

**REMREED NETWORK FAULT  
ISOLATION UTILITIES  
SOFTWARE SUBSYSTEM DESCRIPTION  
NO. 3 ELECTRONIC SWITCHING SYSTEM**

CONTENTS	PAGE	CONTENTS	PAGE
1. GENERAL . . . . .	1	5. The Common Pegging (CPEG) Subroutine . . . . .	20
2. PERIPHERAL UNIT FAULT RECOVERY (PURC) INTERFACE—OVERVIEW . . . . .	3	6. Manual Start of the Grid Error Count Function . . . . .	21
3. GRID ERROR COUNT (GEC) FUNCTION . . . . .	3	7. Grid Error Count State Diagram . . . . .	23
A. Automatic Operation . . . . .	3	8. Automatic and Manual Start of the Remreed Pattern Analysis . . . . .	24
B. Manual Operation . . . . .	6	9. Pattern Analysis Multiscan Function . . . . .	25
4. REMREED PATTERN ANALYSIS (RPA) . . . . .	8	10. Pattern Analysis Control Bit Information . . . . .	26
A. Automatic Start . . . . .	8	11. Send the Set of Maintenance Network Orders (Toggle) . . . . .	27
B. Manual Start . . . . .	9		
C. Send Network Order Set . . . . .	9		
D. Analysis of Group Check Pattern . . . . .	10	Tables	
E. Print Fault Location . . . . .	11	A. Grid Error Count Results . . . . .	12
5. GLOSSARY . . . . .	11	B. Subroutines Controlled by the Network Analysis MSF (NWAMSF) . . . . .	13

**Figures**

1. Automatic Operation of the Grid Error Count Function . . . . .	14
2. Grid Error Count Control Block . . . . .	15
3. The AUTO_ST and AUTO_ST_CK Subroutines . . . . .	16
4. Printing of the Grid Error Count Results . . . . .	17

**1. GENERAL**

1.01 This section provides an overview of the software in No. 3 Electronic Switching System (ESS) offices using Issue 2 and later issues of the 3E3 Generic Program to analyze and isolate faults in the nonduplicated portion of the network. The software improvements include two separate diagnostic/recovery type tools that isolate remreed network faults. The two major improvements are:

**NOTICE**

Not for use or disclosure outside the  
Bell System except under written agreement

SOFTWARE TOOL	FUNCTION
(1) Grid Error Count	To tabulate the number of network orders that have failed on each grid accessible from a control complex and to present the results in a form which allows fault isolation to be readily noticeable
(2) Remreed Pattern Analysis	To send a set of maintenance network controller orders and to analyze the group check data results in an attempt to isolate the fault.

A control complex (as defined only for the software covered by this document) is a pair of network controllers and associated grids isolated from other controllers and grids by the control (pulse) path.

**1.02** Whenever this section is reissued, the reason for reissue will be listed in this paragraph.

**1.03** Part 5 contains a glossary of terms, abbreviations, and acronyms necessary for comprehension of the information contained in this document.

**1.04** The format for the results of the grid error count are presented in Table A. Table B contains the major subroutines controlled by the Network Analysis Multiscan Function (MSF) subroutine.

**1.05** The following Sections or documents provide information related to this section.

233-153-140 Peripheral Unit Fault Recovery, Software Subsystem Description, No. 3 Electronic Switching System

OM-3H300-03 Output Message Manual

IM-3H300-03 Input Message Manual

**1.06** Before the Grid Error Count Routine (GDCNT) or Remreed Network Error Analysis Program (NWCEA) can be triggered automatically or manually (via TTY input message), the NW\_ANA bit in translations must be set. The following input message will set the bit:

```
RC:OFFICE/  
SET 0/  
END!
```

See the appropriate input message manual for further details.

**1.07** The most common remreed network faults usually fall into one of the following categories:

DOCUMENT NO.	TITLE	FAILURE	RESULT
233-153-105	Programmed Maintenance Aids, Software Subsystem Description No. 3 Electronic Switching System		
233-153-120	Peripheral Unit Diagnostics, Software Subsystem Description, No. 3 Electronic Switching System	Lightning Damage	Generally causes multiple faults to occur. Both the network fabric and controllers are susceptible to damage.
233-153-135	Error Analysis And Fault Detection, Software Subsystem Description, No. 3 Electronic Switching System	Pulse Matrix Faults	Typically has a widespread effect on the fabric but does not affect the controllers.

Isolating faults is difficult because a large portion of the network can be affected by a single fault, thus concealing the location of a failure. The GDCNT and NWCEA programs provide tools for maintenance personnel to analyze or tabulate data in order to isolate the location of a fault.

**1.08** The GDCNT program pegs certain network orders that fail against the appropriate grids. The results are presented in a tabulated format. (See Table A.) The offending portion of the network is isolated by comparing the results of the entire network. When the offending portion of the network is compared to the entire network, a significant amount of additional failures or the absence of failures is detected. When the number of failures that occur on each grid reaches 65,535, further failures will not increment the counter. Therefore, the results should be outputted via a manual request before the counter reaches its maximum value.

## **2. PERIPHERAL UNIT FAULT RECOVERY (PURC) INTERFACE—OVERVIEW**

**2.01** The Network Controller Queue Monitor (QMON) subroutine in the Network Queue and Timing Hopper Monitor (QTMON) program is called once every 10-millisecond interrupt. The function of QMON is to administer the network queues and issue all network orders needed for peripheral processing. When a network order fails for the first time, QMON calls the RECOVER subroutine in PURC. Call processing interrupts are then blocked to prevent additional tasks from interfering with recovery procedures. The PURC program determines that the recovery request was not caused by bad translation data. Next, any multiscan functions on the off-line side are aborted unless the force or lock keys are depressed. Power control is activated to ensure that peripheral controller +24V power has not been inadvertently turned off. When these conditions exist, QTMON is requested to retry the same order on the same processor. If the order passes, PURC increments the transient error counter for that controller. If the transient error counter is 7 or greater or the retry failed, a SYC (system control) switch is attempted. A second retry is then requested. When the second retry fails, the problem is assumed to be in the nonduplicated portion (remreed network fabric) of the network.

**2.02** When processing in PURC reaches the point that the problem can be assumed to be in the nonduplicated portion, the GDCNT and NWCEA programs are triggered automatically (when not inhibited). When the pattern analysis is being executed, some of the recovery action (in PURC) must be inhibited. A multiple fault condition existing for 75 seconds will cause the EXCESSER subroutine in PURC to initialize the system. The EXCESSER subroutine is not allowed to execute (inhibited) when the pattern analysis is being run to prevent an unintentional abortion. A SYC switch must also be inhibited when the transient error counter is greater than 7 or the first retry fails so that the series of maintenance network orders can be executed from the same SYC.

## **3. GRID ERROR COUNT (GEC) FUNCTION**

**3.01** The GEC function can be invoked automatically by RECOVERY in PURC or manually via TTY input message. When the GEC function is activated, the address of the grid involved in a network order that has failed with a group check or pulser OK failure is determined. A counter is assigned to every grid in the office. Legal failures cause the appropriate counters to be incremented (pegged). The results of pegging by the active GEC function may be printed upon manual request. The Grid Count Pegging (GCPEG) and Grid Count Auto Start (GCAST) subroutines in the Grid Error Count Routine (GDCNT) program control the pegging of legal failures. The GEC function is activated by logic at GCAST. Once activated, subsequent failures on the same control complex are pegged via GCPEG. Legal network orders that fail on both control complexes are pegged by GCAST.

### **A. Automatic Operation**

**3.02** Recovery will start the Grid Error Count (GEC) function automatically when a network order fails the retry attempts on both network controllers of a control complex. (See Fig. 1—Automatic Operation of the GEC Function.) The RECOVER subroutine in PURC calls GCAST in GDCNT. The GCAST subroutine determines if the GEC function is automatically or manually started. Logic beginning at GCAST tests the office options bit NW\_ANA in translations. When NW\_ANA is not set, the GEC function is not allowed even if manually invoked. The legal network orders that are pegged by the GEC function are group check and pulser OK failures.

**SECTION 233-153-145**

All other failures are ignored by GCAST, and no action is taken.

**3.03** The GCAST subroutine passes control to the Application Grid Address Determining (AGCDTRM) subroutine. The initial task of AGCDTRM is to determine if the failure is a legal network order. If so, the control complex involved must be equipped for a start or restart to occur. Logic at ACC\_CK is called to determine if the specified control complex is equipped. When not equipped a "NO-GOOD" response is returned. Otherwise, the side on which the failure occurred

is obtained. The ORDER\_CK logic returns a "NO-GOOD" response unless the failing network order is group check or pulser OK. Only group check and pulser OK failures are pegged.

**3.04** Words one and two of the order sent to the network controller are obtained to determine the grid(s) involved in the failure. The logic at Build Grid Address (BLD\_GD\_ADDR) determines the specific type of network order, number of grids involved, and address of the grid(s) involved as follows:

NETWORK ORDER	GRIDS INVOLVED	BRANCH
Close Stage 1 and Stage 2	2	G2
False Cross and Ground	2	G2
High and Dry	1	G1
Release Stage 1 and Stage 2	2	G2
Close Stage 3	1 (third stage)	G13
Release Stage 3	1 (third stage)	G13

All grid addresses are determined by logic at G1, G2, and G13. The control complex and grid address data are saved. Control returns to the subroutine that called AGCDTRM (GCAST or GCPEG).

Block. (See Fig. 2.) The GEC function is auto started or restarted depending on the current state and previous (old) state. Actual activation depends on the current GEC control state as follows:

**3.05** The current state of the GEC function is obtained from the Grid Error Count Control

<u>CURRENT STATE</u>	<u>ALLOW GEC ACTIVATION</u>	<u>BRANCH TO LOGIC</u>
System Normal (Idle)	Yes	AUTO_ST
Auto Started with Failure	No	PEG_1_SIDE
Manual Started	Yes	AUTO_ST_CK
Manual Started with Failure	Yes	AUTO_ST_CK
System Normal, Auto Inhibited	No	A_OUT
Auto Started, Auto Inhibited with Failure	No	A_OUT
Manual Started, Auto Inhibited	No	A_OUT
Manual Started, Auto Inhibited with Failure	No	A_OUT

**3.06** Subroutine A\_OUT returns control to PURC. Subroutine PEG\_1\_SIDE pegs the failure for the specified side of the control complex which failed via the common pegging subroutine. (See paragraph 3.11 for details.) The AUTO\_ST and AUTO\_ST\_CK subroutines automatically start the GEC function when a network order fails all retries,

and the auto start is not inhibited. (See Fig. 3—The AUTO\_ST and AUTO\_ST\_CK Subroutines.) An auto restart is terminology denoting that an auto start has occurred after a manual start. For example, if the GEC function has been started manually and the old state was system normal, a legal order failing all retries on the same or another

control complex will cause an auto restart. When an auto start or restart occurs, the following output message is printed on the maintenance TTY:

```
A tt NW COUNT START AUTO dd
```

The dd indicates which control complex caused the auto start or restart. The AUTO\_ST subroutine updates the GEC control block. The old state of the GEC control block is replaced with the present current state. The current state is then jammed to control state 001 (auto started with failure).

**3.07** The next task of AUTO\_ST is to determine if the previous counts are to be printed. If the input message OP:NW;COUNT! causes a print to be in progress or the previous current state was system normal, the previous count results are not printed. Otherwise, the results of any previous manual counts will be printed when an auto start or restart occurs. The print state in the GEC control block is updated to print the results.

**3.08** There are two entry points to handle the printing of the GEC results. (See Fig. 4—Printing of the GEC Results.) Entry point GCOP starts or restarts and monitors the printing. The printing of the GEC results is a nondeferrable

base level loop job. The BLMMAST subroutine in the Application Portion of the Base Level Monitor (BLMMA) program gives the logic at GCOP control on even base level loops. The other entry point, GCPC, is given control by the application version of Common TTY Handler (CTTYH) program after the results of the counting for one side has been printed. The GCPC subroutine will then print the other side of the results or terminate printing depending on the print state (PR\_STATE).

**3.09** The GEC print state is obtained from the GEC control block (see Fig. 2) by code in subroutine BGN\_P. The logic at BGN\_P then determines which entry point was called. When the GCPC entry point is called, one of eight branches occurs, based on the print state. Seven of the eight branches result in termination of any printing. The other branch (print state = printing side 0) transfers control to logic at point P\_0\_C when the printing of side 0 has completed. The code at P\_0\_C changes the print state to start printing side 1 since both sides are to be printed. Control is now given to subroutine DTRM\_P. Entry point GCOP also gives control to DTRM\_P. Logic at DTRM\_P branches to one of eight locations and determines the action to take according to the following print states:

<u>GEC PRINT CONTROL</u>	<u>ACTION</u>	<u>BRANCH</u>
Idle	None	Wait
Error	Terminate	P_ERR
Printing Manually Specified Side	Monitor Printing of One Side	P_
Start Printing Manually Specified Side	Start Printing One Side	ST_P
Printing Side 1	Monitor Printing of Side 1	P_1
Start Printing Side 1	Start Printing Side 1 (Side 0 Already Printed)	ST_P_1
Printing Side 0	Monitor Printing of Side 0)	P_0
Start Printing Side 0	Start Printing Both Sides (Side 0 First)	ST_P_0

If an initialization occurs while the printing is being monitored, the printing is restarted via logic at P, P\_1, and P\_0. When an illegal print state is encountered, an error message is printed, the print state is idled, and the printing function is terminated via P\_ERR. Logic at WAIT invokes no action (returns control). Logic at ST\_P\_1, ST\_P\_0, ST\_P specifies the side or sides to be printed. The print state is updated to printing and the data is printed. When printing is complete, the printing function is terminated and all print controls are reset.

**3.10** The AUTO\_ST subroutine continues processing logic at FINISH. (See Fig. 3.) The counters are reset by calling the Grid Error Count Clear (GCCLR) subroutine for auto starts if the previous current state was not system normal. The GCCLR subroutine clears (resets) specified blocs of call store used by the GDCNT program. The GEC clear state is loaded with the clear count block state when the counters are to be reset. Depending on the GEC clear state, GCCLR branches as follows:

BRANCH	CLEAR STATE
C_OUT	Clear Nothing
C_STAMP	Clear a Time and Date Stamp

<u>CURRENT STATE</u>	<u>ACTION</u>	<u>BRANCH</u>
System Normal	None	NO_PEG
Auto Started, Failure	Peg	PEG
Manual Started	Change State and Peg	ST_CH_PEG
Manual Started, Failure	Peg	PEG
System Normal, Auto Inhibited	None	NO_PEG
Auto Started, Auto Inhibited	Peg	PEG
Manual Started, Auto Inhibited	Change State and Peg	ST_CH_PEG
Manual Started, Auto Inhibited Failure	Peg	PEG

The manual-started and manual-started auto-inhibit control states are updated via ST\_CH\_PEG to reflect the failure before pegging is done. Logic at PEG increments the peg count for appropriate counters when the count is less than 65,535. Once the counter has reached 65,535, the counter will not increment until reset. Once the GEC function is started, further failures on the same control complex are pegged by RECOVER calling the GEC peg

C_CNTR	Clear a Set of Counters
C_CONT	Clear the Control Block
C_ALL	Clear all Call Store

When clearing is complete, C\_OUT returns control.

**3.11** The FINISH subroutine then calls ST\_STAMP. The ST\_STAMP subroutine returns the time and date the GEC function was started. The results of the GEC function can be arranged chronologically since the output message contains the time and date. (See Table A.)

**3.12** The Common Pegging (CPEG) subroutine is called next by FINISH to peg the failure for the current side. (See Fig. 5—The Common Pegging Subroutine.) Since the failure occurred on both sides, CPEG is called again to peg the other side. Logic at CPEG determines if the failure should be pegged. The detected failure must be for the same control complex that invoked the GEC function. If not, the failure is not pegged. For the case when both sides fail, CPEG is forced by FINISH to peg both sides. The action taken by CPEG depends on the current GEC control state as follows:

(GCPEG) subroutine. Logic at GCAST does the pegging when a network order fails all retries.

#### B. Manual Operation

**3.13** The Grid Error Count (GEC) function may be started manually at any time via a TTY input message. (See Fig. 6—Manual Start of the GEC Function.) Invoking the GEC function manually

aids in isolation of a transient network problem. The following input messages are used in controlling

the GEC function states (See Fig. 7—GEC State Diagram.) as follows:

<u>MESSAGE</u>	<u>PURPOSE</u>
ALW:NW;COUNT!	To allow auto start or restart
BEGIN:NW c;COUNT!	To manually start or restart GEC
INH:NW; COUNT!	To inhibit auto start or restart
OP:NW s;COUNT!	To output results of current GEC counters
RESET:NW;COUNT!	To idle GEC function and abort counting

**3.14** The Grid Error Count Input Message Handler (GCIM) subroutine in GDCNT is called to handle the GEC function input messages. Logic at GCIM detects which input message has been entered. For a reset request, the entire grid count call store area is cleared (via GCCLR). A reset will change the GEC function control state to system normal (idle). The reset message should be used after a fault has been removed from the system to prevent the pegging of different faults from affecting the results. For an output request of the current GEC to occur, the print state must be idle and the GEC function must be active. A legal side should also be specified. The counters are used interchangeably at different times to store counts or copy counts from one set to the other set. A network fabric fault can be detected differently by each of the two network controllers of the control complex. Therefore, two sets of counters, one for each side, are used. Two additional sets of counters are used to store the counts accumulated for a control complex for printing. The GEC Copy (GCCPY) subroutine copies the current counters along with the time and date stamp to the other counters and stamp block for printing. The results of the counters are now ready to be printed. The new print control state is stored to allow printing.

**3.15** For the begin (manual start or restart) request, logic beginning at GCIM checks for a specified control complex side that is equipped. The time and date of invokement is saved via STAMP\_. To insure valid results, the counters to be used are cleared (via GCCLR). The current state and old state are updated and stored. When

logic at GCIM detects the allow request, the INHIBIT bit is zeroed. The INHIBIT bit is set for the inhibit request. The old state and current state are updated.

**3.16** The input messages provided are used to terminate the automatic and manual operation. A manual restart (BEGIN:NW c;COUNT!) or manual reset (RESET:NW;COUNT!) ends an auto-started function. A manual start or restart may be terminated by an auto restart, a manual reset, or another manual restart.

**3.17** When a manual restart occurs after an auto start, legal failures are pegged (via GCPEG) as long as the failures occur on the same control complex specified by the manual restart. When a different control complex is involved in a failure, an auto restart (via GCAST) will occur to do the pegging. The results of the previous pegging are automatically printed.

**3.18** The Grid Count Pegging (GCPEG) subroutine is called to do the pegging for the automatic and manual operation when only one side of the control complex is involved in failures. Once the GEC function is started, all subsequent failures on the same control complex are pegged against the appropriate grids by GCPEG. Logic at GCPEG determines if the GEC function is allowed by testing bit NW\_ANA for a set condition. The AGCDTRM subroutine (described in paragraphs 3.03 and 3.04) is called to determine if the failure is to be pegged. Legal network order failures are pegged according to the current control state as follows:

<u>CURRENT STATE</u>	<u>BRANCH</u>
System Normal	NOT_LEGAL
Auto Started, Failure	LEGAL
Manual Started	LEGAL
Manual Started, Failure	LEGAL
System Normal, Auto Inhibited	NOT_LEGAL
Auto Started, Auto Inhibited, Failure	LEGAL
Manual Started, Auto Inhibited	LEGAL
Manual Started, Auto Inhibited, Failure	LEGAL

Logic at LEGAL pegs the failure via CPEG (see paragraph 3.11 for details) and updates the control block if necessary. Control is returned via NOT\_LEGAL when the failure is not to be pegged.

#### 4. REMREED PATTERN ANALYSIS (RPA)

##### A. Automatic Start

**4.01** The remreed pattern analysis (RPA) is designed to isolate single pulse matrix faults in the network fabric. Certain combinations of multiple faults may also be isolated by the analysis. Greater success in isolating a fault is achieved by performing the analysis before another fault occurs or a traffic-sensitive fault disappears. Therefore, the Remreed Network Error Analysis (NWCEA) program is resident, and triggered automatically when a network order fails on both controllers.

**4.02** The Peripheral Unit Fault Recovery (PURC) program (described in Part 2) uses three subroutines in NWCEA as an interface (PTNINH, NWANACT, and PTN\_EX). (See Fig. 8—Automatic and Manual Start of the Remreed Pattern Analysis.) Once the fault is assumed to be in the nonduplicated portion of the network fabric, subroutine PTN\_EX in NWCEA is called. The logic at PTN\_EX requests the RPA to be automatically started by calling subroutine MSFREQ in the Common Base Level Monitor (CBLM) when the following conditions are met:

- (1) NW\_ANA bit is set
- (2) RPA has not been executed in the past 30 minutes (check PTN\_INH bit)

(3) A network order has failed on both controllers

(4) The failing order has a nonzero group check response

(5) RPA is not in progress (check PTN\_ACT bit).

The MSFREQ subroutine (part of the multiscan function controller) determines if the automatic start request is accepted or denied. The NWCEA program is a resident program under the control of the multiscan function controller. Work is done on NWCEA at the end of every base level loop when free time is available.

**4.03** The Pattern Analysis Inhibit Counter (PTNINH) subroutine prevents RPA from being automatically invoked if the recent changeable NW\_ANA bit is zero. The PTN\_INH bit is set (via EXITNWA) whenever RPA is completed. When the PTN\_INH bit is first set, RPA cannot be automatically invoked for 30 minutes. The PTNINH subroutine is called once every second via FLTRECS in PURC. The records are incremented at the PTN\_CNT counter. The PTN\_INH bit is reset after 30 minutes to re-enable the analysis.

**4.04** When an automatic request for the MSF is accepted, the AUTO bit (indicates an automatic start) is set so logic at Network Analysis MSF will call the AUTOPT subroutine to print the auto start message. When RPA is automatically invoked via PURC, the NW ANAL START AUTO output message is printed. The PTN\_EX subroutine then sets the PTN\_ACT bit to indicate RPA is in progress. Logic at PTN\_EX will not try to re-invoke RPA before it is finished. The PTN\_EX subroutine

then determines which control complex (Control Frame 0 or 1) failed the network order and stores it in the PTN\_CC bit. Control now returns to RECOVER in PURC to choose the best operating SYC before returning control to QTMON.

## B. Manual Start

**4.05** The TTY input messages are handled by the Network Analysis, Grid Error Count,

and Status Input Message Interface subroutine in NWCEA. The Grid Error Count input messages (described in paragraph 3.12) cause control to be given to subroutine GCIM (GEC Input Message Handler—described in paragraph 3.13). The network analysis and status input messages used for manual control are:

<u>MESSAGE</u>	<u>PURPOSE</u>
ALW:MSF 12	To enable auto start of RPA when inhibited
BEGIN:NW;ANAL	To manually start RPA on a specified control complex
INH:MSF 12	To inhibit auto start of RPA
OP:NW;ANAL	To print results of last RPA
OP:NW;STAT	To output the network status

For the OP:NW;STAT message, control is given to subroutine OP\_NW in the Trunk, Line, Service Circuit, and Network Link TTY Message handler (TSUREQ). The analysis Input Message Handler (ANALIM) is given control when network analysis input messages are detected. The ANALIM subroutine processes the input messages and then calls MSFREQ in CBLM to request the MSF.

**4.06** When the dispatcher gives control to the MSF controller (request accepted) the Network Analysis MSF (NWAMSF) subroutine is called. (See Fig. 9—Pattern Analysis Multiscan Function.) The NWAMSF subroutine is the main supervisor for control and execution of the RPA. Logic at NWAMSF branches to several subroutines (see Table B) as determined by the control bits (See Fig. 10—Pattern Analysis Control Bit Information.) and the following entry codes:

- MSF In Progress Entry
- MSF Abort Entry
- Per Scan Entry
- First Entry
- Aborting an In-Progress MSF
- Aborting a Request.

**4.07** The NWAMSF subroutine distinguishes between an automatic and manual start by testing the AUTO control bit. When set, subroutine AUTOPT is called to print the auto start message. Otherwise (a manual start), the CCVALID control bit is checked to determine if a control complex has been specified. If not specified, an abort occurs (via ABRTNWA). Logic at points STINT (Start Initialization) and EINIT (End Initialization) initializes the buffer areas and call store needed to send the set of maintenance network orders and to collect and analyze the group check responses.

## C. Send Network Order Set

**4.08** A set of A-link, high and dry, release stage 2, and stage 3 network orders are sent from each network controller in order to access all portions of the network. (Only the on-line processor sends the set of orders.) The prescribed set of network orders must access every pulse path multiple and gate lead multiple (for each grid spanned by the multiple) at least once. The set of orders cannot exceed a maximum of 528 orders per controller. The pulser is inhibited since only group check data is needed and the state of any crosspoint does not change. Call processing can continue while the set of orders is being sent. An input message (BEGIN:NW;ANAL,CG) can be used to access a small portion of the network when necessary. A fault signature is obtained from the results of the group check responses. When the fault signature

from both controllers agree, the fault is expected to be in the remreed network fabric and not in one of the controllers.

**4.09** The NWAMSF subroutine sets the NORDGO control bit after zeroing the interrupt counter (INT\_CNT). The RPA is enabled to send the set of network orders when NORDGO is set. Every 10 milliseconds, a hardware interrupt occurs giving QTMON control. Logic at QTMON calls the QMON\_NW subroutine every interrupt. (See Fig. 11—Send Prescribed Set of Network Orders [Toggle].) Logic beginning at QMON\_NW tests the NORDGO control bit. If not set, control returns to QTMON. When set and work is to be done, subroutine ANAWRK is given control. The CMTOG (toggle) control bit is tested by ANAWRK. If set, a call processing order is sent. If not set, a maintenance order is sent. Every 10-millisecond interrupt a toggle occurs until the set of network orders has been sent. When maintenance orders are to be sent, the network queues are blocked. If a maintenance order was sent at the last interrupt, the group check response is stored (via READ\_GC). Otherwise, the next maintenance order in the sequence is sent via Send Maintenance Network Orders for RPA (SEND\_MO). Logic beginning at SEND\_MO calls CALNWAO to calculate the next network order to be sent. If no more orders are to be sent (orders complete), subroutine XQNWCMPL is given control. Logic at XQNWCMPL sets the ORD\_CMPL bit and zeroes the NORDGO bit. The pulser is enabled (via ENAPULS) and control returns to QTMON. When there are more network orders to be sent, SEND\_MO calls FORMORDA to format the network order for the No. 3 ESS. The pulser is inhibited and the first two words of the maintenance order are sent. Maintenance orders are sent until the interrupt counter (INT\_CNT) is greater than or equal to the toggle counter (INT\_TOG). Then, the CMTOG bit is set to allow a call processing order to be sent during the next interrupt.

**4.10** The next interrupt QMON\_NW is given control again. Since the CMTOG control bit is set, subroutine CPTINT is given control. Logic at CPTINIT unblocks the network queues. If a maintenance order has been sent during the last interrupt, the group check response is stored and the pulser is enabled. Then processing continues as if no maintenance order has been sent. Call processing orders are sent until the interrupt counter is greater than or equal to the toggle counter.

When this condition occurs ( $INT\_CNT \geq INT\_TOG$ ), the CMTOG control bit is zeroed to send a maintenance order during the next interrupt. The MBIT control bit is also zeroed to denote that a call processing order has been sent during this interrupt. Control is then returned to QTMON. The toggling back and forth continues by alternately sending a call processing order and a maintenance order until the prescribed set of network orders has been sent. When a failure is detected or the orders have completed, the pulser is enabled for call processing. Control then returns to QTMON.

**4.11** While the maintenance network orders are being sent, the NWAMSF subroutine forces a break every 100 milliseconds. (See Fig. 9) At the end of the base level loop, the MSF controller is given control. Control is then passed to NWAMSF where the ORD\_CMPL (orders complete) control bit is checked. When set, QMON\_NW has completed executing the prescribed set of network orders correctly. Otherwise, logic at NWAMSF continues to wait (via a 100-milliseconds break) until the orders have completed. When the orders are complete, the ORD\_ACT (order active) control bit is zeroed. The set of network orders has been sent by one controller. If the system control is locked, logic at ONEPASS performs the RPA for one controller (via CPA\_CTL). Otherwise, logic at SWCTRL complements the CTRL (controller) control bit to send the set of orders by the other controller. (SW\_CTL is called to switch controllers.) When the task of sending the set of orders by both controllers has been completed, a SYC switch (via APPL\_SW) occurs to run on the best side, if necessary (side with the fewest out-of-service controllers.)

#### D. Analysis of Group Check Pattern

**4.12** The group check responses from the set of maintenance orders form a group check signature. The signature is a pattern which represents the state of the remreed network fabric during the time that the prescribed set of network orders was sent. The Common Pattern Analysis Controller (CPA\_CTL) subroutine is given control by NWAMSF to determine if one or both controllers are to be analyzed. The Common Network Analysis (CNWANAL) subroutine is given control by CPA\_CTL to perform the RPA.

**4.13** The RPA verifies that both controllers are good when the results from both controllers

match. A mismatch indicates a controller problem may exist. The results of the RPA are printed by subroutine RESPT. After the analysis is complete for both controllers, the PTN\_INH bit is set to inhibit an automatic start of RPA. The PTN\_ACT, ORD\_ACT, and AUTO bits are zeroed before control returns to normal call processing.

#### E. Print Fault Location

**4.14** The results of RPA for one controller are printed on the maintenance TTY in one block (3 lines) as follows:

```
tt REPT NW ANAL

cs 1ST cg conc isg isw
cs 2ND cg conc osg osw
cs 3RD 3sg lo/hi 3sw

cs — c=Control Frame, s=SYC

cg — Concentrator Group (1-7 if c=0) or
      (8-15 if c=1)

conc — Concentrator (0-1)

isg — Input Switch Group (0-2)

isw — Input Switch (0-7)

osg — Output Switch Group (0-1)

osw — Output Switch (0-3)

3sg — 3rd Stage Switch Group (0-3)

lo/hi — 3rd Stage Grid (L or H)

3sw — 3rd Stage Switch (0-7)
```

The data following 1ST, 2ND, and 3RD (stages of the network) indicates fault isolation. A “\_” character in any data field indicates no isolation. When the RPA is run on both controllers, the results are printed on two blocks (six lines), and one block (three lines) if RPA is run on one controller.

**4.15** The RPA can isolate a fault to an identified portion of the network (switch, grid, or

two grids). For example, the REPT NW ANAL output message may identify a grid. In some cases, replacing the grid will not clear the fault. When a fault is on the cabling to a grid, the fault appears to be in the grid (by RPA). Thus, the fault has only been isolated to a portion of the network associated with the specified grid.

#### 5. GLOSSARY

**5.01** To aid in the use and understanding of this document, definitions of terms used frequently are given.

**Auto Restart**—An auto restart occurs when a manual start is followed by an automatic start.

**Auto Start**—An automatic activation of the GEC orders fail all retries.

**Control Complex**—(As defined, applies to the software covered by this document.) A pair of network controllers and associated grids isolated from other controllers and grids by the control (pulse) path.

**Counter**—A storage device used to represent the number of occurrences of an event.

ESS—Electronic Switching System

**Grid Error Count (GEC)**—The pegging of certain network orders that have failed on each grid accessible from a control complex; performed in order to present the results in a tabulated form, which allows fault isolation.

**Group Check Signature**—A pattern which represents the state of the remreed network fabric (insofar as the success or failure of PNP selection) during the time that the prescribed set of network orders was sent.

**MSF**—Multiscan Function

**Pegging**—Incrementing a counter to record the number of times an event occurs in a certain amount of time.

**Remreed Pattern Analysis (RPA)**—An attempt to isolate a network fabric fault by sending a set

of maintenance network controller orders and analyzing the group check data results.

**SYC**—System Control

**TTY**—Teletypewriter

**SSD**—Software Subsystem Description

TABLE A  
GRID ERROR COUNT RESULTS

tt	OP	NW	COUNT	c	s	date	time				
			CONCENTRATOR 0			CONCENTRATOR 1			OUTPUT	SW GP	
CG			0	1	2	0	1	2	0	1	
0		12345	0	0	0	0	0	0	0	0	
1			0	0	0	0	0	0	0	0	
2			0	0	0	0	0	0	0	0	
3			0	0	0	0	0	0	0	0	
4			0	0	0	0	0	0	0	0	
5			0	0	0	0	0	0	0	0	
6			0	0	0	0	0	0	0	0	
7			0	0	0	0	0	0	0	0	
			STAGE 3 SWITCH GROUP								
CF		0L	1L	0H	1H	2L	3L		2H	3H	
0		12345	0	0	0	0	0		0	0	

*Note:* Simulated output message format for one controller.

TABLE B

## SUBROUTINES CONTROLLED BY THE NETWORK ANALYSIS MSF (NWAMSF)

SUBROUTINE	DESCRIPTION
ABRTNWA (Abort Network Analysis)	Aborts entire RPA, prints abort message (ABT NW ANAL), and calls EXITNWA; or performs RPA for one controller only.
APPL_SW (Application Controller Switch)	A primary MSF subroutine that performs a SYNC switch to run on the best side.
AUTOPT (Automatic Print)	Prints the auto start output message (NW ANAL START AUTO).
CPA_CTL (Common Pattern Analysis Controller)	A primary MSF subroutine that determines if one or both controllers are to be analyzed and calls the CNWANAL (Common Network Analysis) subroutine to analyze the collected group check data.
CUR_CTL (Determine Current Controller)	A primary MSF subroutine that determines the currently active controller.
EXITNWA (Exit Network Analysis)	Inhibits RPA auto start for 30 minutes. Deactivates RPA and maintenance orders.
INT_RES (Initialize Results)	A MSF initialization subroutine that initializes the results of RPA for a specific controller.
INT_TREE (Initialize Tree)	A MSF initialization subroutine that initializes the network trees for controller 0 or 1.
MODFEQP (Modify Equipped Masks)	A MSF initialization subroutine that modifies masks of equipped network elements.
RESPT (Results Print)	Prints the results of RPA.
SET_EQP (Set Equipped Masks)	A MSF initialization subroutine that sets equipped mask bits in the network trees.
SW_CTL (Switch Controllers)	A primary MSF subroutine that switches peripheral controllers to send the set of maintenance orders to the other controller.
ZERODTA (Zero Group Check Data)	A MSF initialization subroutine that zeros the group check data for one controller.

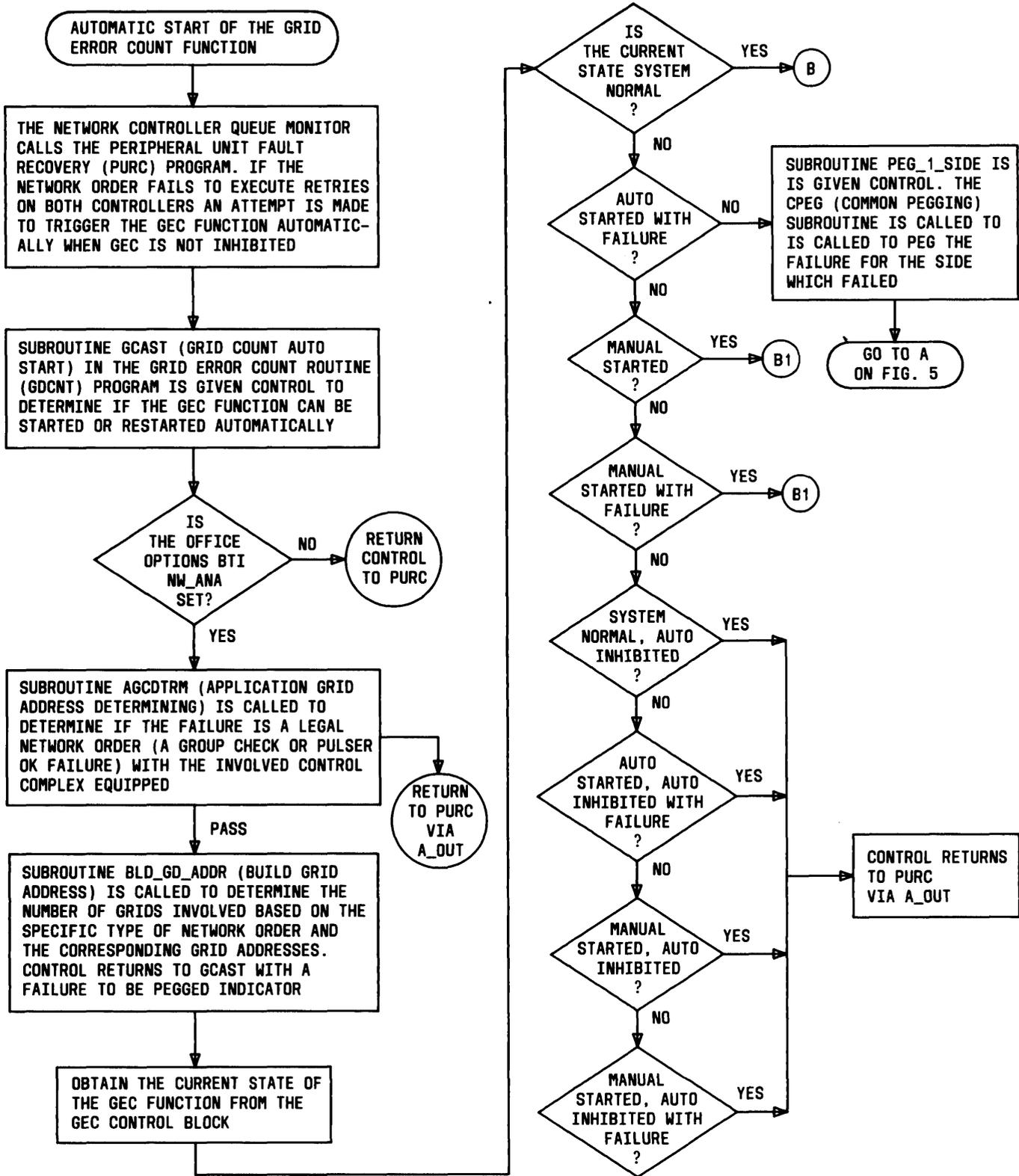
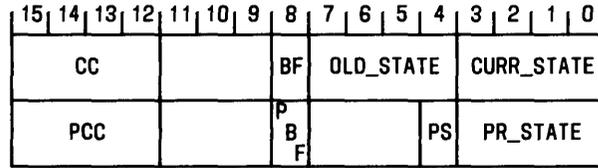


Fig. 1—Automatic Operation of the Grid Error Count Function



- CURR\_STATE - CURRENT CONTROL STATE OF GEC FUNCTION
- OLD\_STATE - OLD CONTROL STATE OF GEC FUNCTION
- BF (BLK\_FCTR) - STAMP AND COUNTER INDEX
- CC - CONTROL COMPLEX TO BE PEGGED
- PR\_STATE - PRINT STATE
- PS - SIDE TO BE PRINTED
- PBF (PBLK\_FCTR) - PRINT BLOCK FACTOR
- PCC - CONTROL COMPLEX TO BE PRINTED

CURRENT STATE	PRINT STATE
000 - SYSTEM NORMAL (IDLE)	000 - IDLE
001 - AUTO STARTED, FAILURE	001 - ERROR
010 - MANUAL STARTED	010 - PRINTING MANUALLY SPECIFIED SIDE
011 - MANUAL STARTED, FAILURE	011 - START PRINTING MANUALLY SPECIFIED SIDE
100 - SYSTEM NORMAL, FAILURE	100 - PRINTING SIDE 1
101 - AUTO STARTED, AUTO INHIBITED	101 - START PRINTING SIDE 1
110 - MANUAL STARTED, AUTO INHIBITED	110 - PRINTING SIDE 0
111 - MANUAL STARTED, AUTO INHIBITED, FAILURE	111 - START PRINTING SIDE 0

Fig. 2—Grid Error Count Control Block

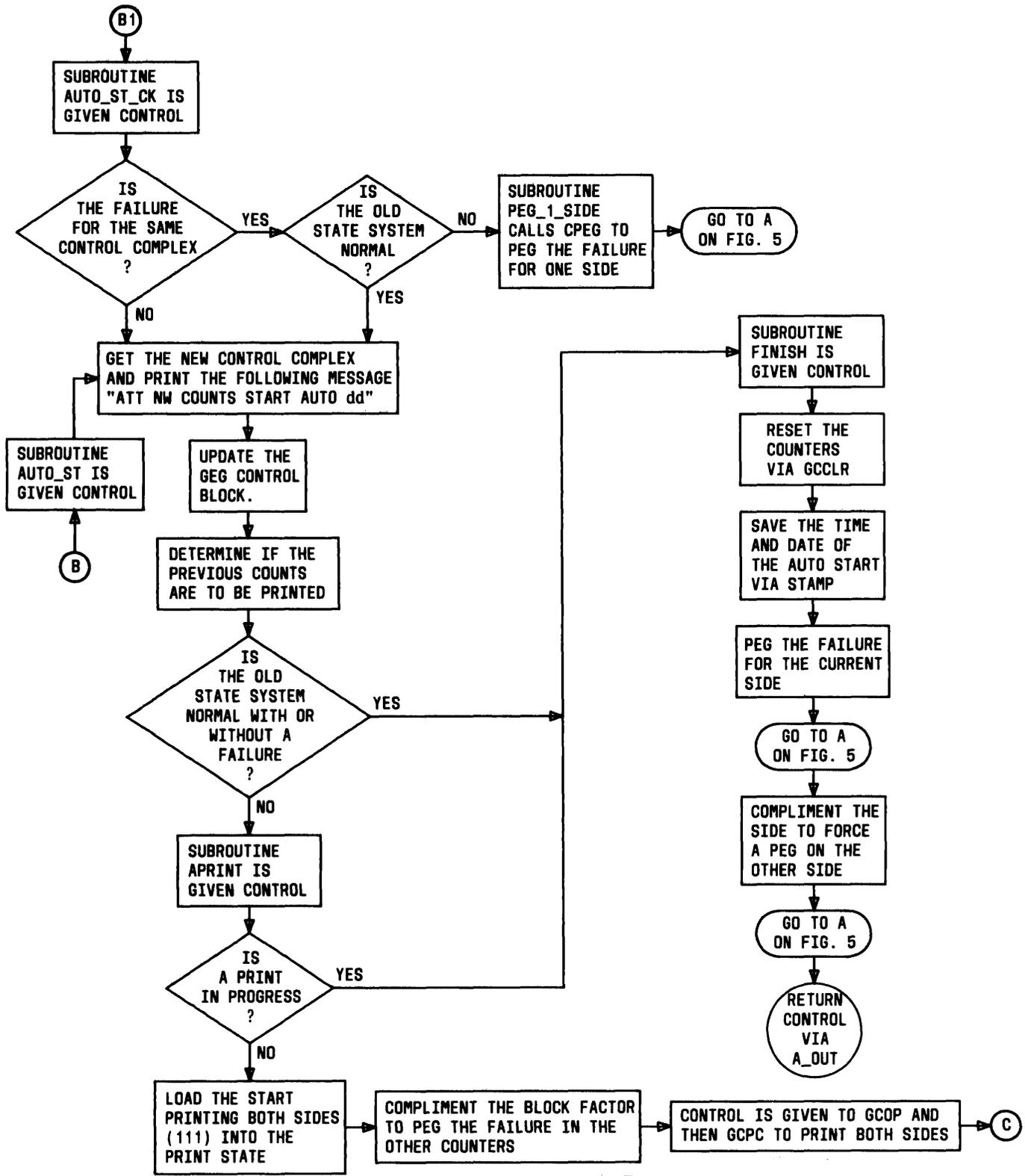


Fig. 3—The AUTO\_ST and AUTO\_ST\_CHK Subroutines

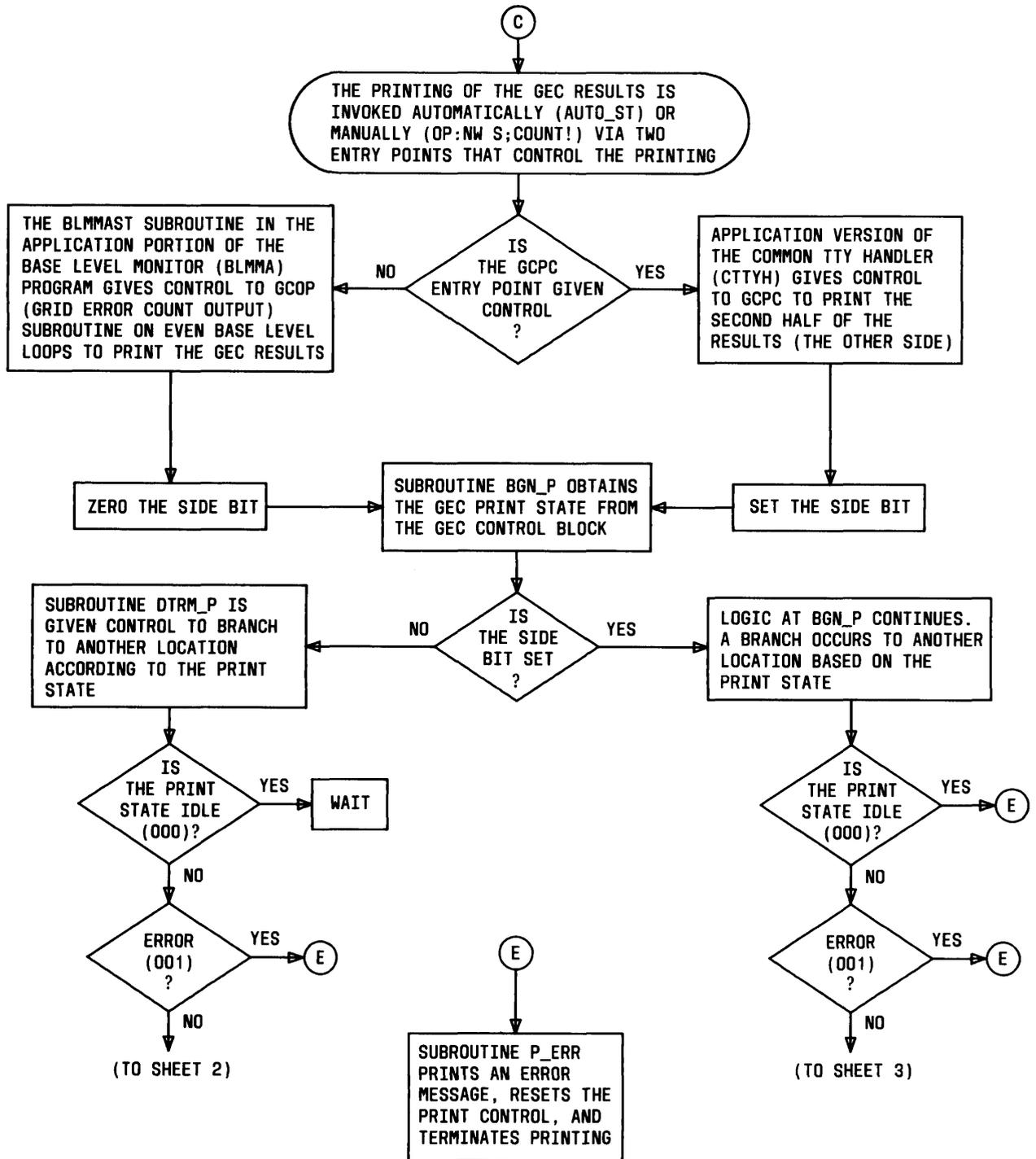


Fig. 4—Printing of the Grid Error Count Results (Sheet 1 of 3)

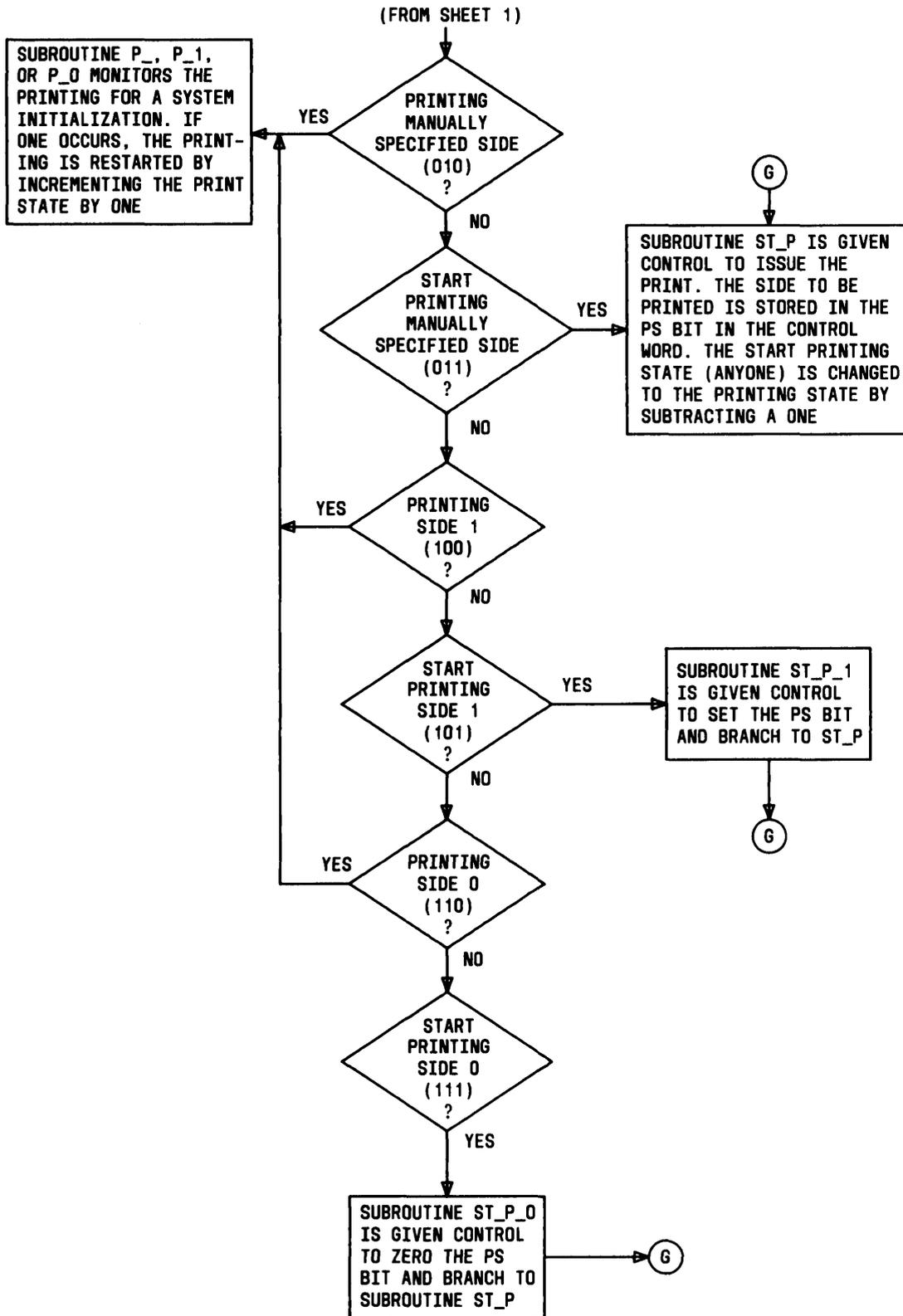


Fig. 4—Printing of the Grid Error Count Results (Sheet 2 of 3)

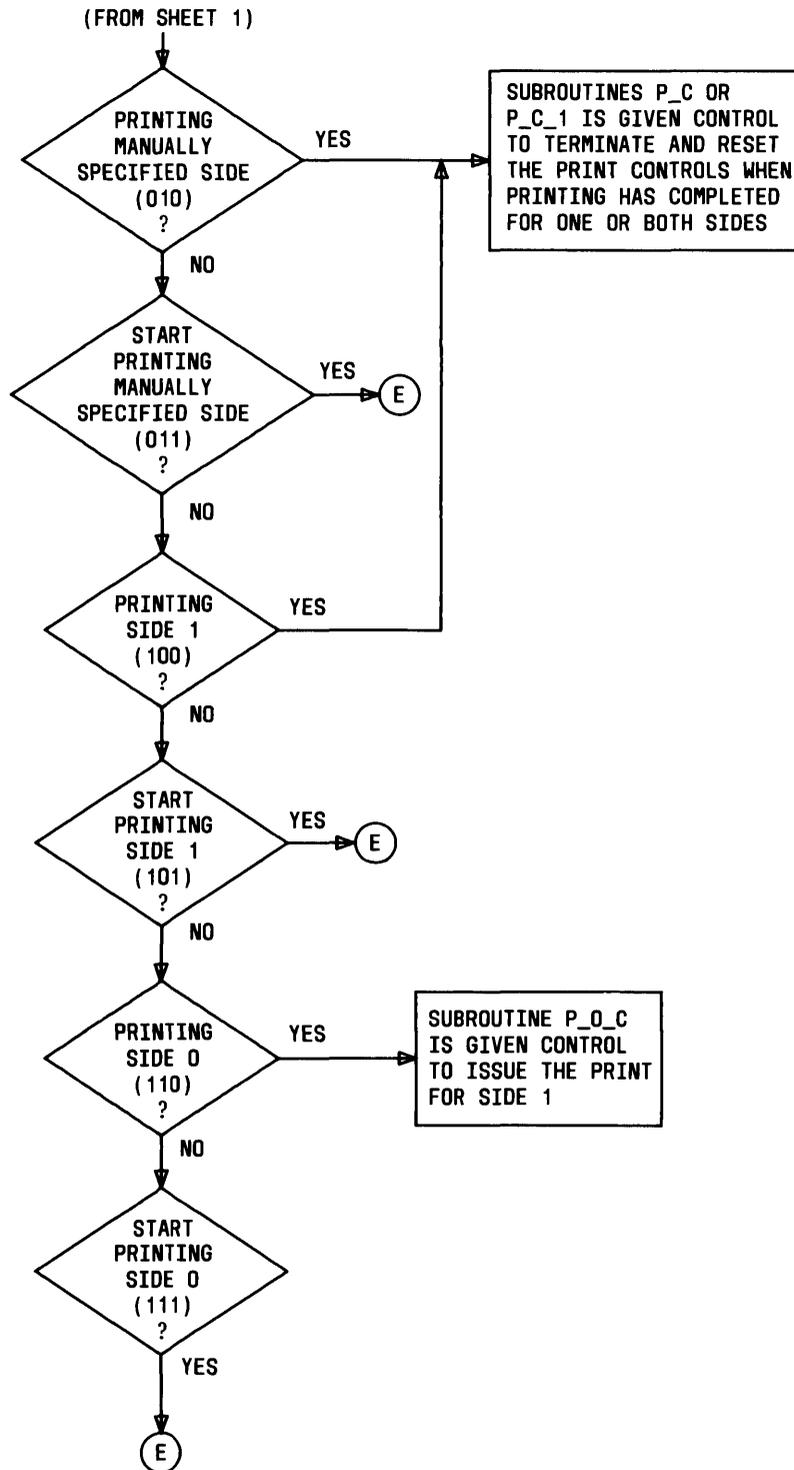


Fig. 4—Printing of the Grid Error Count Results (Sheet 3 of 3)

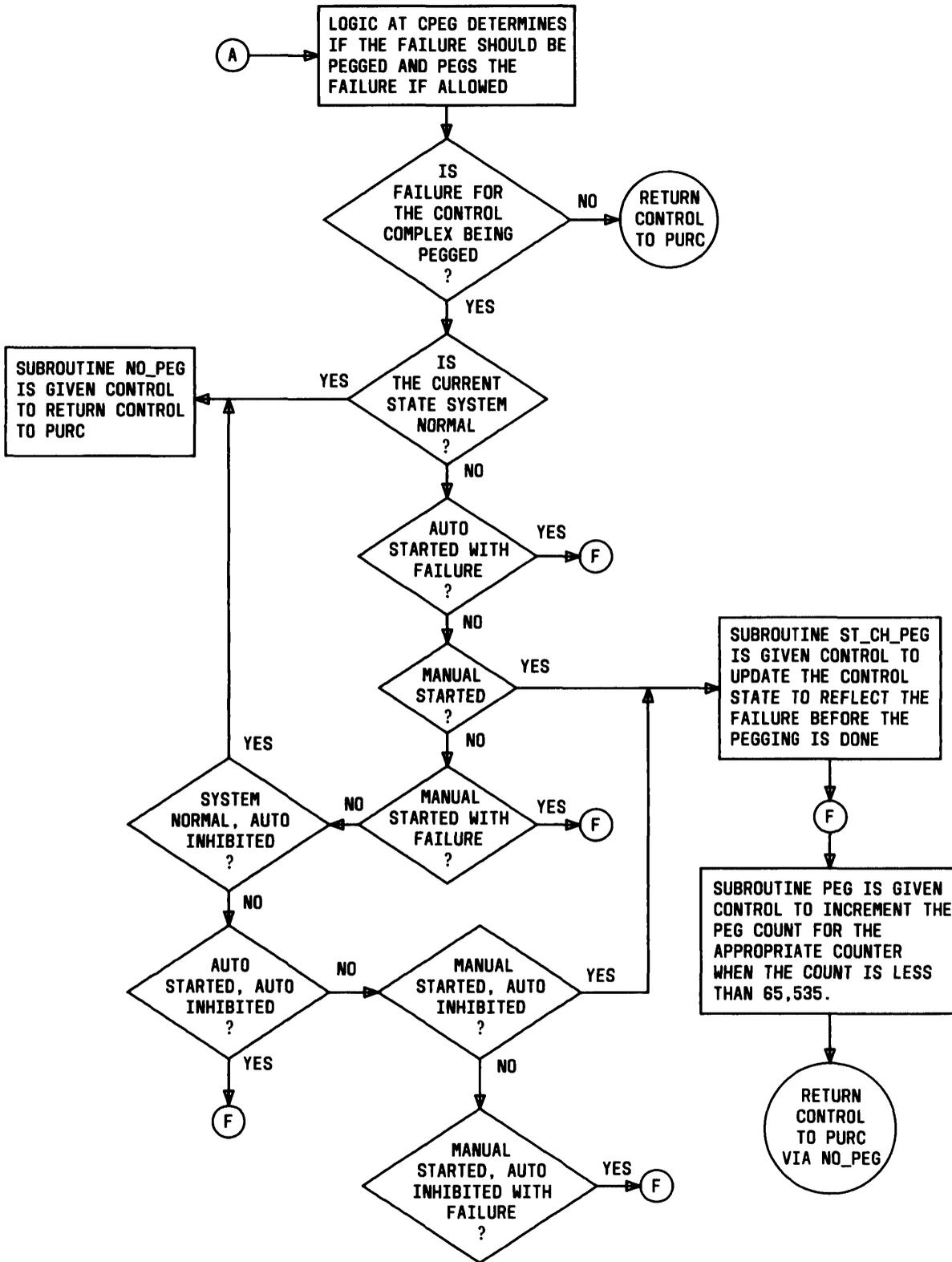


Fig. 5—The Common Pegging (CPEG) Subroutine

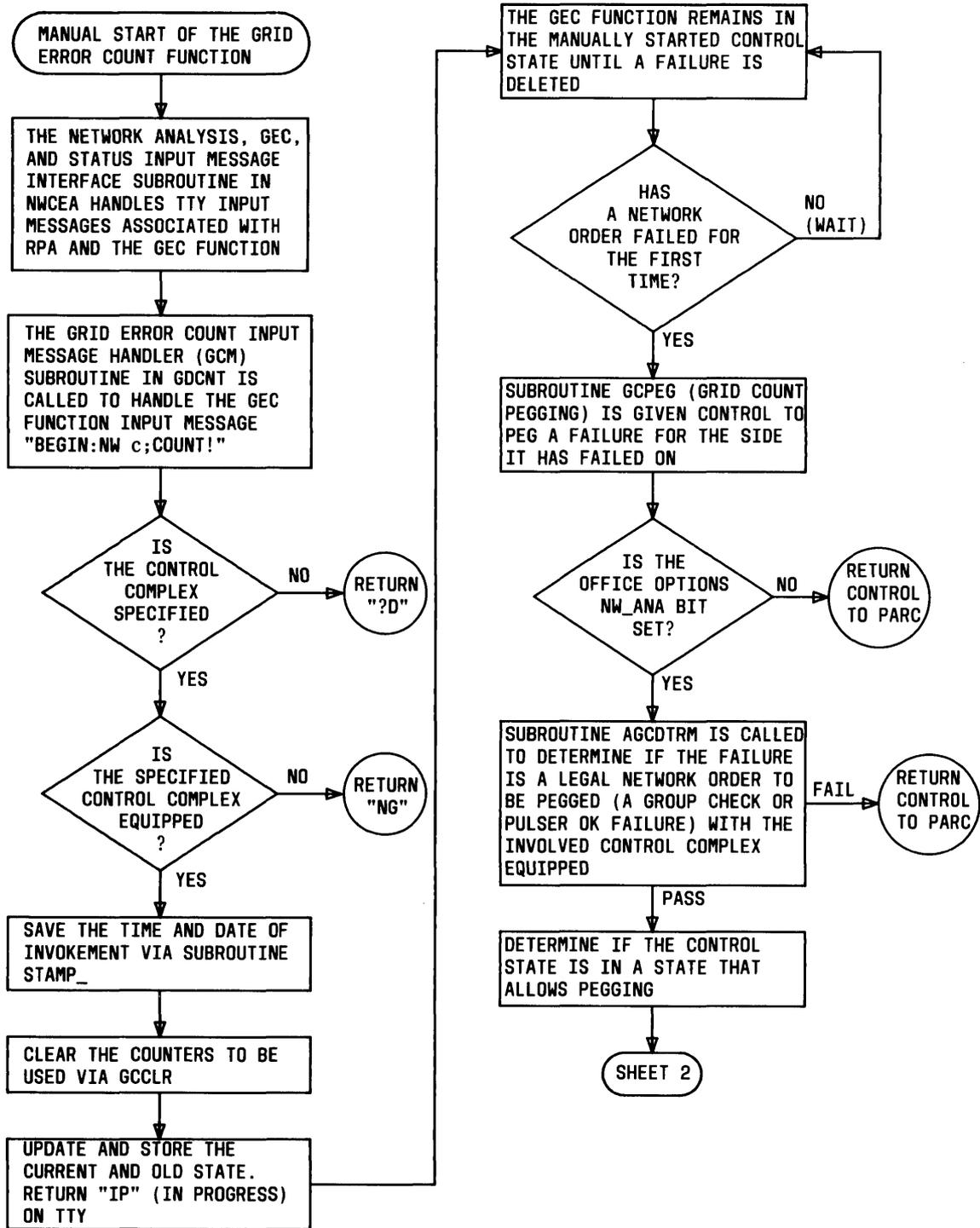


Fig. 6—Manual Start of the Grid Error Count Function (Sheet 1 of 2)

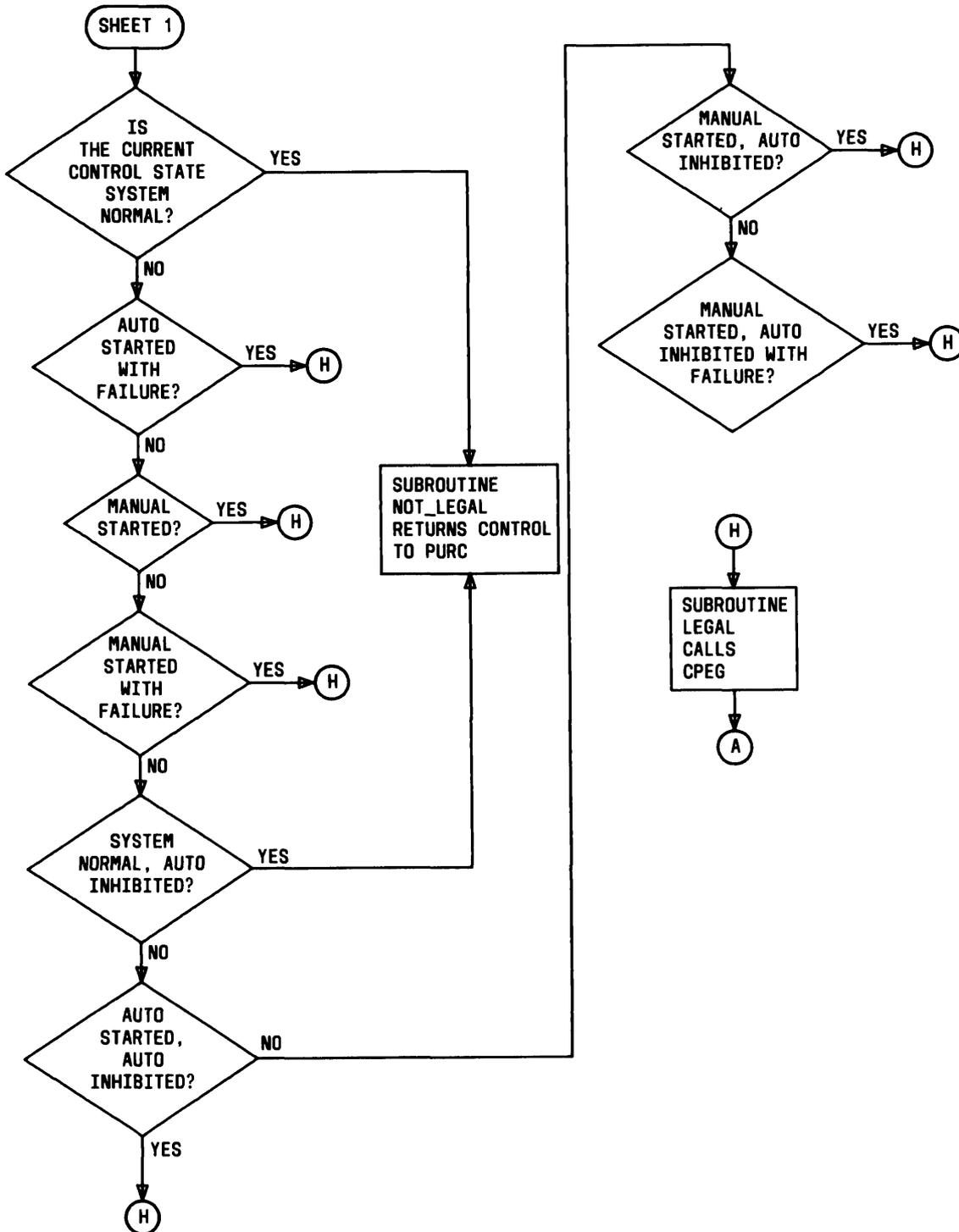


Fig. 6—Manual Start of the Grid Error Count Function (Sheet 2 of 2)

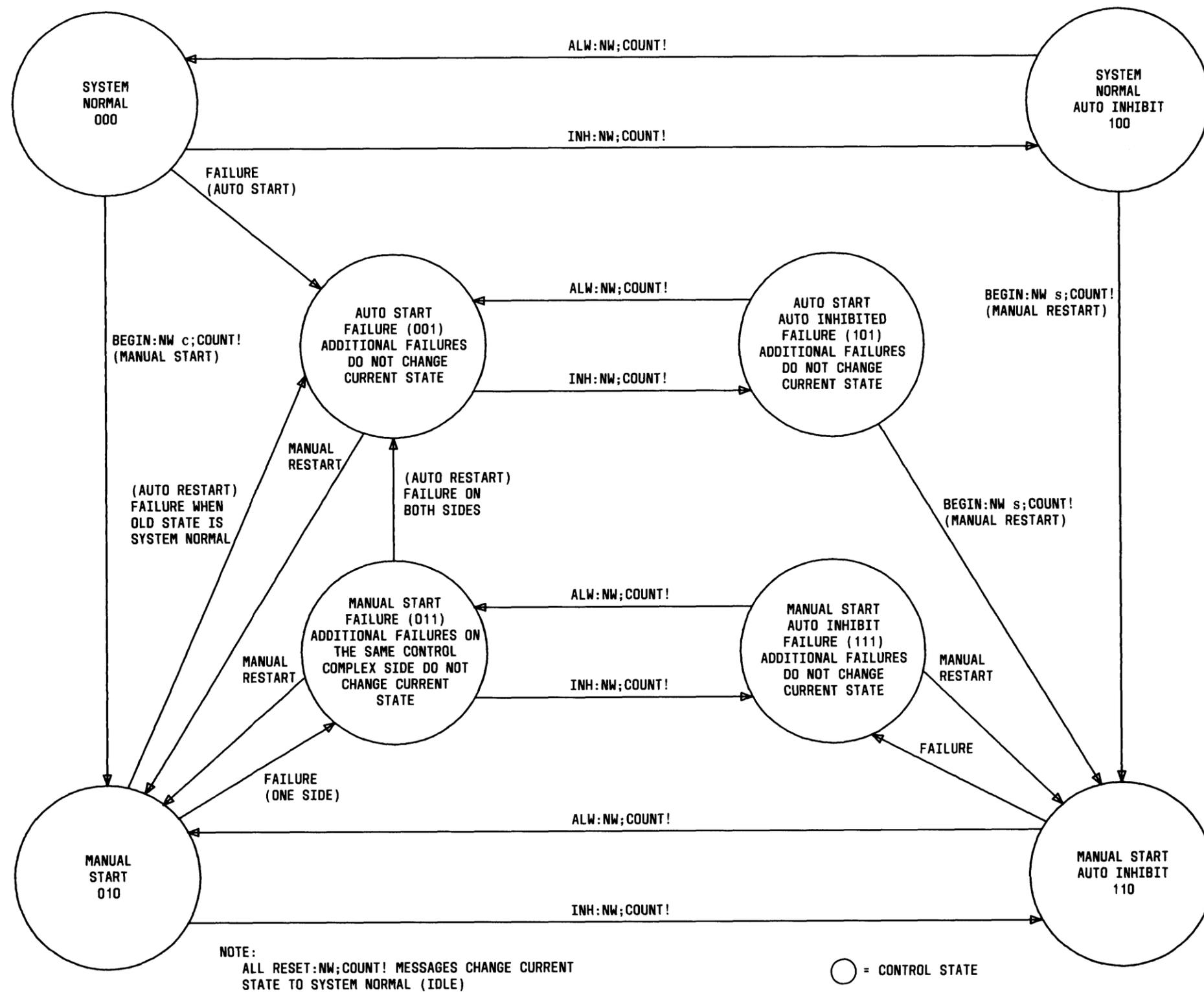


Fig. 7—Grid Error Count State Diagram



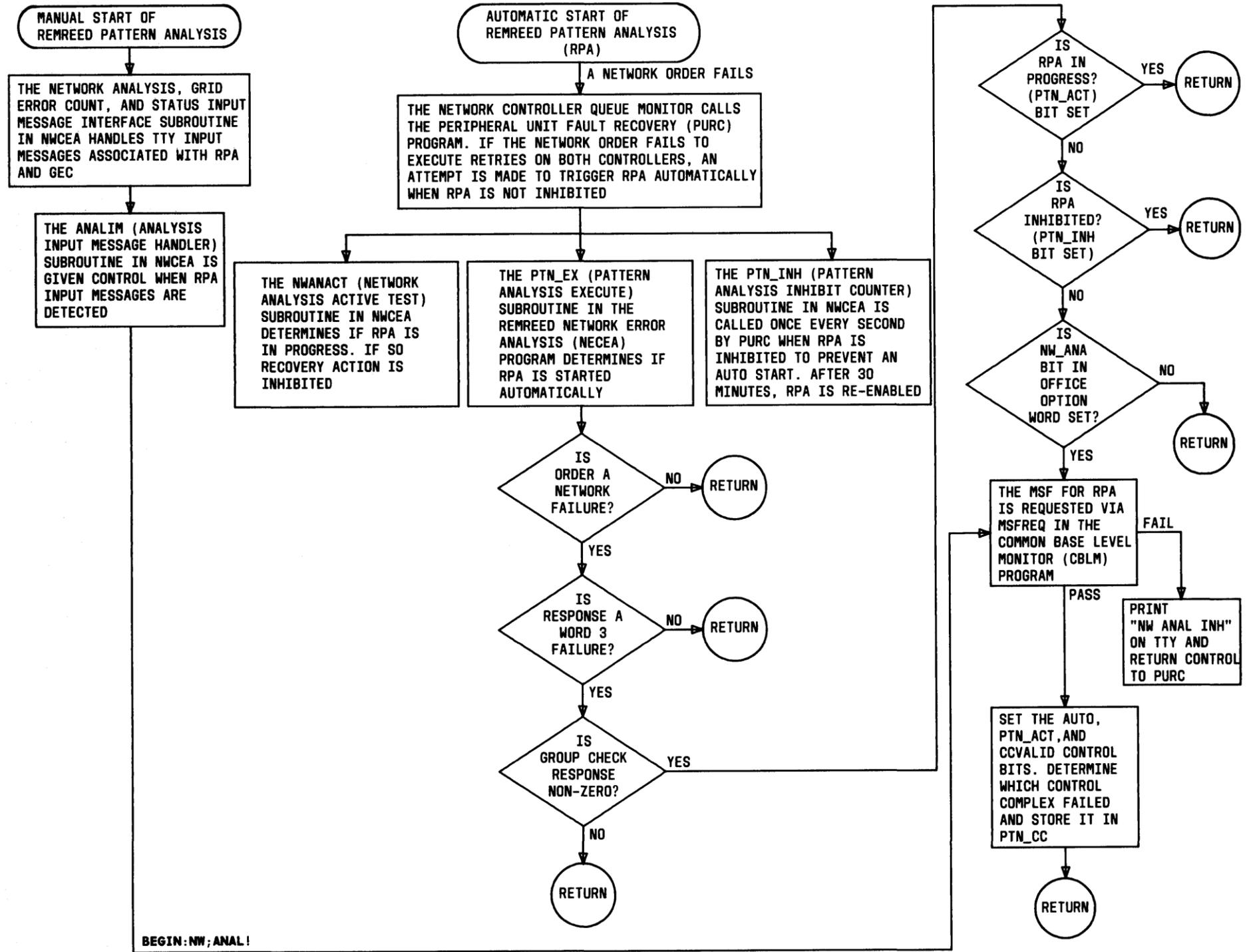


Fig. 8—Automatic and Manual Start of the Remreed Pattern Analysis



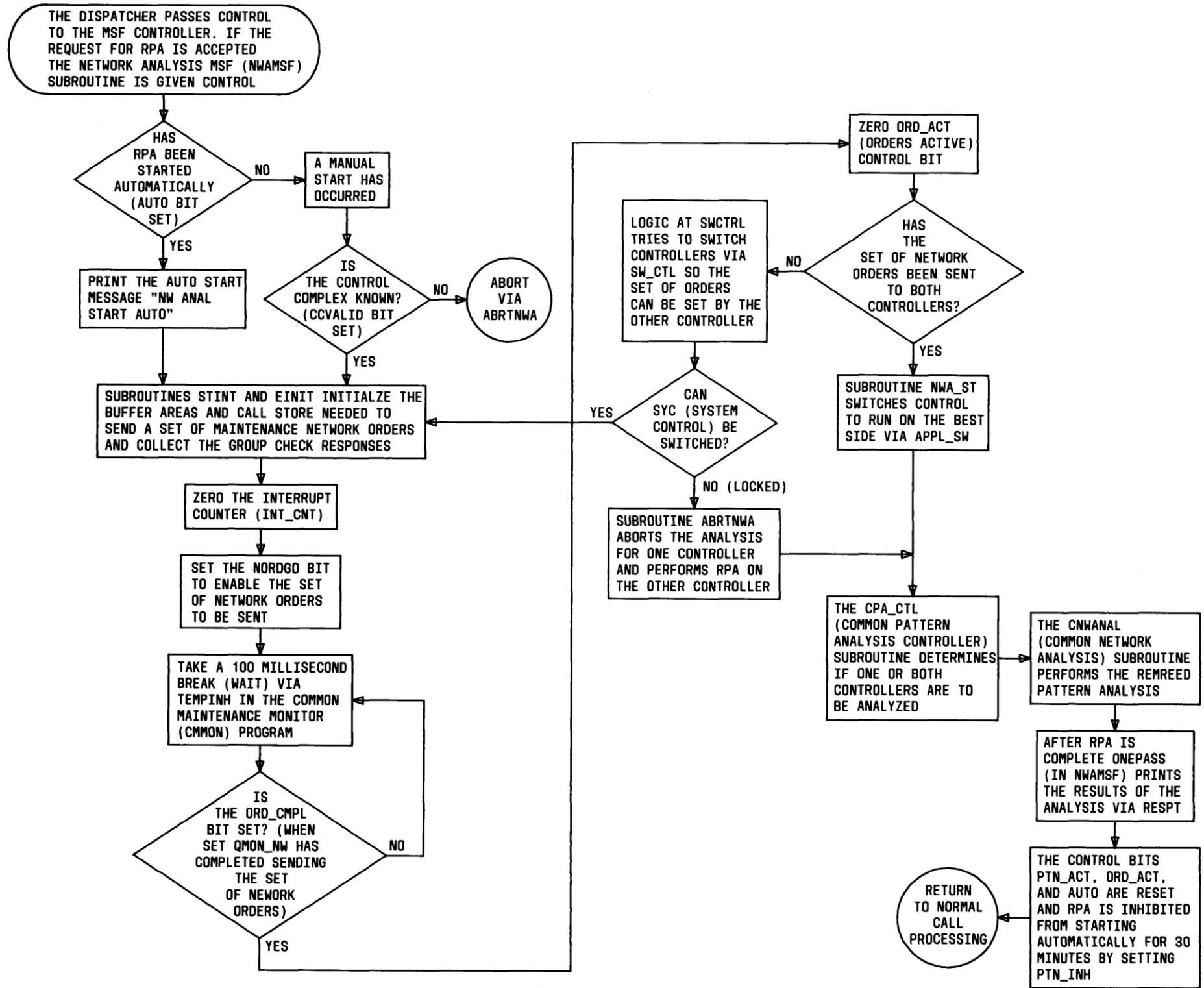


Fig. 9—Pattern Analysis Multiscan Function



	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
00	STOR_DLY				CTRL_F	PASS	MBIT	CMTOG	ORDFAIL	NORDGO	ORD_CMP	C1CPL	COCPL	CTRL	AUTO	PTN_ACT
01											ORDRET				ORD_ACT	
02	MOD_EQP		HIGHCG				LOWCG			CCVALID	PTN_CC					

- PTN\_ACT - REMREED PATTERN ANALYSIS IS IN PROGRESS (SET)
- AUTO - AN AUTOMATIC START HAS OCCURRED (SET)
- CTRL - CONTROLLER 0 OR 1
- COCPL - CONTROLLER 0 HAS COMPLETED SENDING SET OF ORDERS (SET)
- C1CPL - CONTROLLER 1 HAS COMPLETED SENDING SET OF ORDERS (SET)
- ORD\_CMP - ORDERS COMPLETE (SET)
- NORDGO - MAINTENANCE ORDERS ENABLED TO BE SENT (SET)
- ORDFAIL - MAINTENANCE ORDER HAS FAILED (SET)
- CMTOG - PASS CONTROL TO SEND A CALL PROCESSING ORDER (SET)  
PASS CONTROL TO SEND A MAINTENANCE ORDER (RESET)
- MBIT - A MAINTENANCE ORDER WAS SENT THE LAST INTERRUPT (SET)
- PASS - PRINTING IS FINISHED (RESET)
- CTRL\_F - SYNC SWITCH OCCURRED WHILE SENDING ORDERS (SET)
- STOR\_DLY - NUMBER OF ORDERS (MINUS 1) SENT PER NETWORK ADDRESS
- ORD\_ACT - DEACTIVATE MAINTENANCE ORDERS (RESET)
- ORDRET - ERROR RETURN CODE
- PTN\_CC - CONTROL COMPLEX 0 OR 1 FAILED A NETWORK ORDER
- CCVALID - CONTROL COMPLEX IS KNOWN TO BE VALID (SET)
- LOWCG - LOWEST CONCENTRATOR GROUP
- HIGHCG - HIGH CONCENTRATOR GROUP
- MOD\_EQP - MODIFY EQUIPPED MASKS (SET)

Fig. 10—Pattern Analysis Control Bit Information



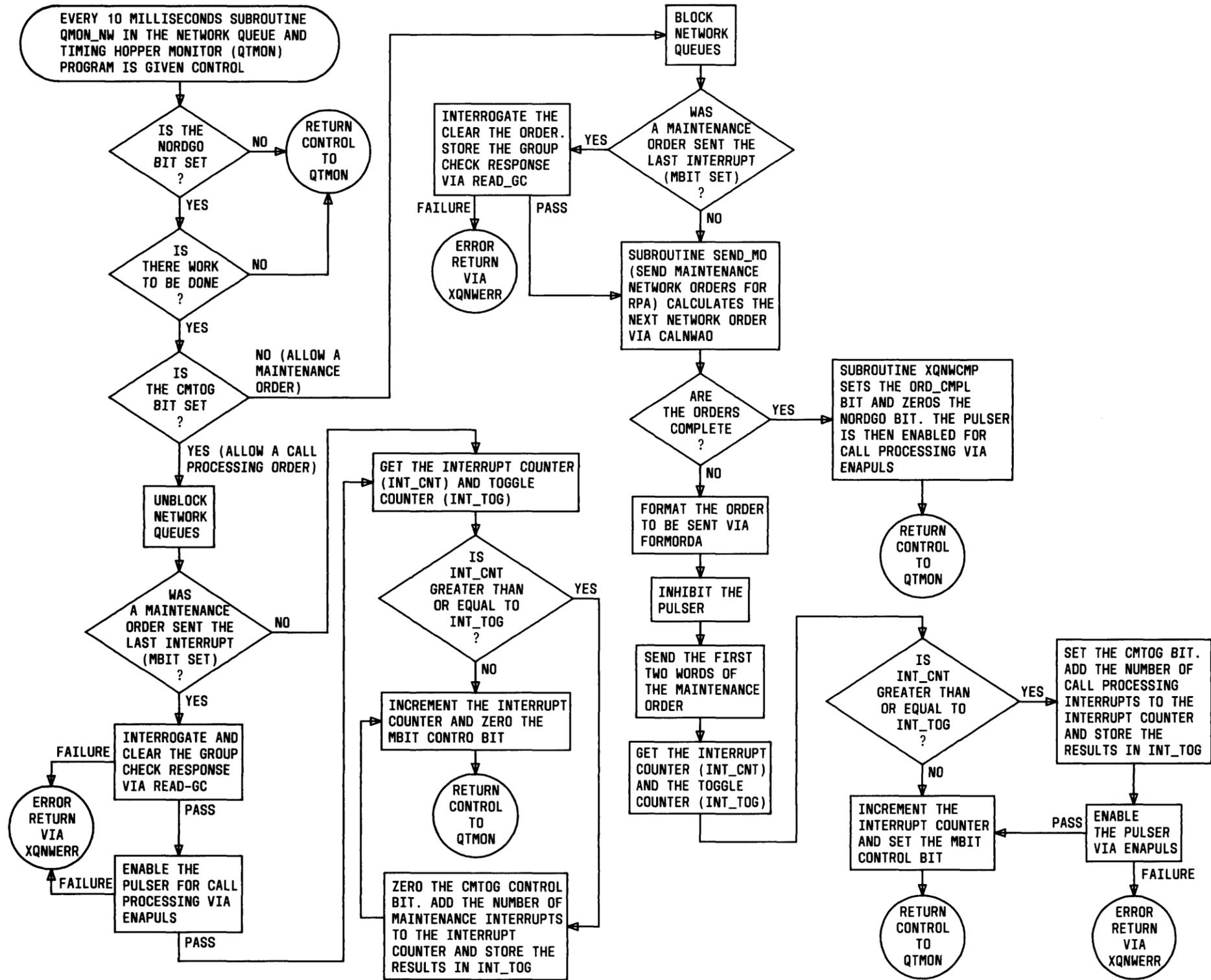


Fig. 11—Send the Set of Maintenance Network Orders (Toggle)

