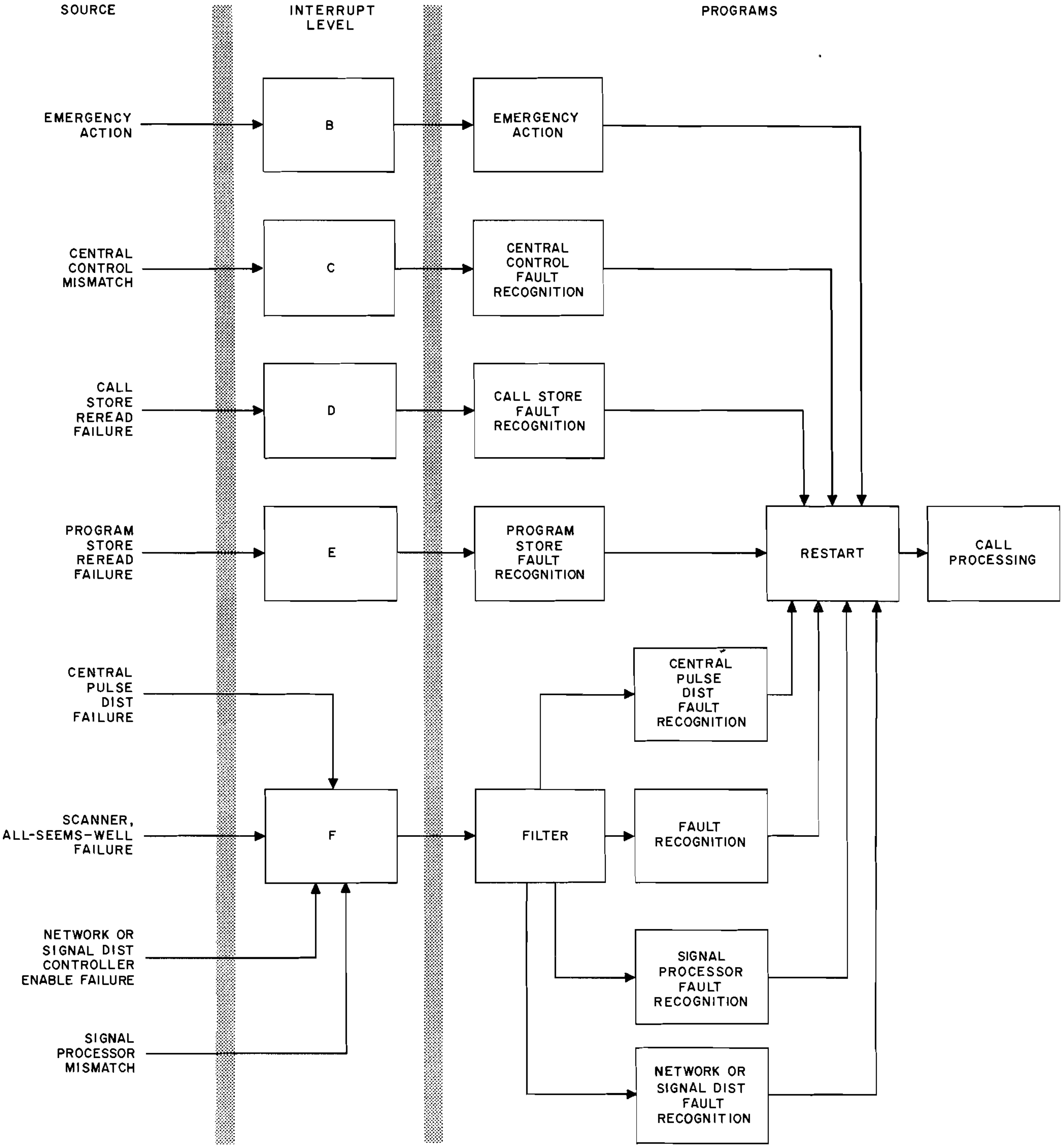

Fig. 2—Initiation of Fault Recognition Programs

(c) To remove the faulty CC from service and to substitute the standby for the active CC if necessary

(d) To request a diagnostic of the faulty CC when necessary

(e) To record pertinent CC error and fault information

(f) To initiate actions required to return the system to normal operation at the conclusion of the program.

4.10 To distinguish a fault from an error, test exercises are executed by the CCs. Failure of any of these exercises indicates a fault. If all the exercises pass the tests, the malfunction is considered an error (not reproducible).

4.11 Isolation of the faulty unit is accomplished by conditional transfer orders in the test exercises and in the CC match circuits. A typical test is the operation on data within the CC and is followed by a conditional transfer order based on the result of this operation. If the proper decision is made by the active CC, it passes the test. If the active CC passes the test and the standby CC has not made the proper decision, a mismatch is detected and the standby CC fails the test. Consequently, the two objectives of distinguishing a fault from an error and of isolating the faulty unit are met simultaneously.

4.12 The CC-FOR first-look program tests the CC hardware that most likely caused the mismatch. When the mismatch is detected, information is saved within the CC to define the hardware being matched at that time and the class of instruction being executed. This information is then used by the CC-FOR program to decide which CC tests to run. For example, if the mismatch is detected at the index adder output register, part of the CC-FOR program tests the index adder circuitry, the registers, and the buses which have access to the index adder. If these tests are passed, it is assumed that the mismatch was generated by an error rather than a fault.

4.13 The complete CC-FOR program serves as a backup to the first-look program and is used either when it appears there are CC faults not being detected by the first-look program or when abnormal conditions exist. This CC-FOR program is time consuming and is used only when there are strong indications that a fault exists in one CC.

CALL STORE FAULT RECOGNITION

4.14 The call store-fault recognition (CS-FOR) programs (PD-1A018) perform the fault recognition task for the CSs assciated with CCs (CC-CS), the CSs associated with signal processors (SP-CS), the related bus systems, and the circuits in the CCs and SPs associated with either or both bus systems used to communicate with the CSs.

4.15 The CS-FOR programs restore, or recover, call processing capability by switching various CS units and buses until a workable CS configuration is established. The CS-FOR programs also determine if further action by the diagnostic programs is required to analyze trouble in a subsystem.

4.16 The programs are divided into recovery and deferred phases. Programs executed during an interrupt are called recovery phases. The system does not handle telephone calls while the recovery phases are in process. Therefore an operational system configuration must be established as soon as possible.

4.17 In addition to recovering an operational system configuration, the CS-FOR programs isolate the faulty units. The isolation programs are called the deferred phases. During these phases, the CS-FOR program determines which subsystems contain faults. Faulty subsystems are switched out of service, and diagnosis of faulty units is requested. The deferred portion of the CS-FOR program is interleaved with call processing programs; therefore, telephone calls are handled while faults are being isolated.

4.18 Each CS contains circuits for checking its operation. The CC also contains circuits for checking the operation of the complex of CSs and CS buses. Conditions detected by these maintenance circuits result in bringing the CS-FOR programs into operation whenever necessary.

4.19 A CS generates an all-seems-well CS (ASWC) signal if it passes certain internal checks during an operation. The ASWC signal is a summary of internal checks made by the CS circuits. Generation of the ASWC signal is inhibited to inform CC that the CS has not functioned properly. When CC does not receiver the ASWC signal, it requests a CS reread. If the CS reread also fails, a level D interrupt occurs. This interrupt causes a wired transfer to the CS-FOR program. Further processing of calls is delayed until a workable CS configuration is established.

4.20 The CC also calculates a parity bit over the address and the data word that are written into the CS memory. The parity bit is stored in the memory as bit 23 of the 24-bit CS word. When CC reads the word out of the memory, it checks parity. If the parity check fails, a CS reread occurs.

4.21 Failures that are not detected by the ASWC check and the parity check are detected when the duplicate copies of CS information are matched by the two CCs. Failure of one copy causes a mismatch that leads to a level C interrupt. The CC-FOR programs then transfer control to the CS-FOR programs.

4.22 Following a level D interrupt, the interrupt control program is entered. This program consists of the first-look, the access test, and the bootstrap subroutines.

4.23 The first-look subroutine is used to attempt a quick recovery of the system. This subroutine uses the address of the failure (stored in the CC) and the states of the system bus control flip-flops to determine what CC-CS bus configuration was in use at the time of the failure. From this, the subroutine determines which CS failed and removes that CS from service. It then adjusts the bus control flip-flops of the duplicate CS so that it can supply the duplicate information.

4.24 The access test subroutine checks that an operational system exists by addressing and by getting return readouts from each information block in the CS system. If the access test is successful, the CC-CS complex is considered operational and call processing is restarted. If the access test fails, the first-look recovery was not successful and the interrupt control program calls in the bootstrap subroutine.

4.25 The bootstrap subroutine attempts to recover an operational CC-CS complex on one bus system. The standby bus is checked later during the deferred phase of CS-FOR program. Two levels of operation are provided to obtain the CC-CS configuration. First, only CSs which were in service at the time of the interrupt are tested. If an operational configuration cannot be obtained from these CSs, then all CSs are tested. The configurations for the sequences of switching during bootstrap recovery are shown in PF-1A018. If the bootstrap subroutine cannot recover the system, no automatic recourse exists.

PROGRAM STORE FAULT RECOGNITION

4.26 The program store-fault recognition (PS-FOR) programs (PD-1A120) are concerned with operating irregularities which require testing of PSs, buses, and/or associated equipment. The PS-FOR program is entered after a PS reread failure interrupt.

4.27 The PS-FOR program performs the following operations.

(a) Determines whether the trouble is a fault or an error. (A trouble is classified as a fault if the tests conducted during the fault recognition sequences reproduce the trouble indications or generate new trouble indications; otherwise, the trouble is an error.)

(b) Identifies any faulty subsystem, removes the unit from service, and requests appropriate diagnostic programs.

(c) Reconfigures interconnections among fault-free equipment where necessary to create a workable overall system configuration.

(d) Updates status records for all subsystems.

(e) Records pertinent error data and analyzes this data to determine whether some particular subsystem is marginal. If such a marginal unit is found, the program treats it as faulty.

(f) Returns the system to call processing within a few milliseconds after the interrupt occurs.

CENTRAL PULSE DISTRIBUTOR FAULT RECOGNITION

4.28 The central pulse distributor fault recognition (CPFR) program (PD-1A022) is entered whenever a peripheral (level F) interrupt occurs. Such an interrupt occurs if the maintenance check of a peripheral instruction fails and the level F interrupt pest control flip-flop in the buffer bus register of the CC has not been set to suppress the interrupt.

4.29 If the interrupt occurred on a peripheral address, a retry of the instruction is normally performed to determine whether the trouble is transient or whether a fault exists. If the retry passes, peripheral error counters are incremented and a teletypewriter message with details of the error is printed out. If the error occurred during supervisory scanning, the scanner enable route is changed to select a new bus and a controller. This is done because the retry of a supervisory scan instruction may address a different row from the one that failed. Changing the scanner enable route is likely to switch out the trouble.

4.30 If the retry failed, the program attempts to isolate the source of trouble by trying again with a different central pulse distributor (CPD) and CPD bus configuration. If a good configuration is found, the system routing information is updated and diagnosis of the faulty subsystem is requested. Call processing is then resumed.

4.31 If a working configuration is not found, the program determines whether a scan order failed. If a scan order did fail, the CPFR program transfers control to the scanner FOR program for further investigation. Otherwise, a switch of CCs is requested since a translator or a cable pulser in CC could be at fault.

4.32 The error retry procedure does not occur if the interrupt occurs during a network instruction or a signal distributor instruction. In these instances, control is transferred directly to the network FOR program. In the event of an enable verify mismatch wherein two frames return verify signals, the noisy frame and the controller are determined from the verify answer signal. A message identifying the noisy frame is printed out so that power can be removed from that frame. The CPD that has been active at the time of the mismatch is switched out to prevent further mismatches in case the CPD is at fault.

SCANNER FAULT RECOGNITION

4.33 When the CC addresses a scanner (PD-1A026), certain responses are expected from the scanner and the CPD to indicate that the instruction has been successfully executed. These responses include the enable verify signal which indicates that the proper peripheral frame had been enabled and the all-seems-well scanner (ASWS) signal which indicates that the scanner has sent back answers from one and only one row of scan points. If these responses are absent, a level F interrupt occurs and the CPFR program is entered. To filter out transient errors, the CPFR program retries the instruction that failed. If the trouble indication persists, the CPFR program checks the type of instruction and checks the failure indications to determine whether scanner trouble exists. If the scanner is at fault, control is transferred to the scanner fault recognition program (SCFR).

4.34 The SCFR program attempts to find a working configuration of a peripheral unit bus, a CPD, a scanner controller, and an answer bus by retrying the failing instruction with various combinations of these subsystems. If a good configuration is found, the SCFR program requests diagnosis of the scanner and updates the appropriate route information so that the faulty configuration is switched out. Also the appropriate trouble lamp corresponding to the type of scanner is lighted at the master control center.

4.35 If a good route is not found after all configurations have been tried, a validity check is made on the scanner instruction to determine if it contains a legitimate scanner enable address and to determine if a scanner translator is specified for decoding the scanner row address. If the validity check fails, the program transfers to a routine to regenerate the route tables from backup information in the PS. If the validity check passes, a bad CC, an open scanner row, or a double trouble is indicated. To handle these contingencies, the appropriate primary trouble lamp is lighted at the master control center; the CCs are switched; the scanner diagnosis is requested; and the ASWS request bit is set to 0. With the ASWS request bit zeroed, the system can run without level F interrupts in supervisory scanning. The ASWS request bit is restored to 1 if the scanner passes the diagnosis.

NETWORK AND SIGNAL DISTRIBUTOR FAULT RECOGNITION

4.36 In order of priority, the network and signal distributor fault recognition (NMFOR) program (PD-1A028) has the following purposes:

(1) To recognize and to record malfunctions of network controllers or signal distributors (SDs)

(2) To utilize standby duplicate units in order to find a workable configuration for completing the requested peripheral action

(3) To alter the master enable tables, thus directing the system to use the best possible configurations of network controllers, SDs, peripheral unit buses, and CPDs

(4) To take requested failure action on failing orders

(5) To request diagnostic action on suspect peripheral units

(6) To inform maintenance personnel of failures to transfer supervision on network actions

(7) To administer a linked list of requested switching actions involving network and SD units

(8) To provide certain services to client programs (chiefly the network and the SD diagnostic), such as sending diagnostic orders, timing delays, and recording data.

4.37 The network consists of the fabric and the controllers. The fabric through which talking and signaling paths are set up is the actual body of switches and links. The controllers are logical circuits which serve as interface between the CC or the signal processor and the fabric. When some change in the fabric is requested by the CC or the signal processor, an appropriate order is sent to the proper network controller via the peripheral unit bus. The controller decodes the order and performs the requested function by actually opening or closing the desired ferreed switches or the relay contacts in the fabric. For the purpose of fault recognition, some internal checks are provided on controller operations. Assigned to each controller for the purpose of fault recognition are master scanner points F, S, and T that monitor the states of the F, S, and T flip-flops. The combination of the states of these flip-flops specifies the condition of the controller. Malfunctions in the controller circuitry are indicated by its failure to cycle properly and to finally reset itself, thus bringing the F, S, and T scan points back to the reset state. The possibliity of fault or error detection is provided for the controller and the ferreed coils, but is not provided for the ferreed switch.

4.38 The SD includes the controller, the matrix, and the magnetic latching relays. The controller receives, decodes, and administers orders received from the CC or the signal processor. The matrix consists of wire-spring relays through which a pulsing path may be set up to operate or to release a desired magnetic latching relay. The relays make up the signal distribution paths.

4.39 Malfunctions of the controller, the matrix, or the selected magnetic latching relay are indicated by the inability of the controller to cycle completely and to return to the reset state as checked by the F, S, and T scan points.

TELETYPEWRITER MAINTENANCE

4.40 The teletypewriter (TTY) maintenance program (PD-1A029) is a diagnostic program (TTDN) that isolates a fault within a given TTY circuit to a small number of replaceable circuit packages within the transmit-receive unit or to the TTY loop. The TTY itself is not diagnosed but the absence of current which takes the form of an open loop is noted. The program performs a series of exercises on the TTY circuit, collects the results of the exercises, compares them with expected results, and reports the results of the comparison to maintenance personnel via the maintenance TTY.

AUTOMATIC MESSAGE ACCOUNTING MAINTENANCE

4.41 The automatic message accounting (AMA) maintenance programs (PD-1A030) consist of the AMA-fault recognition (AMA-FOR) program and the AMA diagnostic program. The AMA-FOR program is entered via the AMA output programs whenever a condition is detected which makes the recording of call data impossible or of questionable accuracy. The primary purpose of the AMA-FOR program is to determine whether or not the trouble has permanently affected unit operation. It is possible that the detected condition was merely a transient error and that the unit can continue to be used for recording data output. If the trouble is of a more permanent nature, it is necessary to place the standby AMA into service to restore normal operation.

4.42 The AMA-FOR program is started when any of the following conditions occur during the data output program.

(a) A not-all-zero condition is encountered upon interrogation of the AMA master scanner points prior to transferring a block of data.

(b) A positive signal exists on an alarm output lead (read every 5 milliseconds during data transfer) indicating that at least one of the master scanner points has gone high.

(c) Two consecutive tape reading errors occur during a data transfer check sequence.

4.43 The AMA-FOR program consists of the following major parts:

(a) Master scanner point check sequence

(b) Tape bypass operation in maintenance mode No. 1

(c) Normal mode recording and check sequence.

5. MAINTENANCE CONTROL PROGRAM

5.01 The purpose of the MAC program is to control and to administer various deferrable maintenance programs specified in the MACR. These maintenance programs include diagnostic, audit, routine exercise, and deferrable fault recognition programs. The specified maintenance program is called the client program. The MAC program administers operation of the client program and interleaves segments of the client program with the processing of calls (Fig. 3). Control is returned to the call processing programs after each segment of the client program so that telephone service is not interrupted. Some of the tasks carried out by the MAC program are as follows.

(a) Answers diagnostic requests usually from an FOR program recorded in the subsystem status table. These diagnositc requests are handled on a priority basis. For example, units such as CCs and PSs have the highest priority.

(b) Answers routine exercise requests in the recorded routine request table. Four priority levels are available: level A, the highest, is for requests made via the maintenance TTY; level B is for requests made by programs; and levels C and D are for requests made by automatic scheduled routine exercises.

(c) Initiates certain trouble alarm scans periodically which perform trouble detection functions. For example, the single-error counter in the CC is scanned periodically. An excessive error count initiates an error-evaluation program which leads to a diagnostic program.

(d) Ensures that no client program holds the MACR for more than ***10 minutes.***

(e) Handles the communication of all client programs with the maintenance TTY channels.

(f) Performs the following tasks that are common to many of the maintenance programs:

- Timing
- Maintenance bookkeeping, such as updating system status and recording error information
- Control of common maintenance facilities, such as the audible and the visual alarms and the monitor bus.

(g) Administers repeated tests.

5.02 The MAC program (PD-1A005) is entered from the main program as the lowest priority job (level E). The time between any two consecutive MAC programs varies with traffic conditions. The MAC program starts the client program in the MACR. The client program carries out a segment of work which has a maximum duration of 10 milliseconds excluding interrupts. Upon completion of the work segment, the client program relinquishes control to the main program. Fig. 3 shows the interleaving of deferrable jobs (base level) for call processing and for maintenance with 5-millisecond interrupt input-output jobs (level J).

5.03 The MAC program determines whether the MACR is busy. The register is busy when it is assigned to any client program. If the register is idle, the MAC program searches for diagnostic requests and then for routine exercise requests. If a request is found, the appropriate client program is initiated and the MACR is assigned to this program. If no requests are found, some very low priority jobs (such as diagnostics) are performed.

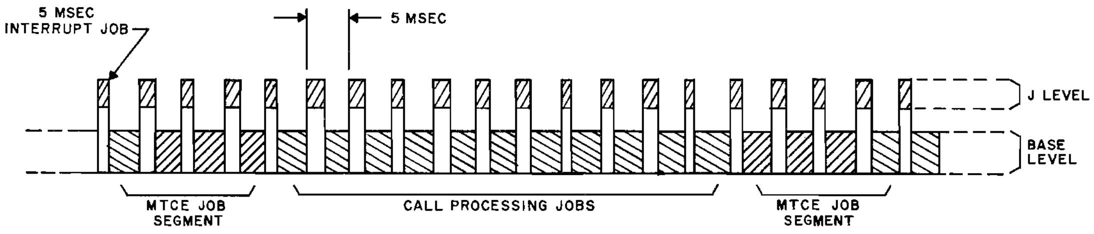

Fig. 3—Interleaving of Maintenance and Call Processing Jobs

6. MAINTENANCE CONTROL REGISTER

6.01 The MACR is a memory area in the CS used primarily by the deferrable programs. The MAC program administers, coordinates, and schedules this register. The maintenance program specified by the MACR is called the client program.

6.02 The MACR consists of a group of about 500 CS words which are used for administering the maintenance programs and for recording the results obtained. The MACR is made up of two parts. The first part is the work area which is a private memory area for a client program while it is in control of the register. The second part is the long-term record area which provides the client program with reference data pertaining to the subsystems.

6.03 The work area is used by a client program to record the results of the various tests that the program performs. Also the work area contains administrative information, such as the identity of the unit under test and of the type of program being carried out.

6.04 The long-term record area consists of the following tables.

(a) ***Routine Request Table:*** This table is a waiting list for routine exercise programs, TTY messages (inputs), etc.

(b) ***General Buffer Table:*** This table is used by maintenance programs for passing information on to other maintenance programs which will be used at a later time. For example, the table is used when a FOR program finds a fault condition and requests subsequent diagnostic action. While determining that the malfunction is a fault and is not an error, the FOR program uncovers some information that would be helpful to the diagnostic program. This information is entered into the general buffer area.

(c) ***Error Record Table:*** This table is used by an FOR program for recording error data, such as the pattern of units in operation at the time an error is detected. Also this table contains an error counter for certain subsystems. An error counter is incremented when the FOR program associated with the unit under test determines that an error has occurred. Error counters are examined periodically, and appropriate action is taken if a high-error count is found.

(d) ***Subsystem Status Table:*** This table is a history file on the various subsystems. The table records whether the status of each subsystem for the major processing units (CC, CS, PS, and CPD) is active, standby and ready to be active, standby and out of service, or standby and requesting diagnostic action.

(e) ***Miscellaneous Record Table:*** This table contains specialized information for the various maintenance programs. Memory spaces in this table are permanently assigned to a client program for use in recording private administrative records.

7. DIAGNOSTIC PROGRAMS

7.01 Diagnostic programs localize a fault to a small number of plug-in circuit packs within a system unit that has failed diagnosis and that has been taken out of service. A diagnostic program carries out a fixed sequence of tests that are performed by monitoring either the normal outputs of a unit or the test points located within the unit. The test points are monitored by a scanner during a control read operation. The scanner examines information either directly or via a diagnostic bus.

7.02 A diagnostic program records in the MACR whether each test passes or fails. From these test results, the number generation program determines the trouble number which is printed out by the maintenance TTY. Maintenance personnel can request a raw data printout via the maintenance TTY. This is a direct printing of the pass or the fail results of each diagnostic test.

7.03 Diagnostic programs are relatively low priority maintenance tasks. These programs are requested by a FOR program that has detected a fault or by an unusually high error rate. The diagnostic programs are generally long and require a substantial amount of CS memory for recording tests results. The MACR, under control of the MAC program, serves as a temporary memory for each diagnostic program.

TROUBLE LOCATING NUMBER PRODUCTION

7.04 At fixed points in each test program, the stored test results are sent to a number generation routine. A number of operations are performed on the test results so that new numbers are generated. At the completion of testing, a number reduction routine performs further operations on the previously generated numbers and then combines them into a 23-bit trouble locating number. If the circuits involved in the test respond as expected during the entire test procedure, the trouble locating number is the all-tests-passed number that is stored in memory and is applicable to that test. This number is stored in memory for each circuit that has a diagnostic program. After the tests are completed, a comparison is made between the generated number and the all-tests-passed number. A mismatch indicates a test failure or failures requiring a TTY message. This message includes the generated trouble locating number to be looked up in the trouble locating manual which, in turn, indicates the particular fault encountered in the test.

8. ROUTINE EXERCISE PROGRAMS

8.01 There are two types of routine exercise programs: scheduled automatic routine exercise programs and demand routine exercise programs.

SCHEDULED AUTOMATIC ROUTINE EXERCISE PROGRAMS

8.02 The automatic routine exercise programs have periodic schedules. There are three automatic program classes which are determined by the scheduling technique.

8.03 Class I programs are rigorously scheduled at a relatively high frequency. These programs are entered from the high-priority main program regardless of the office traffic. They must of necessity be fairly short, since they are performed even during the busy hour. An example of a program assigned to this class is that program which interrogates and resets the store error counters. This must be done on a strict schedule to ensure meaningful interpretation of the contents of the counters.

8.04 Class II programs are also rigorously scheduled, but at a much lower frequency. Programs which must be performed every hour, or at some specific time during the day, are assigned to this class. An example is a program to request routine exercise of matchers.

8.05 Class III programs are the lowest priority programs in the system. These exercises are performed in system spare time when no other jobs are waiting. These exercises are ordered in a circular list, so that when one is completed, the next one in the list is initiated. Most of the routine exercises are assigned to this class. An example is programs to verify and update status words in temporary memory to insure that they agree with the actual system status.

DEMAND ROUTINE EXERCISE PROGRAMS

8.06 The demand routine exercise programs are initiated upon request. The request can be initiated by the TTY or by some other program. All automatic programs can be requested as demand programs. Some examples of demand programs are

(a) A program to remove a unit from service,

(b) A program to restore a unit to service,

(c) A program to print out the status of a particular unit,

(d) A program to print out the contents of a specified call store location, etc, and

(e) Programs that audit the network memory.

8.07 All of the maintenance programs are normally executed by the active CC or by both the active and standby systems. Demand routine exercise programs are provided to allow the execution of almost all of these programs on a repeated basis by the standby system. The need for this ability may arise if a marginal trouble develops which is not detected by the diagnostic programs. To run the standby CC independently of the active CC (that is, off-line), all that is required is to interconnect the CCs, CSs, and PSs so that the CCs operate as two independent systems.

8.08 If it is desired to execute some program continuously, starting at address A and ending at address B, this could be accomplished in the following manner. An input message would request that the standby system execute the program from address A to address B. This demand exercise would modify CS and PS configurations to set up two independent systems. The active CC would stop the standby CC and control write the start address A into the standby program address register. It would then set up the breakpoint match mode to monitor address B with the interrupt and stop standby options specified. Next, it would start the standby CC and return to call processing. The standby system would then begin executing the program at address A while the active system ran its normal call programs. When the standby system reached address B, the active matchers would be alerted, stop the standby system, and interrupt the active system with a G-level interrupt. This interrupt program could restart the standby system at A and return to call processing, etc.

9. TRUNK AND SERVICE CIRCUIT TEST PROGRAMS

FUNCTION

9.01 A request for trunk and service circuit automatic testing may be originated directly or indirectly. Direct requests are from the maintenance TTY, from the trunk and line test panel, or from the main program timing. Indirect requests result in the setting of a request flag in the MACR which honors the request when higher priority maintenance work has been completed. The administration program performs the actions required to prepare a circuit for testing. Control is then passed to the diagnostic program for that particular type of circuit. For test program purposes, any circuit that has an assigned trunk network number and that is connected to the trunk link network is treated as a trunk circuit.

9.02 A diagnosis requires a network connection between the circuit to be tested and the one or more test circuits. The connected test circuits are set to their various states, and scans are made to check the responses of the circuit being tested. Upon completion of testing, the test results indicate whether a diagnosis passed or failed.

9.03 The administration program disposes of the tested circuit by returning it to an idle list, by putting it on the trunk maintenance request list, or by putting it on the trunk out-of-service list depending upon the results of the test and upon certain system conditions. A TTY message is given for all failures and, in some cases, for all tests that passed.

AUTOMATIC PROGRESSION TESTING

9.04 The automatic progression testing program is entered from the main program on a scheduled basis. Each time the automatic progression test program is entered, the trunk maintenance request list (TMRL) is examined for trunks that have remained on the list since the previous entry to this program. All such trunks are removed from the TMRL and are identified by a TTY message (TN01, TN02, TN03, or TN04). The program continues with an entry to the trunk network maintenance subroutine which selects trunks for testing when they appear on the TMRL. When the TMRL tests are completed, trunk automatic progression testing begins. Trunks are selected numerically in the order of their trunk network numbers. The selection process generally includes idle, busy, and unassigned trunks.

9.05 The trunk network diagnostic subroutine is entered with the selected trunk, and the trunk is prepared for testing if it is idle. Control is then passed to the particular diagnostic program responsible for the circuit actions needed to test the selected trunk. A trunk that passes the test is restored to the idle list, while a trunk that fails the test is entered on the TMRL. Before another trunk is selected for diagnosis, control is transferred to an MACR so that higher priority maintenance work can be done before returning to this program.

9.06 Trunks that are service busy, out of service, or unassigned are passed by when automatic progression testing is being done; however, each time an out-of-service trunk is found, its trunk network number is printed.

TESTING TRUNKS APPEARING ON LISTS

9.07 The TMRL is a fixed length list that contains information for five trunks. When a trunk is to be entered on it, the first available space is used. Upon each entry of the trunk network maintenance subroutine, the entry appearing uppermost in the TMRL is placed at the bottom and all other entries are rotated toward the top. The trunk network number occupying the bottom position is given to the trunk network diagnostic subroutine for testing. A variety of programs may request a diagnosis of a trunk by entering appropriate information on the TMRL or (2-wire ESS only) on the auxiliary trunk maintenance list (TMLA). Periodically or interleaved with automatic progression testing, one of the trunks on the TMRL or the TMLA is selected for testing when an entry is made to a trunk network maintenance subroutine. The selected trunk is prepared for testing by the trunk network diagnostic subroutine and is tested by the diagnostic program.

9.08 Test results are returned to the trunk network maintenance subroutine. A trunk that was not tested is marked maintenance busy and is left on the TMRL for subsequent retrials or for disposition as specified by automatic progression testing. Trunks that pass the test are idled, while those that fail the test are entered on the out-of-service list if possible.

9.09 Only 15 trunks can be taken out of service by automatic trunk maintenance between trunk network program entries. The number of trunks that may be taken out of service in a particular trunk group is also limited as follows:

- One trunk in groups containing one through four trunks
- Two trunks in groups containing five through eight trunks
- Three trunks in groups containing nine through twelve trunks
- Four trunks in all larger groups.

When a limit is reached, additional failing trunks are returned to service.

9.10 Test results are given by a TTY message after each test whether it passed or failed, and the trunk is removed from the TMRL.

9.11 In the 2-wire ESS when the trunks specified in the TMRL have been tested, trunks specified by the TMLA are idled in memory and in hardware. The list of trunks in the TMLA is shifted to the TMRL if vacancies occur and if the TMLA contains fewer than five trunks. If both of these conditions are not met, trunks from the TMLA are restored to service until the conditions are met. In 4-wire ESS, trunks which are placed on the TMLA are not handled by the trunk and service circuit test programs. Instead they are idled in memory and in hardware by a program operating in level L independently of the MACR. When processing is completed, automatic progression testing is resumed if it had started; otherwise, control is returned to the MACR.

TESTING TRUNK GROUPS

9.12 On reception of a request for testing a trunk group, a TTY output message is initiated to indicate that testing has started. In numerical sequence all trunk network numbers in the office are examined, and only those numbers belonging to the requested group are tested if they are idle. Test results (pass, found busy, or failure) are printed out by the TTY. When a diagnosis is blocked, a second attempt is made before the TTY is activated. Upon completion of testing, an end-of-test message is printed out by the TTY.

9.13 Time breaks are taken during trunk group testing through interaction with the MACR.

INTERACTION WITH MAINTENANCE CONTROL REGISTER

9.14 Except for the functions performed on level J, the trunk maintenance jobs are relegated to a low-priority program level. Various time breaks are incorporated in the trunk maintenance program by returning control to the MACR each time a circuit has been tested or when 64 consecutive circuits have been found busy or out-of-service. This ensures that higher priority maintenance work is given preference.

PREPARATION OF CIRCUIT FOR TESTING

9.15 The trunk network diagnostic subroutine is given the trunk network number of the selected trunk. The requesting program is notified that the test cannot be made if the trunk is busy, out-of-service, or unassigned. Trunks that appear idle in memory are blind idled. This procedure sends release orders to all SD and CPD points associated with the trunk. If an automatic diagnostic program exists for the selected circuit, that program is entered. If no diagnosis is available, the requesting program is informed of that condition. The trunk network diagnostic subroutine may be entered directly from the trunk and line test panel program when that unit requests a diagnosis for a particular trunk network number.

CIRCUIT TESTS

9.16 The test procedure for all circuits requires connection to related circuits, thus setting the circuits to their various states and checking their reaction by making scans on some or on all of the associated ferrods. Each individual test program proceeds with a fixed test pattern which, when good circuits are involved, returns a fixed pattern of test results. The test results are obtained by storing in memory all scan results and by storing an indication of the success or the failure of hardware actions (network operations, SD operations, etc). Testing is not suspended at the time a failure occurs, but only when the entire test procedure has been completed.

9.17 When testing is initiated by the trunk and line test panel, it may request a printout of the raw data as it is received by the test program. This feature requires the operation of keys at the alarm, display, and control panel; otherwise, the results produce a trouble locating number.

TEST CIRCUIT FAILURES

9.18 When a trunk diagnosis fails, the trouble may be in the test circuit. The method used to detect test circuit troubles is as follows. Each test circuit has an associated memory bit indicating whether the previous test passed or failed. This memory bit is updated after each test. When two consecutive tests using the same test circuit have failed, the test circuit is entered on the TMRL. A diagnostic routine for the test circuit is requested as a result of TMRL processing, and the test circuit is connected at random to any available circuit of the type specified by the test program. This time a failure (the third consecutive failure using the test circuit) causes the test circuit to be entered on the trunk out-of-service list; therefore, subsequent tests requiring this test circuit are blocked.

10. TRUNK ERROR ANALYSIS PROGRAM (2-WIRE ESS ONLY)

10.01 Whenever the system encounters a trouble condition during call processing or routine trunk testing, it refers any trunk or service circuit involved in the action to the trunk and service circuit maintenance programs for diagnosis. The word trunk is used to include service circuits or trunk circuits with their transmission facilities. If the trunk circuit fails both the subsequent diagnostic test and the automatic retest, it is considered to have a detectable fault and the system attempts to remove it from service. On the other hand, if the trunk circuit passes a diagnostic test or if for some reason a diagnostic test is not completed, then an error is attributed to the circuit. The trunk error analysis program processes these errors.

10.02 Trouble symptoms on trunks include network continuity check failures, SD failures, and failures on any other peripheral order bus action when type 1 or type 2 failure options are used. Other trouble symptoms include transmitter timeouts, internal circuit check failures, invalid trunk state conditions found by audits, receiver partial dials, permanent signals, etc. Also a trunk failure during an automatic progression test is considered a trouble symptom.

10.03 After a trouble occurs, many suspected circuits will pass a subsequent diagnosis since many trouble symptoms are independent of the individual trunk circuit submitted for test. Each trunk circuit that passes a diagnostic test initiated by a trouble symptom is considered to have been in error. A statistical check is made to ensure that such a circuit is not performing marginally or that it does not have a fault which cannot be detected by the diagnosis. All other trouble symptoms which do not result in diagnostic failures are also considered to be errors. Included are the cases where

(a) A diagnostic test is not provided

(b) A diagnostic test is blocked

(c) A diagnostic test request is denied on the number of trunks already out of service or on the maintenance request list

(d) A diagnostic test is skipped because a trouble condition is suspected in the connecting office (that is, transmitter timeouts).

10.04 The processing of errors by the trunk error analysis program consists of a relatively simple comparison of the number of errors associated with a particular trunk to the number of errors occurring on other trunks in the same group. A maximum of 20 trunks can be analyzed simultaneously. Running counts are maintained on the number of errors occurring on each trunk under study and on other trunks in the group. If a sufficient number of errors occurs on other trunks of the group before a specified number occurs on the particular trunk, the trunk is considered to be normal. On the other hand, if the specified limit of errors on the particular trunk is reached first, the trunk is considered to have a high-error rate. It is understood in this determination that the trunk has been involved in a statistically improbable percentage of the total number of errors occurring in the group.

10.05 The validity of this method of error analysis depends on the errors occurring on a group of normal (no fault) circuits that are distributed either randomly or uniformly over the group. In general, this is not the distribution case when the system has a fixed preference for selecting trunks of normal call usage. Unequal trunk usage implies unequal probability of errors for trunks in the same group. For this reason, error analysis is not applied to incoming trunks or to 2-way trunks.

10.06 A high-error rate on a trunk implies either a marginal performance or a fault undetectable by a diagnostic program. When a trunk is found to have a high-error rate, the trunk error analysis program removes the trunk from service unless a maintenance limit is exceeded or unless, at the time the high-error condition is detected, the system has not completed abandoning a call involving the trunk. When the system removes such a trunk from service, it initiates a repeat test of the circuit in an effort to trap a marginal condition.

10.07 Because of the statistical check of trunk errors, the need for individual trunk error output messages is obviated. Once daily and upon request, this program prints the number of errors for various sources. Also counts are made of the disposition of errors treated by the trunk error analysis program. Although the system usually selects the trunks upon which error analysis operates, facility is also provided for manual selection.

11. LINE INSULATION TESTS (2-WIRE ESS ONLY)

11.01 The automatic line insulation testing (ALIT) program tests line insulation values. All idle lines, except ground-start PBX lines, are tested. The ALIT program may be started at a specified time of day by program control or by requests from either the local test desk or the maintenance TTY at the master control center. The following tests are performed.

(a) ***Short Circuit and Ring to Ground (SRG):*** This test is used to detect trouble in drop wire and inside wire at the subscriber's premises (leaks between tip and ring) and in open wire conductors (leaks from ring to ground). It also checks for end point contactor closure.

(b) ***Tip and Ring to Ground (TRG):*** This test checks for trouble in the cable terminals and the cable sheath (leaks from tip or ring to ground).

(c) ***Foreign Potential on Tip or Ring (FEMF):*** This test is used to detect defects in underground and in overhead cable sheaths (leaks from tip or ring to battery).

11.02 The sensitivity of the test circuit is variable depending upon the range specified and the cross-connections of the circuit. There are four test ranges available for each of the three sets of cross-connections.

11.03 When a line insulation failure is detected, it is reported via the ALIT TTY. Since more than one office may have access to the same ALIT TTY, additional messages are typed at the start and the end of a test cycle to identify the type of test, range, and originating office.

12. AUDIT PROGRAMS

12.01 Audit programs operate under control of the MAC program. The function of audit programs is to detect and to eliminate erroneous or inconsistent information in the CS memory. All of the audit programs are scheduled by the MAC program for routine execution throughout the day (usually on an hourly basis). Most of the audit programs can be requested via the TTY. Also the audit programs are requested by the MAC program when program indicators warn of possible trouble.

12.02 To detect errors in the CS, audit programs use the following methods.

(a) Compare redundant (distinct from duplicate) information. (Redundant information exists if it can be found in a number of different forms within the CS. Duplicate information exists when an exact copy of it is available in a PS or in a different CS.)

(b) Determine whether portions of the information violate certain restrictions.

(c) Compare information in the CS with associated backup information in the PSs.

12.03 When the audit program detects either a restriction violation or a disagreement between redundant information, all equipment associated with the suspect CS memory location is made idle and is available for use when required. When a disagreement between the PS and the CS memory is detected, the audit program corrects the CS. Audit program results are printed out by the maintenance TTY of the master control center.

12.04 The emergency action control program combines the individual audits into groups for use when the existence of trouble makes it necessary to check and/or to reconstruct the information of certain CS areas or of an entire CS memory. The emergency action control program is nondeferrable and must be completed before call processing can be resumed.

12.05 The audits are used in effecting the phases of system reinitialization (Section 231-113-301 for 2-wire ESS or 231-413-301 for 4-wire ESS). The phases of system reinitialization can be used only if a working configuration of a CC, a PS, and a PS bus exists. In the event of severe trouble, a working configuration is established by emergency action circuits in the CC.

12.06 All phases of reinitialization are called for by the program when certain trouble conditions are detected. The phases are performed in order starting with phase 1 and continuing until the trouble is corrected. Each phase performs a more extensive operation than the preceding phase. The maintenance attendant can use switches to select any phase of reinitialization. The phases should be selected in order.

13. OFF-LINE MAINTENANCE PROGRAMS

13.01 Off-line maintenance is performed when the CC, the PS, the PS bus, and in some cases the CS are separated from the rest of the system and are combined to repeatedly carry out selected programs. Off-line maintenance is used either to isolate marginal conditions that may be frequency-dependent or to isolate marginal troubles that may be configuration-dependent. For example, a high-error count may be experienced only when a particular CC bus and a PS are used together. The source of errors, in this case, cannot be isolated by normal methods because switching out any of the units involved causes the errors to stop. Off-line maintenance is initiated manually via the maintenance TTY. Appropriate tests can be performed repeatedly off-line until the source of trouble is identified.