

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
DEPARTMENT OF AERONAUTICS AND ASTRONAUTICS
INSTRUMENTATION LABORATORY
CAMBRIDGE, MASS. 02139

C. S. DRAPER
DIRECTOR

AG #58-67
23 February 1967

THROUGH: NASA Resident Apollo Office at MIT
Massachusetts Institute of Technology
Instrumentation Laboratory
Cambridge, Massachusetts

TO: National Aeronautics and Space Administration
Manned Spacecraft Center
Houston, Texas

Attention: Mr. W. Kelly

SUBJECT: Block II Restarts

The Block II CGC and LGC has a single indicator "restart" for several computer failure modes. Any failure among one of the following items will cause an indication of restart:

- +28 Volt Loss
- +14 Volt Loss
- + 4 Volt Loss
- Rupt Lock
- TC Trap
- Night Watchman
- Oscillator Fail
- Parity Fail

A more specific description of the PGNCS failure monitor is extracted from R-547 GSOP for AS-278.

As you are aware from our testing experience the restart alarm can be generated from external sources, a hardware failure internal in the computer, or programming and hardware design deficiencies. Examples of the latter are the Pseudo Verb 36 and the double entry problem in Block I.

AG #58-67
23 February 1967
Page 2.

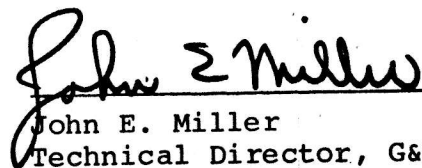
Every attempt should be made to understand and explain each restart. Because of the variety of sources for generating restart and less diagnostic information in Block II, some restarts will remain unexplained. For this reason a restart criterion has been submitted by MIT and a copy is enclosed.

The question always arises as to what happens if a restart occurs in flight. The action the computer takes upon receipt of a restart is that it is forced to location 4000. This starts the restart sequence. It is assumed that the computer central registers no longer contain good information. Each program which is restart protected has a number of points in the program which can be considered restart points. Each time the program reaches this point, sufficient information is stored to allow the program to be picked up at that point. Thus, had the program proceeded past that point and had a restart it would return to that point and sufficient information would exist for it to again proceed. This information is stored in duplicate - the information and the complement of the information in the erasable memory. A check is made, called a phase check, to determine the validity of this information. If the information is good, then restart has been accomplished. This check has then verified a small portion of the erasable memory and the assumption is made that if that is good then the whole of the erasable memory is good.

If the computer fails in its restart sequence, then it is assumed that the whole erasable memory is bad. A fresh start sequence is initiated.

The exact nature of the good restart sequence and the consequences of this sequence depend entirely upon the programs in progress at the time. The gyro compass program for example, can experience a reasonable number of restarts and still have a good alignment.

There seems to be no question that a hard failure creates the correct action.



John E. Miller

Technical Director, G&N System
Apollo Guidance and Navigation

JEM:vw

DISTRIBUTION:

R. Ragan
D. Hoag
L. Larson
A. Laats
B. Howie
E. Copps
A. Hopkins
E. Hall
R. Crisp
P. McGathy/MSC
R. Southers/MSC

NASA/RASPO/MIT
RASPO/ACED
MIT/IL at MSC
S. Laquidara MIT/MSC
T. Hempker MIT/NAA
M. Adams MIT/GAEC
G. Silver MIT/KSC

Massachusetts Institute of Technology
Instrumentation Laboratory
Cambridge, Massachusetts

AGC Programming Memo # 24

TO: Distribution
FROM: Edward M. Copps
DATE: February 23, 1967
SUBJECT: Discussion of the Software Associated with Computer Restart

This memo is intended to be briefing material on the problem of restarts.
There are 6 sections

1. Groundrules for Restart Protection
2. Software Associated with Restart
3. Problems of Restart Protection
4. Diagnostic Information Available
5. Memory Assigned to Restart Protection
6. Typical Program Flow During Restart

1. Groundrules for Restart Protection

The program designer presumes;

- a) that the information in the central and special registers should not be trusted,
- b) that if a small section of erasable is tested and found to contain correct information, that all of erasable is to be trusted,
- c) that fixed memory is to be trusted,
- d) that restarts are transient events, that they should be expected at infrequent intervals and that the program should be protected against infrequent restarts,
- e) that since some sections of program are impossible to restart protect and others are extremely difficult to restart protect, requiring great amounts of fixed storage and program design, no guarantee can be made about restart protection. However, experience with the design of flight programs for 204 and 206 indicate that the time that a program is restart susceptible is on the order of tens of microseconds per second.

2. Software Associated with Restart

The software associated with recovery from restart is in three log sections:

1. Fresh Start and Restart
2. Restart Tables and Restarts Routine
3. Phase Table Maintenance

The Fresh Start and Restart routines are closely related and have much common programming to save memory locations. This section contains coding executed following the involuntary transfer to location 4000_g caused by the restart.

The major tasks performed by the program immediately following a restart are a) to save diagnostic information, b) to initialize the system programs, such as the EXECUTIVE, T4Rupt, etc., to a dormant and "well behaved" condition that will permit a return to the processors that were active at the time of the restart, c) to check for the mark reject-error light reset "lock out" prevention signal, d) to decide whether an erasable may be presumed good, e) to determine if there were programs in process and whether they were designed to be restartable.

If the results of this processing are successful, the Restart Tables and Restarts Routine is called in to set up the EXECUTIVE and the Waitlist to reestablish, as if uninterrupted, the tasks at hand when the restart happened.

3. Problems of Restart Protection

The method used by the programmer to obtain restart protection may be of interest. Basically, he designs a region of program such that the program may be re-entered at the beginning, (the restart point). When a programming action is required which renders the restart point invalid, a new restart point is defined and the Phase pointer, which defines the return point, is advanced. The Phase Table Maintenance routines are used for this purpose.

This process is complicated by the fact that it often occurs that several programs must be in process at the same time without synchronization one to the other. The problem of asynchronous program operation is solved by defining several Groups. Within a Group, the restart phase is updated as restart points are passed, and these operations are independent of the restart Phase of programs assigned to another Group.

Another significant problem arises from the need for precise timing of events without regard to the time spent reprocessing due to restarts. This is handled by

reading the AGC clock at the time a restart point is defined and again after a restart occurs.

Several other difficulties arise in development of sound restart protection, and it is certain that the time to generate restart protected coding is significant in terms of the development cycle for flight programs.

4. Diagnostic Information Available

Flight 206 program Sunburst will save the address contained in the Z register at the time that the malfunction was detected and the contents of the BBANK register, in consecutive registers RSBB&Q and RSBB&Q +1.

$C(\text{BBANK}) \rightarrow \text{RSBB\&Q}$

$C(\text{ Z }) \rightarrow \text{RSBB\&Q} + 1$

In addition, for restarts generated via the CSHOLE and ABORT routines, the 2CADR of the location following the transfer to these routines is saved in ALMCADR and ALMCADR +1.

These processes are expected to be available on future programs.

5. Memory Assigned to Restart Protection

These figures are extracted from Flight 206: The Restart Tables and Restarts Routine occupies 450 words. Phase Table Maintenance occupies 160 words. The fresh start and restarts routine would be about 150 words shorter without restarts. In addition, the defining of restart points within the program requires 450 words. There is an additional harder-to-define word penalty associated with designing a program under the constraint of restart protection.

6. Typical Program Flow during Restart

This section covers the processing that occurs from the time a "GO" sequence is initiated (involuntary transfer to location 4000_g), until the logic has decided that the control can be returned to the programs in process. If a successful restart can not be accomplished, a fresh start, or an AGC idle state is entered. The flow included in this section is for information purposes only. It should be recalled that the design of the program involves shared coding with other processes besides restart. Therefore some logical processes indicated in the flow chart are not necessary, although they are compatible with the restart logic.

Restart (location 4000_g)

- 1) increment redoctr
- 2) C(BBANK&Q) → RSBB&Q, RSBB&Q +1
(address of next location at time of restart)
- 3) zero → channels 5, 6, 12
- 4) set Time3, Time4, Time5 to overflow
shortly, but not synchronously
- 5) zero → channels 7, 11, 14
- 6) enable interrupts, by placing 1's in bits 12, 13, 14 of channel 13
- 7) clear and initialize Waitlist, EXECUTIVE
- 8) turn off DSKY displays
(not the indicator panel)
- 9) clear inlink (45 register)
- 10) initialize pinball
- 11) initialize stalling routines
- 12) initialize gyro torquing routines
- 13) initialize radar sampling routines
- 14) set T4Rupt to ignore PIPA, or telemetry failures on first
recovering from restart
- 15) initialize radar portion of T4Rupt
- 16) initialize self check
- 17) set up nominal downlink list

18) test for oscillator failure

no
AGC warning light on?
yes → fresh start
(assume multiple restarts)

19) are mark reject and error light reset buttons depressed?

no → yes fresh start
(pass key when lock out occurs)

20) was self check testing erasable when restart occurred?

yes → fresh start
(erasable is probably bad)

21) blank DSKY displays associated with DSPTAB + 11D except: program alarm
gimbal lock warning
no attitude

22) NO ATTITUDE LIGHT was on leave ISS in course align

23) leave IMU failure inhibits intact, reset all failure codes

24) set T5Rupt to do DAPIDLER

25) Display ABORT code

26) check to see if LMP command was in out channel. If so, resend command.

27) turn throttle counter on. (if empty no harm done)

check the restartability flag
(is the program now in process restartable?)
(this flag is zero for intentionally generated software restarts)

flag is on

flag is off

Fakestrt

Display alarm 0316
(restart but program not restartable)

turn engine off
(it might be on)

close down servicer, delta-v monitor etc.,
to preserve state vector properly. (In 206,
some mission phases will not be restart
protected due to schedule constraints. This
section preserves the state vector and leaves
the LGC ready for uptelemetry.)

set DAP to maximum deadband attitude
hold

make all restart groups inactive except
servicer and state vector copy

set timer/phase pairs to idle state

goprogram 2

compare phase (restart) table with complemented phase table. Is
erasable "safe"?

yes

no

display alarm 1107

fresh start

check groups for active items

active items

proceed to restarts routine,
to process phase information
and to make proper Waitlist
and EXECUTIVE calls to put
proper programs on the air.
This is the normal path for
successful restart.

no active items

display alarm 1110
(restart with no active groups)

is the restartability flag set?

yes

display major mode, start dummyjob.
This is the path obtained when no pro-
grams are active.

no

this path occurs if a non restart
protected mission phase suffers
restart. Program reverts to
idling mode just as if mission
phase was successfully completed.

4.3 PGNCS Failure Monitor

The PGNCS performance and operational readiness are self-monitored and caution and warning information are displayed to the crew. Two warning (red) lamps are actuated by the PGNCS on the Caution/Warning Panel: LGC Warning indicates computer failure; INERTIAL REFERENCE warning indicates failure of the inertial subsystem. Also a PGNCS Caution (amber) lamp is actuated to indicate non-critical problems in the system. Further detail regarding the caution items is displayed by means of the DSKY event lamps and the DSKY data registers (in the event of a program alarm.) The failure monitor mechanization is shown in Figure 4-2.

4.3.1 LGC Warning.

An LGC warning alarm is generated in the event of LGC power failure, scaler failure of either of two types, restart or counter fail during LGC operate, or in response to an alarm test program. As is shown in Figure 4-2 a scaler fail or prime power fail result in immediate alarm indication whereas the other inputs are buffered by a filter so as to prevent momentary transient disturbances which recover from causing a warning alarm. In this subsection the various inputs and conditions associated with LGC warning are defined.

- (a) SCAFAL - Occurs if scaler stage 17 (1.28 sec period) fails to produce pulses. This provides a check on the timing for all logic alarms.
- (b) COUNTER FAIL - Occurs if counter increments happen too frequently or else fail to happen following an increment request. "Too frequently" means continuous counter requests and/or incrementing for from .625 ms to 1.875 ms.
- (c) SCADBL - Occurs if the 100 pps scaler stage operates at a pulse rate of 200 pps or more.
- (d) PARITY FAIL - Occurs if any accessed word in fixed or erasable memory whose address is octal 10 or greater contains an even number of "ones".
- (e) RUPT LOCK - Occurs if interrupt is either too long or too infrequent. The criterion for "too long" is phase dependent varying from 140 ms to 300 ms. Likewise the criterion for "too infrequent" varies from 140 ms to 300 ms.

- (f) **TC TRAP** - Occurs if too many consecutive TC or TCF instructions are run or TCF instructions are too infrequent. The criterion for "too many" varies from 5 ms to 15 ms duration. The criterion for "too infrequent" varies from 5 ms to 15 ms absence.
- (g) **NIGHT WATCHMAN**- Occurs if the computer should fail to access address 67 within a period whose duration varies from .64 sec to 1.02 sec.
- (h) **V FAIL** - Occurs if the LGC voltages (28, 14, 4) are out of limits. This signal produces STRT1 if it stays on for a period of between 157 and 470 μ sec. If the computer is in the STANDBY mode, an input to the LGC WARNING FILTER is generated simultaneous with STRT1. The following criteria apply for V FAIL:
- | | |
|--------------------|----------------------|
| 4 V Supply > 4.4V | 14 V Supply > 16V |
| 4 V Supply < 3.65V | 14 V Supply < 12.5V |
| | 28 V Supply < ~22.6V |
- (i) **STANDBY** - This is a signal which turns on RESTART and turns off the switchable +4 and +14 voltage, thus putting the LGC into a low power mode where only the scaler, timing signal, and a few auxiliary signals are operative. STANDBY is initiated by first setting the ENABLE STANDBY outbit (CH13 B11), and then pressing the STANDBY button on the DSKY for a time which varies from .64 sec to 1.02 sec, at the end of which time the STANDBY LIGHT is turned on. (All LGC alarms are inhibited during the standby mode with the exception of LGC WARNING, which can be caused by VOLTAGE FAIL or SCALER FAIL; and TEMPERATURE CAUTION, which can be caused by TEMP ALARM.) Normal operation is resumed by pressing the STANDBY button on the DSKY again, time of depression same as above.
- (j) **RESTART** - RESTART occurs at next time 12 following occurrence of any one or more of the parameters shown in Figure 4-2 except OSCILLATOR FAIL.

RESTART occurs immediately and forces time counter to 12 upon occurrence of OSCILLATOR FAIL. (See paragraph "1" below.)

RESTART causes the computer to transfer control to address 4000 as soon as it disappears. It sets a flip-flop which lights the RESTART CAUTION lamp in the DSKY.

The Flip-flop is reset either by the ALARM RESET hard wired signal or by the CAUTION RESET output CH11 B10. ALARM TEST operates the lamp but not the flip-flop.

- (k) **WARNING FILTER** - This circuit is used to operate the LGC WARNING output following repeated or prolonged occurrences of the parameters shown in Figure 4-2. All occurrences of these signals are stretched so that no more than one input to the filter is generated in each 160 millisecond period. Approximately six consecutive stretched pulses cause LGC WARNING to turn on for about 5 seconds. Non-consecutive stretched pulses may also cause LGC WARNING after an interval dependent on the frequency of the pulses. The output will not occur if input pulses occur at a frequency of less than 0.9 pps; and the output will remain on if pulses occur at a frequency of 0.6 pps or more. The threshold of the filter resumes its normal level with a time constant of many seconds after the filter has received inputs. An immediate reset of the LGC FAIL due to a WARNING FILTER output is therefore not possible.
- (l) **OSCILLATOR FAIL** - Occurs if the oscillator stops. Has nominal 250-millisecond delay to keep signal present after the oscillator starts. Also occurs when LGC is in STANDBY because of loss of power to front end of circuit. This gives 250-millisecond delay in starting when LGC comes out of STANDBY into OPERATE. Causes immediate restart without waiting for time pulse 12.

4.3.2 Inertial Reference Warning

The Inertial Reference Warning signal is the logical "OR" of the following parameters, any one of which will cause an Inertial Reference Warning under the following conditions:

(a) IMU Fail

- (1) IG Servo Error - greater than 2.9 mr for 2 sec
- (2) MG Servo Error - greater than 2.9 mr for 2 sec
- (3) OG Servo Error - greater than 2.9 mr for 2 sec
- (4) 3200 cps - decrease to 50% of normal level
- (5) 800 wheel supply - decrease to 50% of normal level

These parameters are generated in the Inertial Subsystem. However, the "FAIL" signal itself is under LGC program control. It is ignored by the LGC program when the G&N System is in the

Coarse Align Mode and during the 5 second interval following Coarse Align. During this mode the servo errors normally exceed the above criteria.

(b) PIPA FAIL

Occurs if no pulses arrive from a PIPA during a 312.5μsec period, or else if both plus and minus pulses occur, or if a "long time" elapses without at least one plus pulse and at least one minus pulse arriving. By "long time" is meant a period of between 1.28 sec and 3.84 sec.

This FAIL signal is generated totally within the LGC and thus is completely under LGC program control. Its generation is enabled by the LGC only during LGC controlled translation or thrusting maneuvers.

(c) ISS CDU FAIL (Monitored for each of 3 CDU's)

- (1) CDU fine error - in excess of 1.0V rms
- (2) CDU coarse error - in excess of 2.5V rms
- (3) READ COUNTER limit cycle - in excess of 160 cps
- (4) $\cos(\theta - \phi)$ - below 2.0V
- (5) +14 DC Supply - decrease to 50% of normal level

These parameters are generated in the Inertial Sub System. However, the "FAIL" signal itself is under LGC program control. It is ignored by the LGC program when the G&N System is in the CDU Zero Mode. During this Mode the CDU errors normally exceed the above criteria.

4.3.3 PGNCs Caution

The PGNCs Caution lamp is actuated by the following undesirable but non-critical events:

- (a) LGC Restart during Operation. In the event of Restart during operate a latch is set in the LGC which maintains the PGNCs Caution alarm and the RESTART lamp on the DSKY until the latch is reset by program or until the latch is manually reset by ALARM RESET. For further detail see section 4.3.4
- (b) Temperature out of Limits. The LGC receives a signal from the IMU when the stable member temperature is in the range 126.3°F to 134.3°F. In the absence of this signal, the Caution alarm and the TEMP lamp on the DSKY are actuated.

- (c) Gimbal Lock. When the LGC determines that the middle gimbal angle (MGA) of the IMU is greater than 70° , the Caution alarm and the Gimbal Lock lamp on the DSKY are actuated. When MGA exceeds 85° the ISS is downmoded to Coarse Align and the No Attitude lamp on the DSKY is actuated.
- (d) Program Alarm. Under a variety of situations a program alarm is generated. Figure 4-2 illustrates one example, that of PIPA fail when the vehicle is not in a thrusting mode. Under program control the LGC inhibits this alarm for 10 sec after system turn-on. The program alarm actuates the Caution alarm and the Program lamp on the DSKY. For further information see section 4.3.4.
- (e) Tracker Alarm. When the Rendezvous Radar or Landing Radar are in use the Caution alarm and Tracker lamp on the DSKY will be energized by any of the following three occurrences:
- (1) RDR CDU Fail (Monitored for both CDU's) defined by same failure modes with same constraints as IMU CDU Fail [see section 4.3.2 (c)]. This fail signal is generated in the CDU and, under program control, generates a Caution alarm only when the PGNCs is in the RDR Auto mode.
 - (2) RR Fail Defined by presence of the Rendezvous Radar Data Good Bar discrete during a data read sequence by the LGC or inability of LGC to successfully get coherent data.
 - (3) Landing Radar Altitude Data Good Bar or Velocity Data Good Bar when the PGNCs is in the LRDR enable mode and when the LGC is attempting to get data from the LRDR or anytime when the LGC is attempting unsuccessfully to get data, the Caution alarm and Tracker lamp on the DSKY are actuated.

4.3.4 Restart and Program Alarms

When the Restart or Program Alarm lamps are illuminated on the DSKY, either V 05 N 31 will automatically appear on the DSKY with a key to the source of the alarm displayed in R1 or by astronaut call up from the DSKY this information can be made available. This allows the astronaut to identify and normally correct the alarm condition. The listing of program alarms is included in Section 4.4.

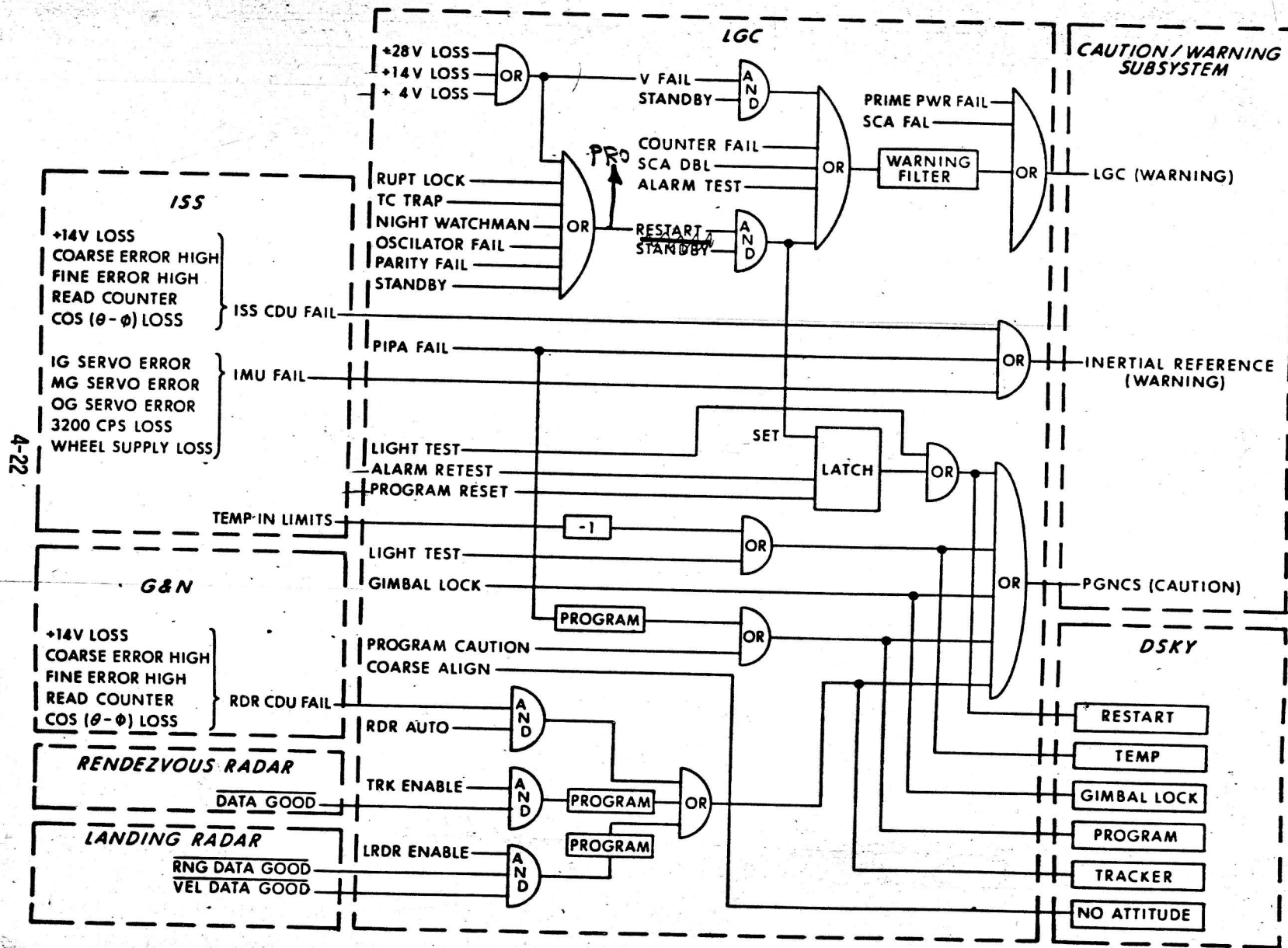


Fig. 4-2 PGNCS Failure Monitor

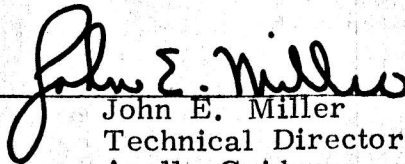
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
Instrumentation Laboratory
Cambridge, Massachusetts

.APOLLO PROJECT MEMO #1626

TO: Distribution
FROM: John E. Miller
DATE: 3 January 1966
SUBJECT: Criteria for Acceptance of CGC and LGC Restarts

An AGC is not considered to be flight-worthy after an inadvertent restart unless:

1. The cause of restart is explained and found acceptable; or
2. Evidence can be compiled that the restart does not recur under the same circumstances in a reasonable number of tries. In most instances, ten tries may be considered reasonable. In the case of unexplained restarts which do not repeat, where the investigation has proven the environment (EMI, power, thermal, etc.) is within specification, two restarts are considered reasonable in a period of running time not exceeding 1000 hours.


John E. Miller

Technical Director, G&N System
Apollo Guidance and Navigation

JEM:sh

cc: R. Ragan
D. Hoag
L. Larson
E. Hall
L. Wilk
A. Laats
R. Crisp
J. Flanders
W. Rhine/NASA
S. Snipes/NASA
R. Lewis/NASA

Instrumentation Laboratory
Massachusetts Institute of Technology
Cambridge, Massachusetts

Apollo Project Memo #1665

To: John Miller
From: Albert Hopkins
Date: 13 February 1967
Subj: Explanation of AGC Alarm Hardware Organization

This memo is written in response to several requests to document the effect of restarts in Apollo Guidance Computers. This subject has two aspects which lend themselves to separate discussion: the alarm hardware of the computer and the AGC mission program's response to a restart. Following is a discussion of the hardware.

1. Need for Restart

A digital computer is only useful when it is executing a coherent sequence of instructions which has been properly initiated. Any transient error may affect the execution of this sequence of instructions so as to produce wrong data, or to cause access of wrong storage locations in memory thus invalidating data, or to modify the instruction sequence, or all of these. In the AGC, nearly a million actions (word transfers) are performed in a second. A random error rate of one in 10^{12} actions would be considered good in today's technology, but at that rate would occur once in 300 hours in an AGC.

Added to the random error rate are numerous modes of induced errors, including anomalous power and interface signals, intermittent computer defects, program/bugs, and operator errors. The incidence of induced errors has so far been very much greater than that of random errors (if indeed there have been any) in AGC experience. The possibility of error presents a requirement for a means of recovery, for the computer itself would still be perfectly useful given that it could be re-initiated into the program. To meet the need of re-initiation capability the AGC is equipped with a restart feature comprising alarms to detect malfunction and a standard initiation sequence which overrides all others following a detected malfunction. If the malfunction is of a transient nature it should thus be possible for the AGC to rerun the sequence in which the malfunction occurred from a rerun point planned in advance. If the malfunction is permanent, a recurring alarm condition would result from attempts to restart.

2. AGC Alarm Logic

The AGC contains a number of checking features which will detect malfunctions of hardware and software, random and induced. During subsystem and system checkout as well as during the mission itself it is important to detect and illuminate all possible sources of computer malfunction whether internal, external, or program dependent. The computer alarms and the DSKY display provide a sensitive detector of such malfunctions, but it is often a major engineering task to sort out the cause from the multiplicity of possible sources.

Much of the AGC is checked by a program in its memory called Selfcheck, which periodically checks its ability to execute instructions, and store data. This program, employing relatively little hardware, checks about two thirds of the computer. The other one third consists of input-output circuitry which is not amenable to testing except by integrated system testing.

The AGC alarm logic continuously monitors numerous voltages and signals vital to computer and system operation. A restart is called for if an instruction sequencing or memory transfer malfunction is detected. Because many program attributes were known at the time of hardware design it was possible to put certain constraints on program sequencing. One such constraint is that there must be periodic transfers of control from one program to another, neither too frequent nor too seldom. The TC Trap alarm reacts to violations of this constraint. Another constraint is that program interrupt must occur neither too seldom nor for too long a time; the rupt lock alarm monitors this. Another alarm, the night watchman, requires that a certain erasable memory address be accessed with sufficient frequency. Any of these three alarms or memory parity failure will cause a restart. Since these alarms depend upon such things as the scaler and the power supplies, these latter circuits are separately checked and provide failure indication. A design prediction so far borne out in practice is that nearly all program sequencing malfunctions are detectable by this group of alarms.

A degree of freedom available to the programmer is the ability to force a restart alarm by a programmed TC Trap if any violations are detected by the program of program constraints. Examples of its use are for operands not in a permissible domain, and for too many executive jobs or waitlist tasks outstanding.

Finally, a restart can be induced, if desired, by the operator, although not conveniently. If the keyboard and display are operative an alarm constraint can be violated (e.g., display a non-existent word in fixed memory, causing parity fail). In any case, temporarily removing power either by entering the standby mode and resuming normal operation or via the main breakers will cause a restart.

Restart is a caution condition, and causes a lamp to be illuminated until reset by the DSKY reset key. Repeated restarts, or any of certain other alarm conditions will generate a warning signal indicating an unusable AGC. If the condition disappears the warning signal is cancelled after a few seconds. The reset key has no effect on the warning signal.

3. Fresh Start Vs. Restart

Strictly speaking, the hardware discussion ends here; but it is appropriate to venture somewhat beyond the interface with the software to identify an important distinction in terminology.

As a general policy, it is desirable to inhibit continuously repetitive restarts in the presence of a permanent malfunction (it is worthy of note that an imperfectly written or incorrectly initialized program can exhibit behavior of the same nature). A restart resets a number of signals in the computer and transfers control to a sequence whose initial address is hard wired in the circuitry. In a "restart protected" program the erasable memory is consulted to determine what the activity status of the computer was prior to the restart. If this information is self consistent the various activities are resumed at appropriate starting points. Inhibition of this program-directed activity is itself a program function. A program generally known as "fresh-start" initializes numerous erasable memory locations to produce a quiescent computer status and terminate all activities other than the idling "dummy job". It is the fresh-start which is invoked to quell a repetitive restart condition.

In most programs in a normally functioning AGC the fresh-start is executed by a verb 36 command. If the display and keyboard are locked out by high restart frequency, a fresh start can be made in many mission programs by simultaneous depression of certain DSKY and mark buttons. Here the restart program looks for these button depressions prior to its normal restart attempt. If it finds them

depressed it branches to fresh-start. It is also possible for the program to detect whether a computer warning condition exists at the time of restart and branch to fresh-start on that basis, though at this writing it is not known whether this will be used.

In conclusion, a restart will attempt to resume all activities if the program is so written, unless a branch is taken to fresh-start which forces the computer to idle. Just what will happen under specific circumstances is a function of the software, and may vary from one mission to another.

Dist.

E.C. Hall
J.S. Miller
J. Nevins
A. Kosmala
D. Hoag
A. Harano

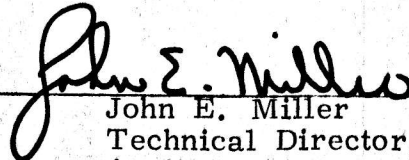
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
Instrumentation Laboratory
Cambridge, Massachusetts

APOLLO PROJECT MEMO #1626

TO: Distribution
FROM: John E. Miller
DATE: 3 January 1966
SUBJECT: Criteria for Acceptance of CGC and LGC Restarts

An AGC is not considered to be flight-worthy after an inadvertent restart unless:

1. The cause of restart is explained and found acceptable; or
2. Evidence can be compiled that the restart does not recur under the same circumstances in a reasonable number of tries. In most instances, ten tries may be considered reasonable. In the case of unexplained restarts which do not repeat, where the investigation has proven the environment (EMI, power, thermal, etc.) is within specification, two restarts are considered reasonable in a period of running time not exceeding 1000 hours.



John E. Miller
Technical Director, G&N System
Apollo Guidance and Navigation

JEM:sh

cc: R. Ragan
D. Hoag
L. Larson
E. Hall
L. Wilk
A. Laats
R. Crisp
J. Flanders
W. Rhine/NASA
S. Snipes/NASA
R. Lewis/NASA

I recommend that corrected Sundial modules be made for use in critical system testing. Only the B3 module needs to be changed. Gyrocompass is important in its exercise of major system functions, and it is essential that an error-free program is available to A.C. Electronics for use in the EMI qualification of the Apollo CG&N system.



Robert Crisp

RC:jdn