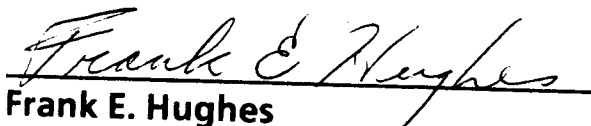


Data Processing System Principles and Interfaces

DPS PRINC 2102



M. L. Richmond
Head, DPS and Navigation Section



Frank E. Hughes
Chief, Flight Training Branch

November 1987

**Mission Operations Directorate
Training Division
Flight Training Branch**

M. D. Erminger/S. L. O'Roark

Preface

The objective of this workbook is to teach non-data processing system (DPS) flight controllers and non-DPS instructors about how DPS affects their system. This workbook has an operational flavor organized by topics, or buzzwords, used over the flight loop. The topics are organized alphabetically for easy reference. Each topic area will explain what is going on, why it is going on, and how it affects other systems. This workbook fills a gap somewhere between workbooks, single systems trainers, and the osmosis learning of DPS through integrated simulations. This workbook will not go in to excruciating detail but will explain what is going on in the DPS world and how it affects other systems.

The following DPS lessons are recommended before reading this workbook:

- 2 CSI 2102
- CSI 2105
- DPS OV 2102
- DPS 2105
- DPS HW/SW 2102

Section 1: Backup Flight System and the Payload Buses

During ascent and entry, there is not a systems management (SM) computer running. During these flight phases, the backup flight system (BFS) performs many of the SM and special processes functions.

Following is a list of some of the functions that the BFS performs during ascent and entry:

- Backup caution and warning (C&W) and SM alerts
- Freon power pump management
- Flash evaporator/ammonia boiler activation/deactivation
- Auxiliary power unit (APU) fuel quantity gauging
- Hydraulic water boiler quantity gauging
- Fuel cells total current and power calculation
- S-band antenna management
- Cabin pressure monitor.

The MODE switch for the BFS general purpose computer (GPC) on panel O6 determines whether or not the BFS has command of the payload data buses (figure 1-1). The BFS is up and running when the MODE switch is in STBY or RUN. The only difference between standby and run is that when the BFS is in RUN, it commands the payload data buses. The BFS is in RUN during ascent and entry, since it needs to command the payload data buses to perform the above functions.

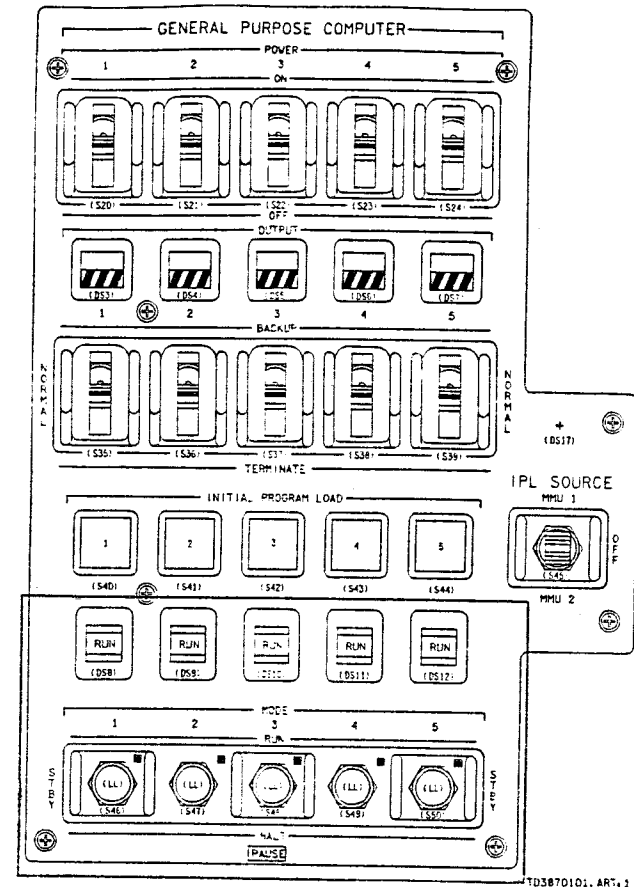


Figure 1-1.- GPC MODE switch.

It is possible for the BFS GPC MODE switch to fail to the STBY position. The crew can determine this from the BFS GNC SYS SUMM 1 display; there will be down arrows for both PL MDMS but no annunciation of I/O ERROR PL1 (2). The DPS flight controllers can also detect this failure on the ground and will notify the crew.

If the crew engages the BFS when it is in standby, it will take command of the payload data buses along with the flight-critical data buses.

Some examples of what happens during ascent and entry when the BFS is in standby follow:

- Class 2 and 3 alarm light and tone annunciations are inhibited.
- Reconfiguration of Freon pump B during a fuel cell failure cannot take place.
- Automatic activation/deactivation of the flash evaporators and ammonia boilers to provide cooling for the Freon loops cannot take place.
- APU fuel quantity gauging or APU FUEL/H₂O QTY meters on panel F8 are not supported.
- S-band antenna selection cannot be changed.
- The backup flight controller (BFC) lights on the eyebrow panels will not flash if the BFS goes disengaged standalone.

When the BFS releases the payload data buses during ascent or entry, the primary avionics software system (PASS) gains command because it is assigned the payload data buses in the memory configuration 1 and 3 nominal bus assignment table (NBAT) (see section 13). Although PASS has command of the payload data buses at this point, it does not attempt to do anything on them, but this gives the instrumentation and communications officer (INCO) the capability to uplink commands to PASS, which are then sent out over the payload data buses. These commands could change antennas, activate ammonia boilers, deactivate flash evaporators, etc.

Onorbit, the SM GPC commands the payload data buses. If the crew powers on the BFS onorbit and takes the BFS to RUN, the SM GPC will automatically relinquish command of the payload data buses to the BFS.

Section 2: Backup Flight System Engage

When to Engage

The BFS GPC is used as a backup to the PASS. Flight rules require the crew to engage the BFS on ascent for loss of redundant set, loss of PASS capability to control the vehicle due to GPC, multiplexer/demultiplexer (MDM), and line replaceable unit (LRU) failures, loss of control due to divergent trajectory, and roll maneuver not initiated as scheduled.

During entry, the BFS engage requirements are more simply stated as engage for loss of redundant set or loss of control.

A redundant set failure during ascent or entry is defined as all PASS GPCs halt or quit, or the redundant set breaks up to all simplex (each operating independently of the others). In the case of all PASS GPCs halting or quitting, no software is being executed, and it is fairly easy to understand why the crew should engage. The redundant set breakup is also called a four-way set split (see redundant set split in section 7). A BFS engage would cause all four GPCs to go to halt and the BFS would take over control of the vehicle.

The next scenario that requires a BFS engage on ascent or entry is loss of control. This flight rule is a little vague but basically leaves it up to the commander to decide if the vehicle is still in control. One caveat to this flight rule for engaging the BFS is that the BFS must be able to pick up two strings out of strings 1, 2, and 3 in order to pick up sufficient accelerometer assemblies (AAs), rate gyro assemblies (RGAs), and flight control system (FCS) channels to maintain vehicle control.

Another scenario on ascent that requires a BFS engage is loss of control due to divergent trajectory. During first stage, loss of control may occur very quickly due to control authority of the solid rocket boosters (SRBs). If this occurs before 1 minute 30 seconds into the flight, the crew should engage the BFS since it is believed that they cannot manually control the vehicle in this region of maximum dynamic pressure. After 1 minute 30 seconds, the crew can attempt to manually control the vehicle, and, if that does not work, they can then engage the BFS.

The final reason for engaging the BFS on ascent is that the roll maneuver is not initiated. The crew must engage in this case because the onboard displays are not adequate for the crew to manually execute the roll/pitch profile through the high Q phase with the PASS.

No-Go for BFS Engage

There are four different reasons for not engaging the BFS: cathode-ray tube (CRT) problems, BFS standalone, power problems, and data path problems.

The flight rules state that the BFS is no-go for engagement if no forward CRT interface exists. In other words, the BFS cannot drive any of the three forward CRTs. This would inhibit certain keyboard inputs necessary to give the BFS information which would help it fly the vehicle.

The second reason for not engaging the BFS is if the BFS has been in standalone mode (will not track PASS) for a specified amount of time. In major modes (MMs) 102, 103, and 601, the time limit is 10 seconds. In MMs 304, 305, 602, and 603, the time limit is 45 seconds. The reason for this flight rule is that the BFS propagates its vehicle state vector forward from the point that it went standalone. This propagation is made without any sensor inputs. After the specified period of time, the BFS state vector is not thought to be good. These numbers are ball park estimates and depend on the circumstances. It is possible that the BFS state vector would be bad on entry after only 10 seconds of standalone if the standalone occurred during a roll reversal.

There are several power problems that can cause the BFS to be no-go for engage. A BFS power failure would be a good reason, but control bus CA1 and essential bus 3AB also cause problems. In the case of control bus CA1, the BFS (assuming it is in GPC 5) does not see the engage request. Essential 3AB is a problem because it effectively does not allow the GPCs to see which GPC is designated as BFS, and PASS and BFS will ignore an engage request. The cue cards for both of these bus failures tell the crew not to engage BFS.

The final reason for not engaging the BFS could be data path problems. There is a cue card used for ascent and entry called multiple data path loss (figure 2-1). In addition to other actions called out by this cue card, the card tells the crew when the BFS is no-go.

There are two cases that this cue card covers for no engagement which involve AAs and RGAs. Basically, the BFS always midvalue selects slots 1, 2, and 3 for AAs and RGAs. If two of the first three LRU slots are lost, the BFS either will read an input of 0 or will use the last-sensed value if commfaulted. It is possible for the crew to do a fourth LRU substitution to put RGA 4 or AA 4 into one of the first three slots if one of the other LRUs has failed. The multiple data path loss cue card keeps track of failures that affect the AAs and RGAs. There is a note at the bottom of the cue card that is always confusing (see figure 2-1). This note simply states that if one of the failures was caused by a GPC problem, an engaged BFS would pick up that string and the associated equipment. Although not directly stated on the card, FCS channel feedbacks are RM'd in the same manner as the AAs and RGAs, and so two of the first three feedbacks must be good and available to the BFS in order for it to be engaged. Fourth LRU substitutions can also be for FCS feedbacks.

MULTIPLE DATA PATH LOSS (Non-Recov) ASCENT
 If two FA MDMs lost during OMS burn, man shutdn of
 one or both OMS ENG reqd

FAILURE	TYPE	ACTIONS
GPC/FF1&2	B	Use R BODY FLAP & TRIM ENA SWs MM102, RTLS, TAL: BFS NO-GO (AAs) (Note)
	I/O FF2 I/O & FF1 B	RTLS, TAL INH ADTA to G&C FLY THETA LIMITS (ADTA 1,2,3 &)
GPC/FF1&3	B or I/O	MM102, RTLS, TAL: BFS NO-GO (AAs) (Note)
GPC/FF2&3	B or I/O	MM102, RTLS, TAL: BFS NO-GO (AAs) (Note)
GPC/FF3&4	B or I/O	Use L BODY FLAP & TRIM ENA SWs RTLS, TAL
	FF3 I/O & FF4 B	INH ADTA to G&C FLY THETA LIMITS (ADTA 2,3,4 &)
GPC/FA1&2	D or I/O	BFS NO-GO (RGAs) (Note) SRB SEP: BFS OPS 103 PRO
	I/O	ET SEP: MPS FDLN RLF ISOL (two) - OP Preburn: L OMS ENG - OFF (TVC) Pre MPS dump: MPS PREVLVs (six) - CL OP
GPC/FA1&3	D or I/O	BFS NO-GO (RGAs) (Note) SRB SEP: BFS OPS 103 PRO
	I/O	ET SEP: MPS FDLN RLF ISOL (two) - OP Preburn: R OMS ENG - OFF (IGN) Pre MPS dump: MPS PREVLVs (six) - CL OP
GPC/FA1&4	I/O	ET SEP: MPS FDLN RLF ISOL (two) - OP Preburn: L OMS ENG - OFF (IGN) Pre MPS dump: MPS PREVLVs (six) - CL OP
GPC/FA2&3	D or I/O	BFS NO-GO (RGAs) (Note) Post MECO: Expect 'SEP INH' (Feedline Disc fail)
	I/O	ET SEP: MPS FDLN RLF ISOL (two) - OP Preburn: L OMS ENG - OFF (IGN) Pre MPS dump: MPS PREVLVs (six) - CL OP
GPC/FA2&4	D or I/O	SRB SEP: BFS OPS 103 PRO Post MECO: Expect 'SEP INH' (Feedline Disc fail)
	I/O	Preburn: R OMS ENG - OFF (IGN) Pre MPS dump: MPS PREVLVs (six) - CL OP
GPC/FA3&4	D or I/O	SRB SEP: BFS OPS 103 PRO Post MECO: Expect 'SEP INH' (Feedline Disc fail)
	I/O	ET SEP: MPS FDLN RLF ISOL (two) - OP Preburn: R OMS ENG - OFF (TVC) Pre MPS dump: MPS PREVLVs (six) - CL OP

NOTE
 If data path loss not due to MDM fail, BFS is
 GO and will pick up MDMs when engaged

MULTIPLE DATA PATH LOSS (Non-Recov) ENTRY
 Assumes completion of 'FF(FA) MDM I/O ERROR' or
 'BCE BYPASS'

FAILURE	TYPE	ACTIONS
GPC/FF1&2	B	Use R BODY FLAP & TRIM ENA SWs BFS NO-GO (AAs) (Note)
	I/O FF2 I/O & FF1 B	INH ADTA TO G&C FLY THETA LIMITS (ADTA 1,2,3 &)
GPC/FF1&3	B or I/O	BFS NO-GO (AAs) (Note)
GPC/FF2&3	B or I/O	BFS NO-GO (AAs) (Note)
GPC/FF3&4	B	Use L BODY FLAP & TRIM ENA SWs RTLS, TAL
	I/O FF3 I/O & FF4 B	INH ADTA TO G&C FLY THETA LIMITS ADTA 2,3,4 &)
GPC/FA1&2 or GPC/FA1&3 or GPC/FA2&3	D or I/O	BFS NO-GO (RGAs) (Note)

NOTE
 If data path loss not due to MDM fail, BFS
 is GO and will pick up MDMs when engaged

Figure 2-1.- Multiple data path loss cue cards.

Losses at BFS Engage

There are two reasons that the vehicle loses capabilities after a BFS engage. The first reason is one over which there is no control. The BFS just has limited capabilities compared to the PASS. The second reason might be due to failures at the BFS that did not affect PASS before the BFS was engaged.

All of the ascent, entry, and abort software is resident in the BFS all of the time, and, as a result, the BFS does not have room for all of the capabilities that PASS does. This workbook does not intend to discuss all of the differences between PASS and BFS, but here is a synopsis of those differences.

∞ BFS uses a simple redundancy management (RM) system and has limited onorbit capabilities. BFS has reduced guidance, navigation, and control (GN&C) capabilities, such as no preflight or onorbit inertial measurement unit (IMU) alignment capability, no microwave landing system (MLS) data processing, and no autoland capability. BFS does not have contingency abort capability. BFS also has reduced SM and special processes capabilities.

The second reason that the vehicle might lose capability at BFS engage is that there may be a problem with the BFS. For example, there might be a flight-critical data bus down at the BFS GPC. As an example of this, assume that flight critical 1 (FC 1) is failed at the BFS GPC pre-engaged. The PASS computers will operate just fine with this problem at the BFS, but the BFS will bypass string 1 thinking that it is a problem with the PASS GPC commanding string 1. When the crew engages the BFS, the BFS will pick up string 1 but will bypass flight forward 1 (FF 1). This is only one example of what might be lost at BFS engage. Other bus failures at the BFS will cause similar problems after an engage. BFS FC bus transmitter problems are hidden prior to the engage, since BFS does not use the transmitters until engaged.

Recoveries After BFS Engage

At engage, the BFS performs engage initialization. The BFS takes control of the vehicle by commanding the flight-critical data buses. PASS GPC failures can cause PASS to lose command capabilities that the BFS will pick up after an engage. For example, loss of GPC 1, and therefore string 1 (which has MDM FF 1 and MDM flight aft 1 (FA 1)), might occur. A BFS engage would recover string 1 since the failure was caused by a PASS GPC failure.

Another failure might be FC 1 at GPC 1. In this case, PASS drops command of MDM FF 1, annunciating an I/O ERROR FF 1 message. The BFS then stops listening to string 1 and annunciates a BCE STRG 1 PASS message. The BFS knows that FF 1 went down because of a PASS GPC problem and does not trust that GPC. It therefore bypasses the whole string. On a BFS engage, the BFS picks up string 1 as it did in the GPC 1 failure case. In this case, IMU 1 and AA 1 were lost when PASS dropped FF 1 and were recovered after a BFS engage.

Section 3: Display Electronics Unit Equivalent Uplink

A display electronics unit (DEU) equivalent uplink is a capability that the DPS flight controllers have to uplink keystrokes to the GPCs. The keystrokes appear to the GPCs just as if a crewmember had entered them on a keyboard.

The DEU equivalent is used for critical tasks like transitioning from OPS 9 to OPS 1 before lift-off, and for mundane tasks such as changing the variable parameters on SPEC 1 (see section 17).

Section 4: Downlist

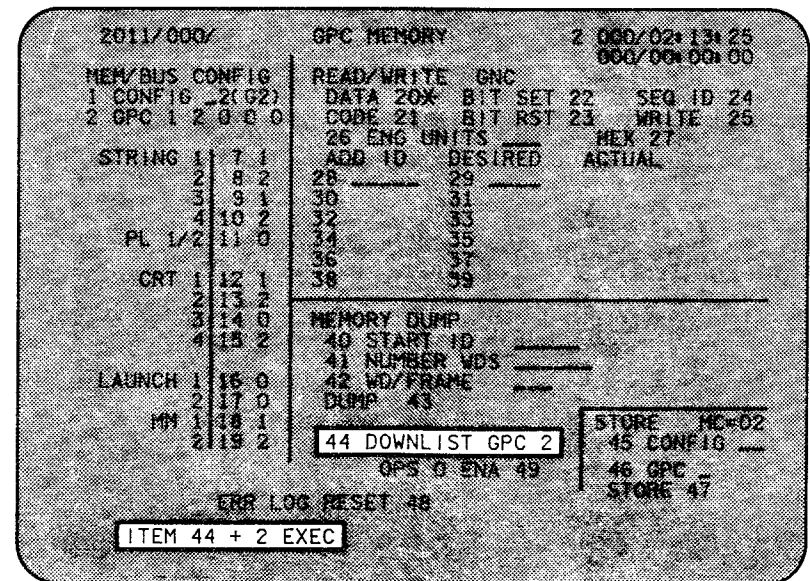
Downlist is what we call the telemetry that comes from the Orbiter GPCs and ends up on displays in the flight control room. It is just a part of the total downlink from the vehicle, which also includes (to name a few) operation instrumentation (OI) data, payload (PL) data, and voice. There are two basic kinds of PASS downlist that are generally talked about: guidance, navigation, and control (GNC) data and SM data. The GNC computer ships downlist information to toggle buffer 1 of the pulse-code modulation master unit (PCMMU), and the SM computer sends information to toggle buffer 2 of the PCMMU. The BFS GPC also sends GNC and SM data to the PCMMU, but its data goes through toggle buffer 5 in the PCMMU. It is possible to lose one type of downlist and not the others. Each type of downlist will be discussed in more detail below.

Downlist GPC

Since PASS GNC is normally run in a redundant set, i.e., there is more than one GPC doing identical tasks, it is necessary to be able to specify which GNC GPC actually sends its downlist data to the PCMMU. This is done via item entry on SPEC 0, where we designate the DOWNLIST GPC. The downlist GPC will usually be that GNC GPC which we want to send downlist to the PCMMU. This item (ITEM 44) actually designates which GPC will send data to PCMMU toggle buffer 1. It is usually automatically selected to be the lowest numbered PASS GPC in command of one of FC 1, 2, or 3. The crew does not normally have to change this item, but, in the event that the current downlist GPC experiences problems, the ground may instruct the crew to use this item.

Switch Downlisters

Sometimes the DPS flight controllers ask the crew to switch downlisters when the ground loses PASS GNC downlist. The crew performs this action by calling up SPEC 0 (figure 4-1) and entering the key strokes ITEM 44 + X EXEC, where X is the computer that will be the new downlist. Switching downlisters normally recovers the GNC downlist.



TD3870401. ART, 2

Figure 4-1.- Switching downlisters.

Lost GNC Downlist

As stated before, the vehicle can lose one type of downlist and not the others. In the above case, GNC downlist is lost. The ground can tell this is a GNC downlist problem as opposed to loss of signal (LOS) because there is still SM data on the displays. The crew does not have any insight into whether they have downlist or not and must rely on a ground call.

The PASS GNC computers initialize with GPC 1 selected as the downlister, and will automatically switch to the next highest GPC if GPC 1 fails and there are three GNC GPCs remaining in the redundant set. When the vehicle is on orbit, there are normally two GNC GPCs in the redundant set, and when one fails, the crew will have to manually switch downlisters. It is also possible for the interface between the GPC selected as downlister and the PCMMU to fail, or for the PCMMU itself to fail.

11 In the GPC interface fail case, the software will annunciate I/O ERROR D/L, and the ground will have to manually switch downlisters. In the PCMMU fail case, the software will annunciate I/O ERROR PCM, and the crew will perform a procedure to switch PCMMUs.

Lost SM Downlist

As in the case where the vehicle lost GNC downlist, the ground can tell that they have lost SM downlist because displays with SM data go static while the GNC displays still have data. There are two possible problems that could cause loss of SM downlist: an SM GPC problem or a PCMMU problem.

If the problem is suspected to be an SM GPC problem, the crew will run a procedure that checks out the ability of the PCMMU to ship data by bringing up the BFS. If the BFS displays SM data normally, then they know that the SM GPC interface to the PCMMU is bad. If the BFS does not display SM data, then the crew knows that the PCMMU is bad and will switch PCMMUs.

Section 5: Dumps

Dump is a term that describes a method that the crew uses to dump, or list, the contents of a GPC's internal memory. The dump is used in the analysis of a GPC failure to give insight into determining why the GPC has failed. It can also be used to verify that software loaded into a GPC is correct.

A GPC dump impacts the entire flight control room because loss of downlist occurs while the dump is taking place. This can be a big surprise, especially when looking at some important data. Fortunately, dumps do not take very long (dump times will be discussed in later paragraphs).

Dumps simultaneously go to the OPS recorders and into the downlist. Because the dump is captured by the OPS recorders, the crew can initiate the dump during LOS, and if they do, downlist may not even be lost AOS.

Two types of GPC dumps, hardware and software, will be discussed in more detail below.

Hardware Dump

A hardware dump, or hardware initiated standalone memory (HISAM) dump, is the hardware set-up capability of each GPC to dump its memory. When a GPC fails on orbit, procedures require the crew to perform a hardware dump on that GPC. The hardware dump utilizes a software subroutine, but is started via switches.

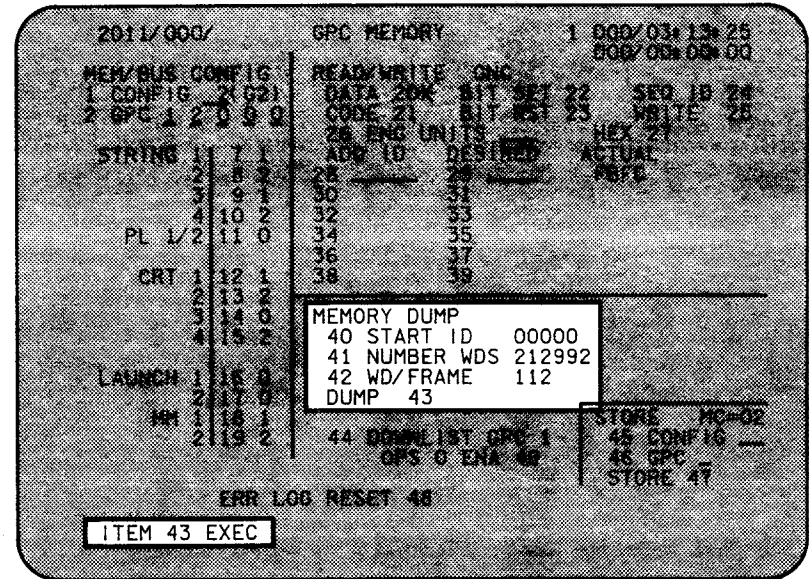
The hardware recognizes an exact switch configuration as a request for a hardware dump. It is difficult for the crew to accidentally start a hardware dump because two switches are on the flight deck and one switch is on the middeck. A GPC that is hardware dumped must be reloaded (IPL'd) because the hardware dump scrambles the GPC's memory.

A double dump of a PASS GPC takes 3 minutes, and a single dump of the BFS takes 6 minutes.

Software Dump

A software dump is a software executed dump using a utility on SPEC 0 (figure 5-1). The software dump uses a software subroutine and is also started via software item entries. Procedures require the crew to perform a software dump to dump the memory of a good GPC that was running at the time of another GPC failure. Analysis engineers can retrieve information from this good GPC software dump (called a sync trace log) that is helpful in determining the nature of the failure of the bad GPC. Software dumps are also used to verify software in the case of installing a spare GPC or after an entry memory uplink (EMU) (see EMU in section 7).

A software dump for PASS GNC takes 2 minutes. PASS SM takes 4 minutes, and PASS OPS 0 takes 6 minutes. The software dump of the BFS will take 6.5 minutes.



TD3870521. ART, 2

Figure 5-1.- Software dump on SPEC 0.

Section 6: Error Log and Error Log Reset

The error log is an internal list of input/output (I/O) errors that is maintained by each GPC. It is similar to, but not the same as, the FAULT SUMM page. This error log is sent to the ground as part of the downlist and is used by the DPS flight controllers.

Error Log Reset

Error log reset is a term not to be confused with I/O reset. Generally, an error log reset does not affect any flight controllers other than DPS. The crew normally performs this action related to GPC failures to reset the computer annunciation matrix (CAM) off-diagonals on panel O1 (figure 6-1). An error log reset is accomplished by doing an ITEM 48 EXEC on SPEC 0 (figure 6-2).

14

Flight data file procedures require the crew to check with Mission Control Center (MCC) prior to performing an error log reset because an error log reset clears the last five GPC-detected errors displayed on the ground. The DPS flight controllers must make sure they have a hard-copy of the latest errors before the crew erases them.

In the GPC failure case, an error log reset is done after the GPC dumps and before the failed computer is IPL'd. The error log reset should not be done before a GPC is dumped because it also clears the sync trace log, which is analyzed by the ground to diagnose GPC problems. In contrast, an error log reset must be accomplished to the failed GPC before it is IPL'd to clear any off-diagonal votes against it. A GPC will not perform an IPL with two or more fail votes against it.

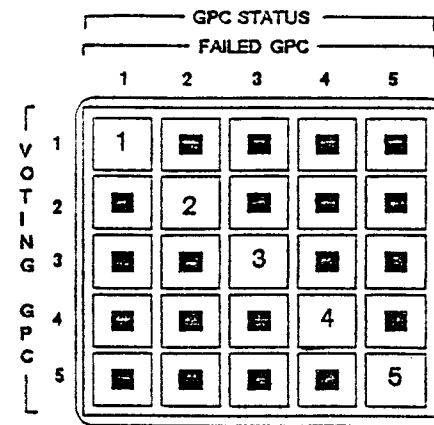
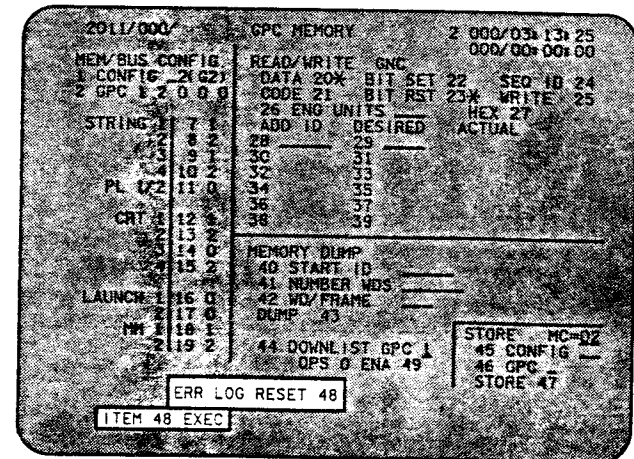


Figure 6-1.— Computer annunciation matrix on panel O1.



TD3870602. ART, 2

Figure 6-2.— Error log reset on SPEC 0.

Section 7: GPC Problems

There are many different kinds of GPC problems that may or may not affect other systems. The intent of this section is not to explain in detail the nature of GPC problems, but to explain briefly what they are and how they may impact other systems.

GPC Failures

Crew procedures for GPC failures vary depending on the flight phase in which the failure occurs.

On ascent, the crew members cannot reach the GPC MODE switches on panel O6 until post-main engine cut-off (MECO) because of the high force of acceleration. Post-MECO, the crew will take the failed GPC to HALT. The crew may also, depending on the flight rules, restring to recover the string that was commanded by the failed GPC.

On orbit, a failed GPC usually does not cause severe problems, except during payload deploys and rendezvous, where special procedures and flight rules apply. The recovery procedures do take quite a while for the crew to accomplish, however. Orbit malfunction procedures require the crew to perform hardware and software dumps (see section 5) and to re-IPL the failed GPC. Orbit procedures also require the crew to restring to different GPCs.

During entry, if the failure occurs before the deorbit burn, the crew will attempt to recover the failed GPC and may even delay deorbit. If the failure occurs between time of ignition (TIG) and MM 304, the crew will take the failed GPC to HALT and restring. Stringent flight rules specify instances where the crew will restring after MM 304.

There are four types of GPC failures that are discussed in the paragraphs below: fail-to-sync (FTS), fail-to-halt (FTH), power failure, and quit.

Fail-to-Sync

An FTS is a condition where one or more GPCs get out of sync with the other GPCs in the set. An FTS is sometimes referred to as a set split (see Set Splits in this section). There are two types of FTSs: redundant set and common set (see Common Set Split and Redundant Set Split in this section). In the case of an FTS, the failed GPC continues to command its assigned buses and, potentially, to send out erroneous commands.

Believe it or not, there are times when the crew expects a GPC to FTS. An example of this is a failure of the GPC's flight critical data bus transmitter and receiver on the string that it is commanding. When the crew restrings that string (assigns it to another GPC), that GPC will then FTS (as expected).

Assume that GPC 1, which is commanding string 1, fails its transmitter and receiver on FC 1. Flight software will announce I/O ERROR FF 1. Both the crew and the ground can tell that this is a GPC problem as opposed to an MDM problem (see At the GPC in section 10). So far this problem could be a transmitter-only problem or a transmitter/receiver problem. There is no insight into the specific problem yet. If the crew restrings (assigns string 1 to another GPC), GPC 1 will FTS. If GPC 1 did not FTS, the crew would know that this was just a transmitter problem. On the subject of transmitters and receivers, a GPC

will FTS immediately if its receiver only fails (regardless of which string it is commanding).

There is a good chance that we can recover a failed-to-sync GPC if the problem is software-related. For a hardware problem, though, such as a transmitter failure, an IPL will not fix the problem, and the GPC may be unusable.

Fail-to-Halt

FTH is a fairly straightforward type of computer failure. The GPC simply stops processing software. Unlike the failed-to-sync case, where the GPC continues to command its string, the failed-to-halt GPC drops its string. A GPC that fails to halt performs an orderly shutdown before stopping its processing.

There is a good chance that the crew will be able to recover a failed-to-halt GPC if the problem was a software error.

Power Failure

A GPC power failure affects us much the same way that an FTH does. Loss of power causes the GPC to FTH. The error signature is slightly different, and a power failure has little chance for recovery, although this is dependent on the location and scope of the power failure.

Quit

A GPC quit is another type of failure that affects us the same way that an FTH does. In the quit case, the GPC essentially goes off in an infinite loop and stops commanding any buses or strings. The effects are the same as in the FTH, but the signatures are slightly different. If the

quit is due to software problems, an IPL will probably recover the GPC.

Set Splits

A set split is an occurrence that is the result of a GPC or a combination of GPCs failing-to-sync (see Fail-to-Sync in this section). When referring to set splits, we are talking about PASS GPCs and not the BFS (which makes sense since the BFS is not a member of any set). There are several types of set splits that will be discussed in the paragraphs below: common set split, redundant set split, four-way set split, 2-on-2 set split, and 2-on-1-on-1 set split.

Common Set Split

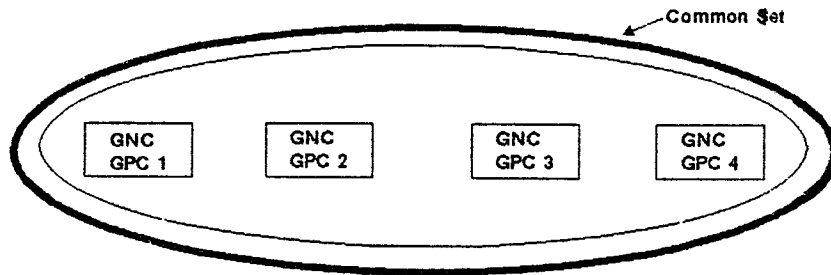
To understand a common set split, understanding of the concept of a common set is necessary.

All PASS GPCs, regardless of the major function in which they are running, operate together in what we call the common set. In the common set, GPCs exchange common set information over the intercomputer communication (ICC) buses and the display/ keyboard (DK) buses. ICC data includes, but is not limited to, state vector information, error status, NBAT configurations, and the identity of the downlist GPC. DK bus data includes CRT assignments, acknowledge and message reset entries, item entries, and major function (MF) switch changes.

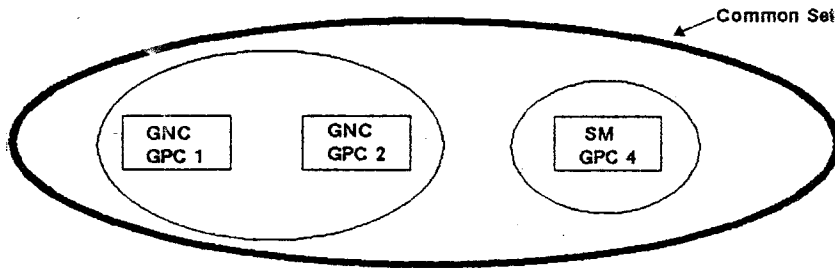
Theoretically, all PASS GPCs that are in RUN are in the common set together. If a PASS GPC is taken to STBY or HALT, it fails out of the common set. If a PASS GPC is taken from STBY to RUN, it joins the common set. Figure 7-1 illustrates the concept of GPCs in the common set during ascent, orbit, and entry.

Now that a common set is defined, explanation of a common set split follows. A common set split is a type of failure where one or more GPCs FTS (see Fail-to-Sync in this section) with the other GPCs in the common set. The failed GPCs continue to run and command their assigned strings and buses. After the GPCs FTS, one or more common sets will run independently (not good). Common sets cannot exchange information between them.

There are several different types of common set splits that we generally think of during ascent and entry that are discussed in the following three sections: four-way set split, 2-on-2 set split, and 2-on-1-on-1 set split.



Ascent/Entry GPC Configuration



Orbit GPC Configuration

Figure 7-1.- GPC common sets.

Four-Way Set Split - A four-way set split occurs when all four GNC computers separate from each other into independent units, each retaining command of its assigned string. Figure 7-1 illustrates how the four computers would normally run, and figure 7-2 illustrates how they will operate after a four-way set split. Flight rules require the crew to engage the BFS during dynamic flight for a four-way set split (see section 2.1).

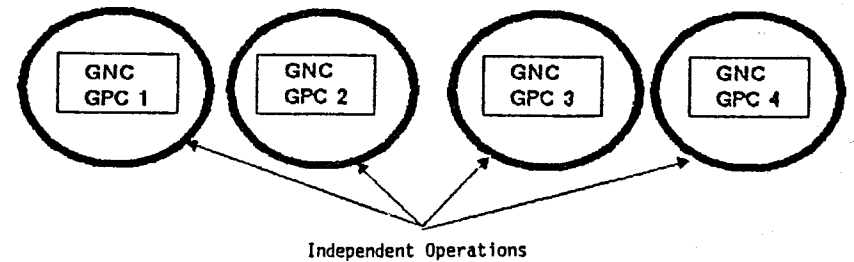


Figure 7-2.- Four-way set split.

Two-on-2 Set Split - A 2-on-2 set split happens when two pairs of GPCs split away from each other. Figure 7-3 illustrates what a 2-on-2 set split looks like. Each pair of GPCs still attempts to control its own strings, and, in most cases, either pair is capable of controlling the vehicle. There are two possible actions when a 2-on-2 set split occurs: engage the BFS or do not engage the BFS.

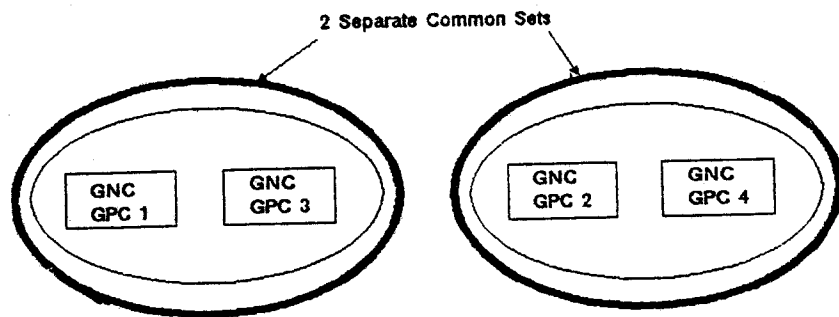


Figure 7-3.- Two-on-2 set split.

Look at the first choice where the crew would have to engage the BFS for a 2-on-2 set split. It is important to know that flight rules do not specifically require the crew to engage the BFS for a 2-on-2 set split. In this case, assume that FCS channels 3 and 4 have previously failed and the vehicle is pre-MECO during ascent. At that point, the crew did not engage the BFS for FCS channels since they still had two FCS channels left (see section 2.1). Now, assume that GPCs 1 and 3 split away from GPCs 2 and 4 as illustrated in figure 7-4. The first pair of GPCs only has FCS channel 1, and the second pair of GPCs has FCS channel 2. Each pair of GPCs has only one FCS channel, and at least two FCS channels are required to fly the vehicle. If the crew engages the BFS, they will pick up FCS channels 1, 2, and 3 on the BFS. Therefore, a BFS engage is required.

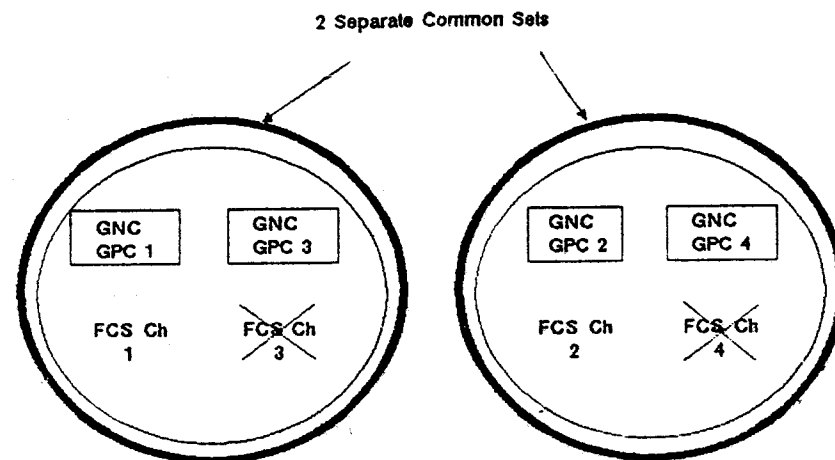


Figure 7-4.- Two-on-2 set split (engage BFS).

In this second choice, the crew will not have to engage the BFS. Assuming that there are no previous failures, the crew and the ground must make a decision as to which pair of GPCs they want to go with. If all things are equal, they will generally choose the pair of GPCs that have the most IMUs. Look back at the previous example and assume that FCS channels 2 and 4 had previously failed; now there is a 2-on-2 set split of GPCs 1 and 3 versus GPCs 2 and 4 (figure 7-5). The crew would choose to fly the vehicle with GPCs 1 and 3 and will not engage the BFS since that pair of GPCs has two good FCS channels.

In both of the above cases, similar considerations must be given to IMUs. At least one good IMU must be available to the GPCs to avoid a BFS engage.

19

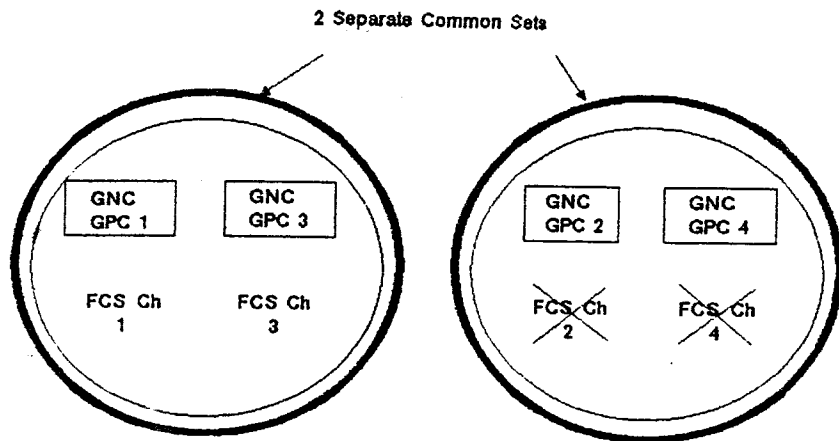


Figure 7-5.- Two-on-2 set split (no BFS engage).

Two-on-1-on-1 Set Split - A 2-on-1-on-1 set split results when two GPCs remain in sync with each other and the other two GPCs split into two individual sets. This type of set split is illustrated in figure 7-6.

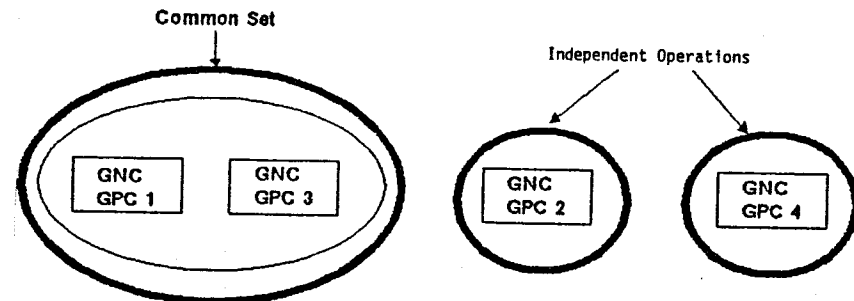


Figure 7-6.- Two-on-1-on-1 set split.

The crew may or may not have to engage the BFS, depending on the capabilities retained by the two GPCs still running together. The same logic for engaging the BFS that was discussed in Two-on-2 Set Split in this section applies here also.

Redundant Set Split

In order to understand redundant set splits, you have to first know what a redundant set is.

A redundant set is a subset of a common set. A redundant set is only allowed in the GNC major function; thus the two terms redundant set and GNC are often used interchangeably. It basically consists of two or more PASS GNC GPCs receiving the same inputs and producing identical outputs. They communicate with each other 400-600 times a second.

During ascent and entry, the redundant set includes all four PASS GNC computers in GNC OPS 1. In this case, the redundant set is also equal to the common set (figure 7-7). The set splits described in Four-Way Set Split, Two-on-2 Set Split, and Two-on-1-on-1 Set Split in this section are both redundant set and common set splits because the redundant set is the same as the common set.

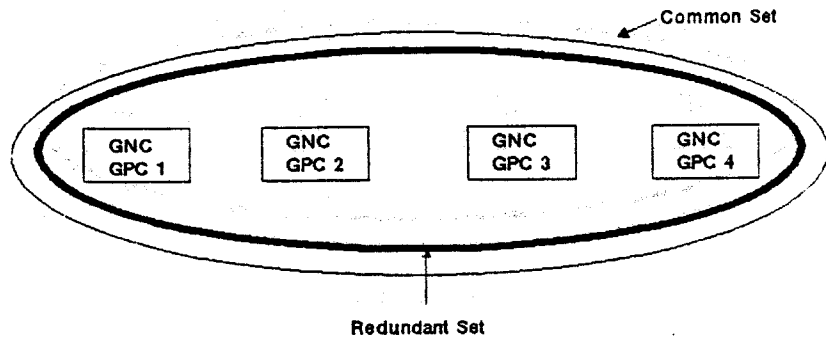


Figure 7-7.– Ascent/entry redundant set.

On orbit, there is a redundant set and a simplex GPC operating within one common set. There are two GNC computers normally running in the redundant set, and the simplex GPC is an SM GPC (figure 7-8).

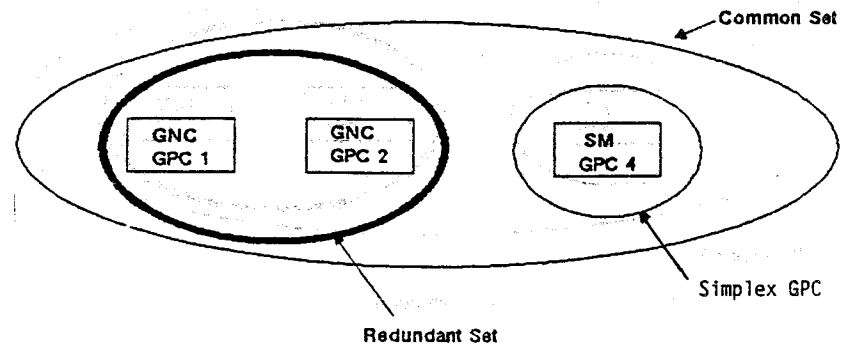


Figure 7-8.– Orbit redundant sets.

Onorbit, since the redundant set does not equal the common set, there can be redundant set splits, in which the affected GPCs will remain in the common set. A redundant set split occurs, for example, when two GNC computers onorbit split apart from each other, but both remain in the common set with the SM computer and each other (figure 7-9). The GNC computers that failed the redundant set will continue to exchange common set information but will not exchange redundant set information. For example, one GNC GPC would not allow the crew to assign the other GNC GPC to its CRT but will give the CRT to the SM computer when the major function on panel C3 is changed to the SM position.

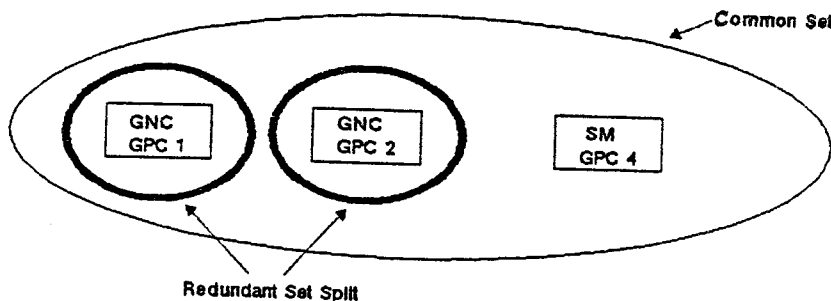


Figure 7-9.— Redundant set split.

Single Hit on a GPC

The DPS flight controllers have insight into the inner workings of each GPC via the error logs discussed previously. Because of this insight, they may detect failures before the crew ever gets a message on board or before the failure affects other systems. A single hit on a GPC is an example of this insight.

Any time a bus control element (BCE) has two or more consecutive I/O transaction failures, the flight software will bypass that element. For example, two consecutive errors on MDM FF 1 will cause FF 1 to be bypassed, and an I/O ERROR FF 1 message will be annunciated on the FAULT page. If there were only one transaction failure on FF 1, the MDM would not be bypassed, but that failure would be registered in the error log.

If DPS is seeing single hits on one of the BCEs in the error log, then there is a good chance that piece of equipment might fail later on. The ground uses this advance warning to plan their actions if the failure should occur.

Moving GPCs

The term “moving GPCs” implies that the crew will have to change the normal software configuration in GPCs for a particular flight phase. Normally, we only attempt to move GPCs while we are on orbit because this procedure takes quite a bit of time and has the potential to cause more failures. Flight rules are very specific as to where the crew will move software and what priorities each type of software has. The crew may have to move GPCs because of GPC problems or because of failures in other systems not directly related to GPCs.

There are several different GPC problems that may require the crew to move the software into another GPC. Some examples of these problems follow.

Flight rules require that BFS be reloaded into a new GPC if it has been declared failed, has lost its downlist capability, has a PL1 or PL2 bus failure, or has one or more flight critical bus receiver failures.

SM GPC software will be reloaded into a new GPC if the original GPC has a failure that prevents it from talking to the PCMMU.

Any type of GPC FTS or FTH will require that GPC’s software be moved while recovery of that GPC is attempted.

There are even a few non-GPC related problems that will require us to move GPCs, such as loss of avionics bay cooling or electrical problems that restrict the capabilities of a GPC.

GPCs require avionics bay cooling in order to operate. If the cooling fails in one of the avionics bays, the crew procedures direct the crew to move the software from the GPCs in that bay to different GPCs. This is a very long and involved procedure.

Electrical problems can cause a GPC to lose capabilities, such as recognizing a BFS engage. Losing control bus CA1 would cause GPCs 3 and 5 to not recognize a BFS engage. As a result, the BFS software would need to be put into another GPC.

Whatever the reason for having to move a GPC, it may impact the ground, since there will probably be loss of downlist during software dumps, hardware dumps, IPLs, etc.

Spare GPC

A spare GPC is actually carried on every flight to use in case one of the other GPCs fails.

The flight rules state three specific cases in which a GPC will be replaced by the spare GPC: if a GPC fails to sync or halt and will not respond to an IPL, if a GPC fails to sync or halt more than once for unknown causes, and if any GPC has one or more failed receivers on flight critical buses 1 through 8 or any ICC bus.

Installing the spare GPC takes 4-1/2 hours and requires the crew members to perform an IFM to put the spare GPC into another GPC’s slot and then activate and check out the spare GPC.

Entry Memory Uplink

Entry memory uplink (EMU) is the process in which the ground uplinks the entry software to a GPC in the Orbiter. EMU will be accomplished if the crew is unable to make a freeze-dried G3 GPC because of failures of both mass memory units (MMUs) during ascent. This procedure is very long and, if everything goes well, requires 45 minutes of uplink to the Orbiter. Uplink can be accomplished through the tracking and data relay satellite (TDRS) or ground stations as necessary. Problems during the uplink process can cause the uplink to have to be started over again, and this extends the amount of time required for the procedure.

There is no GNC downlist during the entry memory uplink process. A minimum duration flight is declared after an EMU unless the crew can recover one of the MMUs. If DPS cannot perform a successful EMU, the crew will perform a BFS entry on flight day 1.

Psychotic IOP

A psychotic input/output processor (IOP) is a failure mode that was identified in the development of the new GPCs and has been designed out of those GPCs. This failure mode was also found to exist in the current GPCs, and some corrective action has been accomplished.

A psychotic IOP could issue spurious commands that could do things such as attempt to open and close the payload bay doors at the same time or release the payload bay door latches. Some actions have been taken to prevent a psychotic IOP from doing things it should not, such as disabling the BFS from opening the payload bay doors during ascent.

Section 8: Input/Output Reset

I/O RESET is a keystroke command entered by the crew members to re-establish I/O with any device that may have been bypassed. An I/O RESET may be done at any time by the crew members with the exception pointed out by the following operations note:

Consecutive I/O RESETs: "Consecutive I/O RESET requests should be at least 10 seconds apart. An ILLEGAL ENTRY message will be issued if a second I/O RESET is requested within 10 seconds of the first." Indications of a successful I/O RESET are: the fault message for the piece of equipment bypassed does not recur and any M's or down arrows for parameters on the affected unit will go away. It makes a difference into which computer an I/O RESET is entered. These differences will be discussed in the following paragraphs.

I/O RESET to PASS

I/O RESET to PASS must be entered to the major function that supports the equipment that was bypassed. An I/O RESET done to the wrong major function will not recover the bypassed equipment.

For example, any of the flight critical MDMs that are bypassed must have an I/O RESET done in the GNC major function in order to recover them. Also, a specific piece of equipment, like a star tracker, is controlled by the GNC major function, and an I/O RESET must also be done to GNC to recover the equipment if it was bypassed.

SM components like the payload MDMs and associated equipment must have an I/O RESET done in the SM major function to clear bypasses.

A port mode (see Port Mode in section 10) does an automatic I/O RESET on the affected bus pair after the port mode is accomplished.

I/O RESET to BFS

Performing an I/O RESET to PASS will normally cause the BFS to upmode the equipment if it is recovered. The exception to this would be if the BFS had dropped an entire string. In this case, an I/O RESET must be done to the BFS to recover the string in the BFS. An I/O RESET to BFS only will never recover equipment for PASS.

The BFS must be able to track at least two PASS strings or it will go disengaged standalone (see No-Go for BFS Engage in section 2). If the BFS should ever drop one of the PASS strings, annunciating BCE STRG X PASS, the crew should do an I/O RESET to the BFS in an attempt to recover the string.

Anytime a PASS FC port mode is accomplished, the BFS will drop the affected string, and an I/O RESET must be done to the BFS to recover the string. If the PL buses are port moded, the BFS will not drop the other PL MDM, but it will not regain the affected MDM, if recovered, until an I/O RESET is done to it.

Note: Since the BFS runs both GNC and SM simultaneously, an I/O RESET in any BFS MF will I/O RESET both GNC and SM.

Section 9: Mass Memory Unit Problems

MMU problems can affect the mission by limiting onorbit operations that use the MMU or by forcing an early end to the mission. MMU failures can also force the crew to engage the BFS if the failures happen at the wrong time. In order to complete a mission, flight rules require two independent sources of deorbit PASS software, which always includes at least one MMU. The second possible source of entry software can be a second MMU, the entry software freeze-dried GPC, or EMU (see EMU in section 7).

Loss of One MMU

Loss of only one MMU does not affect the Orbiter during ascent or entry but does impact our onorbit operations. Onorbit operations with only one MMU forces the crew members to limit access to the MMU, which includes limiting roll-in SPECs, not performing routine SM checkpoints, and limiting telemetry format load (TFL)/data format load (DFL) operations.

Loss of Both MMUs

Loss of both MMUs has different effects, depending on the flight phase in which the failures occurred.

If both MMUs fail during a nominal ascent, the DPS flight controller would have to uplink entry software using the EMU procedure (see EMU in section 7). If DPS cannot uplink the entry software, the crew will have to do a BFS entry on flight day 1. If DPS successfully uplinks the entry

software, the flight director will consider completing flight objectives that do not require using the MMUs. Obviously these objectives could only be completed in OPS 3.

If both MMUs fail prior to OPS 3 during a transatlantic abort landing (TAL), the crew would have to engage the BFS at the OPS 3 transition and continue the abort on the BFS. Crew procedures during a TAL require the crew to keep track of time after they enter OPS 301 PRO to PASS. PASS should transition to OPS 3 in 68 seconds or less, and if it does not, the crew will engage the BFS.

Two MMU failures onorbit require the crew to perform minimum duration flight objectives. Of course, the crew would not be able to call any new roll-in SPECs or perform any activities that require the MMUs. The crew would still be able to get their entry software from the freeze-dried GPC or the EMU procedure. A minimum duration flight is called because one source of entry software has to be an MMU. Consideration is given at this point to performing an in-flight maintenance (IFM) procedure to recover a failed MMU (see MMU IFMs in section 9).

Loss of both MMUs does not affect the mission during entry unless the failures occur before the deorbit burn and the flight director decides to go back to OPS 2 for other reasons, such as an entry wave-off due to bad weather at the landing site. In that case, the crew would not be able to go back to OPS 2 and would be forced to stay in OPS 3.

MMU IFMs

There are two possible IFMs that could be done to recover MMU capability. If a minimum duration flight had been called for loss of both MMUs, a successful MMU IFM could allow the crew to complete a normal mission.

The first MMU IFM is called MMU Contingency Powerup and is used when both MMUs are lost due to power problems. This IFM takes 1-1/2 hours and requires the crew to route alternate power from a dc utility power outlet to an MMU.

The second MMU IFM is called MMU Changeout and is used when one MMU has failed and still has a good data path, while the other MMU is good but has a bad data path. This IFM takes 2 hours and requires the crew to move the good MMU to the good data path.

Section 10: Multiplexer/Demultiplexer Problems

MDMs are very important to everyone, since most of the equipment on the Orbiter passes data through an MDM to the GPC. An MDM is a special purpose digital computer that performs multiplexing, demultiplexing, data buffering, and format conversions of data received from, and of commands sent to, the various Space Shuttle systems. There are many different MDM-type problems that can cause a loss of data. It is important to understand the significance of each type of MDM problem in order to properly manage the system.

First look at how MDMs are attached to a GPC. There are two types of MDMs that are of primary concern: flight critical (FC) and payload (PL).

Figure 10-1 illustrates how the flight critical MDMs, FF 1 and FA 1, are attached to a GPC. The FC 1 data bus is attached to the primary port of FF 1, and the FC 5 data bus is attached to the primary port of FA 1. These data buses are also attached to the secondary ports of the opposite type MDM (see section 11 on Port Moding). Each GPC has an FC data bus (FC 1 through 8) attached to every FF and FA MDM, which allows us to restring so that any GPC can command any MDM (see section 13).

Figure 10-2 illustrates typical equipment that is attached to the FC MDMs. As stated before, all data from the FC LRUs pass through the FC MDMs to the GPC.

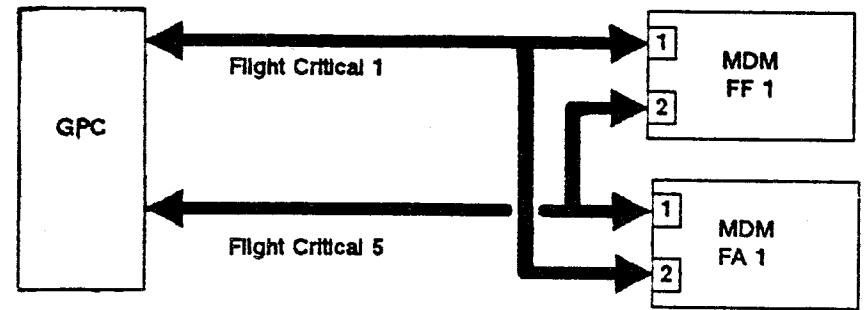


Figure 10-1.- GPC/FC MDM interface.

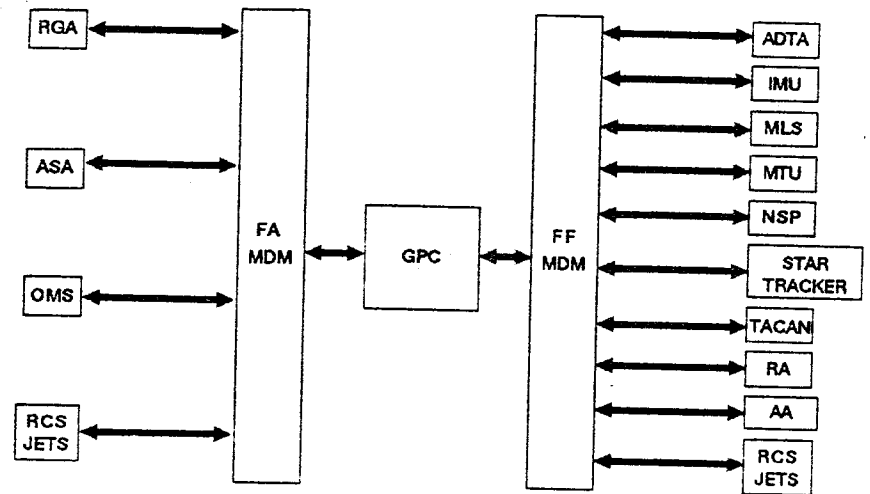


Figure 10-2.- Typical FC MDM LRUs.

27

The other type of MDM that we are concerned about is the payload MDM, which is illustrated in figure 10-3. The payload MDMs handle data related to systems and payload functions on the Orbiter. The payload MDM data buses are wired differently than the flight critical data buses. Payload data bus 1 is attached to port 1 of payload forward 1 (PF 1) and PF 2 (also commonly called payload 1 (PL 1) and PL 2), Payload data bus 2 is attached to port 2 of PF 1 and PF 2. Each GPC is capable of commanding the payload data buses, but normally the BFS has control during ascent and entry, and the SM GPC commands them while on orbit. Flex MDMs, sequence control assemblies (SCAs), and a payload data interleaver (PDI) are also attached to the payload data buses.

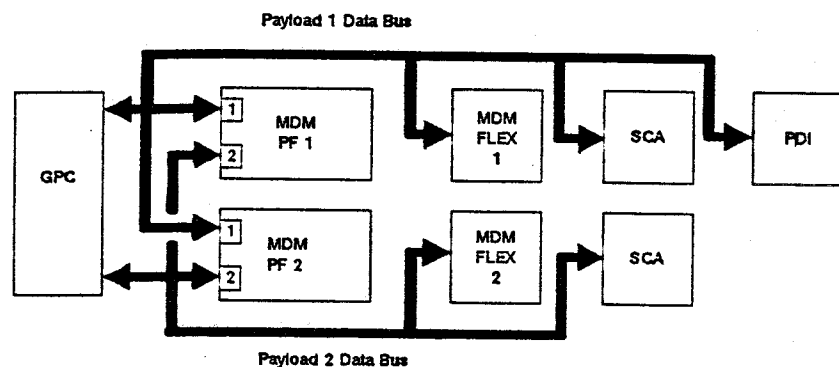


Figure 10-3.- GPC/PL MDM interface.

The following sections discuss several different problems that can affect MDMs.

At the GPC

"At the GPC" is not really an MDM problem, but the signature leads you to believe it is. This failure, as

illustrated in figure 10-4, is a problem with the data bus at the GPC, or a GPC problem as opposed to an MDM problem.

When this failure occurs on an FC data bus, an I/O ERROR FF X or I/O ERROR FA X message is annunciated by the PASS computers, depending on which data bus the failure is on. If there is more than one GPC in the redundant set (see Redundant Set Split in section 7), the DPS flight controllers can tell immediately that the failure is at the GPC because of detailed information on their displays.

During ascent or entry, the BFS detects this as a GPC problem and will bypass the entire string, annunciating BCE STRG X PASS. At this point, the crew also knows it is a GPC problem and will relate this to the ground.

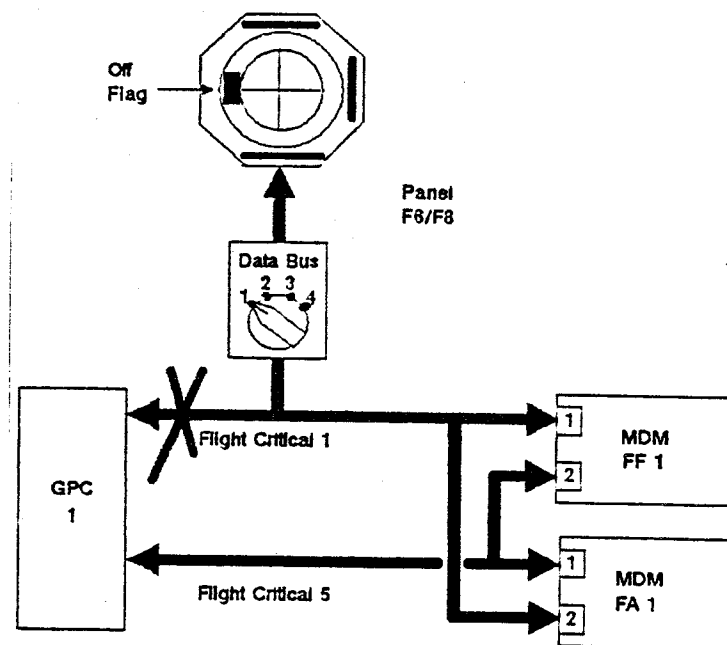


Figure 10-4.- MDM problem at the GPC.

On orbit, the crew might be able to figure out whether this is a GPC or MDM problem using a test that is contained in the flight data file procedure. Look again at figure 10-4 and notice that an attitude direction indicator (ADI) is attached to the FC 1 data bus. The ADI is actually connected to FCs 1, 2, 3, and 4, and the switch in the figure is used to select which data bus will supply data to the ADI. Because of this, the ADI can only be used to determine GPC problems involving the FF MDM, not FA (unless you are already on secondary ports).

In figure 10-4, the break (signified by the large X) is at the GPC side of FC 1. An I/O ERROR FF 1 message will be annunciated by PASS. Orbit procedures would have the crew select FC 1 to drive the ADI, and they would see an off flag, showing that the ADI is not receiving any data. This means that this is a GPC problem. If they did not get an off flag, that would mean that the GPC is still shipping data on FC 1, and the problem was an MDM problem (see At the MDM in section 10).

29

In this case, if the crew port moded string 1 (see section 11), they would recover the FF MDM and lose the FA MDM, since FC 1 would now command the FA MDM on its secondary port.

Procedures require the crew to restring (see section 13) to recover the lost MDM. The GPC that was originally commanding the string may FTS (see Fail-to-Sync in section 7) on the restring, depending on the exact type of failure it has.

The significance of this whole discussion is that if an MDM is lost due to a GPC problem, the crew will be able to get the MDM back if a restring or a BFS engage is done. If a restring is not done, a port mode will allow you to get the MDM back at the expense of losing the other MDM on the string.

This same concept applies to PL MDMs and problems at the SM GPC. Unfortunately, there is no quick way to tell where the failure is, as there was in the FF case. Chances are, however, that SCA, FLEX MDMs, or PDI problems may occur as well as the I/O ERROR PL X if the failure is at the GPC.

At the MDM

The term "at the MDM" describes a type of MDM problem caused by the MDM itself.

In figure 10-5, the break is at the primary port of FF 1. PASS will annunciate I/O ERROR FF 1 for this failure, and, if this is ascent or entry, the BFS would also annunciate I/O ERROR FF 1. Notice that the ADI in figure 10-5 does not have an off flag for this failure because the GPC continues to ship data over the FC 1 data bus.

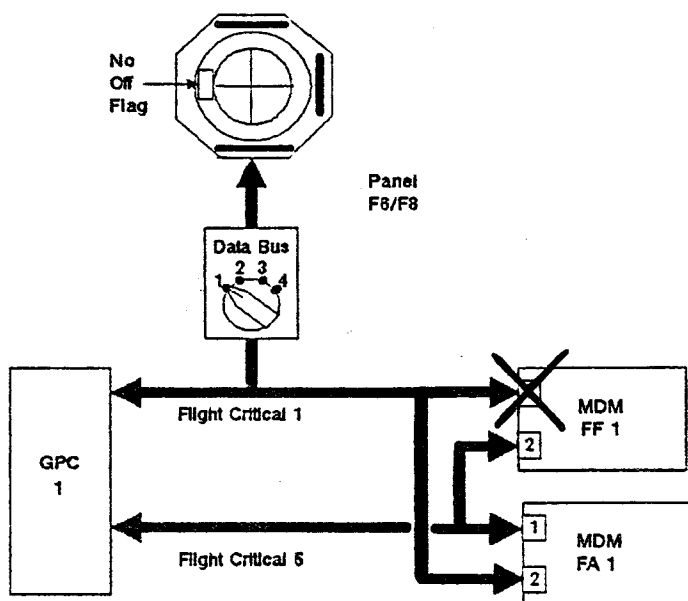


Figure 10-5.- MDM problem at the MDM.

A port mode of string 1 (see section 11) will recover FF 1, because the data will flow through the unaffected secondary port.

A similar analysis can be used for the PL MDMs.

I/O RESET

Sometimes an MDM problem occurs one time and then clears itself. This is called transient. Because of the way flight software operates, the MDM failure is detected, a fault message is annunciated, and the GPC stops listening to the MDM. The MDM remains bypassed by the GPC until the crew performs an action to cause the GPC to try to communicate with the equipment again. An I/O RESET EXEC is normally the first step that procedures call out for the crew to perform. If the problem is transient, the I/O RESET will cause the GPC to start listening to the MDM again. If the problem still exists, the original message will be reannunciated, and the crew will have to perform additional steps to fix the problem.

I/O RESET is covered more extensively in section 8.

Port Mode

Port mode is another way to fix an MDM problem. If an I/O RESET does not recover the MDM, there is a good chance that a port mode will fix the problem. The port mode simply selects an alternate path of data flow to the MDM and also selects a different connection at the MDM.

Port mode is covered in detail in section 11.

Power Cycle

Power cycle is the third way that the crew can fix an MDM problem. Crew procedures generally call for an MDM power cycle if an I/O RESET or a port mode does not work. A power cycle resets the MDM internal logic to an initialized state. Sometimes a power cycle can actually fix whatever was wrong with the MDM. Unlike a port mode, the crew member must perform an I/O RESET after a power cycle to get the GPC to attempt communications with the MDM. If the power cycle was effective, the original fault message will not recur.

When an MDM fails, the analog and discrete outputs are latched to their last state as long as power is still applied to the MDM. A power cycle will cause the MDM to perform initialization and reset the analog and discrete outputs to the zero state. When an MDM is powered off, the analog and discrete MDM outputs are in a zero state.

Power cycles are also called out by procedures for reasons other than trying to fix an MDM problem. An example of using an MDM power cycle for a non-MDM problem is the GPC failure case on ascent. If GPC 1 fails on ascent, the procedures call for the crew to power cycle FF 1. This power cycle will reset to zero the left orbital maneuvering system (OMS) gimbal primary enable command so that the left OMS will not see both the primary and secondary gimbals enabled. Without the power cycle, the primary gimbal will expect to move the engine via commands or thrust vector control (TVC) from FA1. But FA1 is absent due to the GPC 1 failure. If secondary gimbals are enabled while primary gimbals are enabled, the primary TVC and secondary TVC will not agree, and the engine will lock up.

Another example of an MDM power cycle for a non-MDM problem is for an SM GPC failure during payload bay door opening. If the SM GPC fails while commands are being sent to open the payload bay doors, these commands would remain latched in the MDM and the doors would continue to open in an uncontrolled manner. Procedures require the crew to cycle the PL MDMs immediately if the SM GPC should fail during payload bay door opening, which would reset the commands.

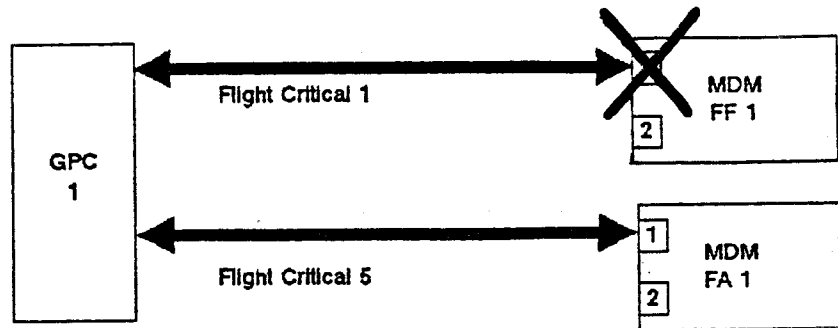
FF/FA Tradeoff

The FF/FA tradeoff case is a classic case that somehow usually ends up on long simulations. This case always gives the flight controllers major problems because it takes them a while to figure what the problem is and how it impacts them.

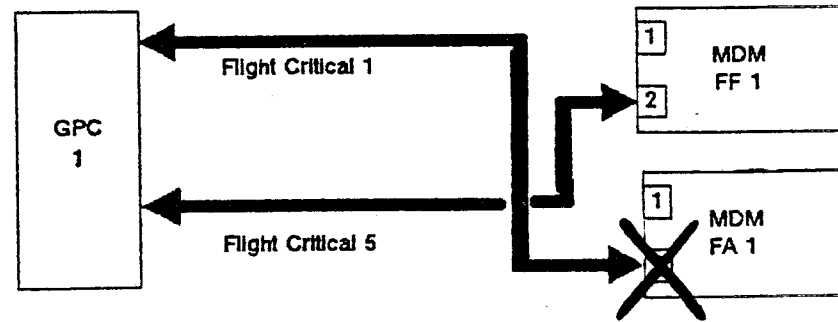
The case is actually a combination of two failures: FF MDM primary port failure and associated FA MDM secondary port failure (or vice versa). This results in a tradeoff between an FF MDM and an FA MDM for a given string; you can have one MDM or the other, but not both at the same time.

As an example, the crew might be forced to trade off vernier jets on the FA side with an IMU on the FF side. The crew is required to port mode on SPEC 1 to select the MDM from which they want to get data.

As an example of this case on string 1, figure 10-6 illustrates where both failures are located. Notice that with primary ports selected, we have data from FA 1 only. With secondary ports selected, we get data from FF 1, but lose data from FA 1.



Primary Port Selected - Down FF 1



Secondary Port Selected - Down FA 1

Figure 10-6.- FF/FA MDM tradeoff case.

This case presents real problems for the flight controllers because they must negotiate among the different disciplines to determine who needs what, and when.

A similar failure case can also arise for a PL MDM tradeoff.

Input Cards

Input cards are the cards in the MDM that provide input data to the GPC (figure 10-7). Unlike the two ports that each MDM has, there are no backup cards for an MDM. The same set of cards is used by both the primary and secondary ports of the MDM. It is possible that normal MDM failure recovery procedures such as an I/O RESET, a port mode, or a power cycle will recover a bypassed card.

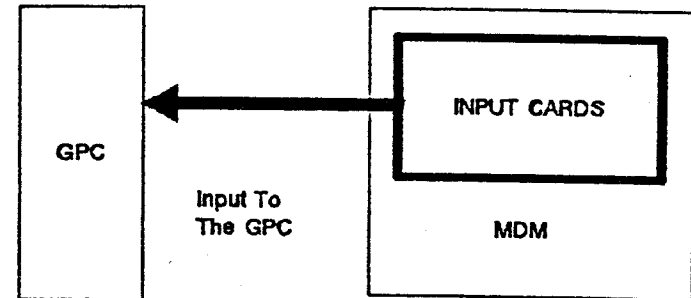


Figure 10-7.- MDM input cards.

When an input card fails, flight software annunciates a fault message to the crew, and the information on that card and all other cards associated with that string element are bypassed. Table 10-1 lists some fault messages associated with card failures. BCE STRG A and C messages are associated with mid-frequency (6.25 Hz) polling of the input cards, and BCE STRG B and D messages are associated with high frequency (25 Hz)

polling of the input cards. Any time a card fails that affects string A or C, the associated string B or D is bypassed also because the mid-frequency data elements are polled again during the high frequency transactions.

Table 10-1.— Fault messages associated with input card failures.

'BCE STRG X A'	X	X	X	X	X	X											
'BCE STRG X B'	X	X	X	X	X	X	X	X									
'BCE STRG X C'																X	X
'BCE STRG X D'									X	X	X	X	X	X	X	X	X
	4	6	7	9	12	15	1	14	1	3	5	8	9	11	6	14	
	FF X MDM Card								FA X MDM Card								

X = Associated MDM (1,2,3, or 4)

As an example, if FF 1 card 6 fails, a BCE STRG 1 A and a BCE STRG 1 B message will be annunciated. In addition, all high frequency and mid-frequency polled data from input cards 4, 7, 9, 12, 15, 1, and 14 will be bypassed.

Look at another example where string B is bypassed only. An FF1 card 14 failure will result in a BCE STRG 1 B message. In this case only the high frequency polled data on cards 4, 6, 7, 9, 12, 15, 1, and 14 will be bypassed. Mid-frequency polled data from cards 4, 6, 7, 9, 12, and 15 will still be passed along to the GPC.

The Reference Data Book in the Flight Data File contains tables in section 2 that list the inputs associated with each type of input card. Table 10-2 is an example of the information contained in the Reference Data Book for FF 1. Notice in the table under the GPC INPUT column that there are two sections labeled A and B. These sections correspond to the string A and B messages you get when

an input card fails. The table lists the input data associated with each type of message. Notice also that there is an "*" by some of the data. The "*" means that this data is also read by another MDM and is not lost by a single failure. In the case of a BCE STRG 1 B message, we definitely lose AA 1, F RCS MANF 1 JET RM, F RCS MANF 1 JETS (OPS 1 and 6), STR TRK -Z PWR discrete, and L ATT REF. All of the other data listed in the table is redundant through another MDM.

Table 10-2.- Reference Data Book FF 1 table.

FF1			
BCE	GPC INPUT	BCE	GPC INPUT
A	F RCS OX He P TK T & P(QTY) - A* MANF 1 P IMU 1 STATUS PNL F6 DISP SWs (eight)	ADTA IMU MLS MTU NSP STKR TAC	ADTA 1 IMU 1 MLS 1 MTU ACCUM 1 NSP 1 -Z S TRK TACAN 1 RADAR ALT 1
B	AA 1 F RCS MANF 1 JET RM JETs (OPS 1 & 6) L ADS DEPLOY -A* STR TRK -Y DOOR -A* -Z DOOR -B* -Z PWR CNTLrs/PBIs/SWs: AFT DAP CNTL -3* RHC -1* SENSE -1* THC -1* FCS CH 1 ORIDE -2* 3 ORIDE -2* 4 ORIDE -3* ENTRY MODE -1* FWD DAP AXES -A* CNTL -1* THC -1* L ATT REF -A* BDY FLP -1* FCS MODE -A* PNL TRIM -A* RHC -1* TRIM INH -A* TRIM -A* RUD PEDAL -1* SBTC -1* MAN -1* MSTR RCS XFEED -1* SEP. ET & SRB -1* NWS POS FDBK -1*	GPC OUTPUT AA TEST ADTA 1 TEST F RCS He PRESS VLV A MANF 1 JETS IMU 1 L OMS GMBL PRI SEL MLS 1 TEST MTU UPDATE -1* NWS ENA (GPC/DIRECT MODES) RADAR ALT 1 TEST TACAN 1 -Z S TRK DISPLAYS: C/W: L RCS (LK/TK P) FWD DAP LIGHTS L FCS MODE LIGHTS BDY FLP AUTO P & R/Y AUTO SPD BRK AUTO RCS COMMAND LIGHTS -A* FWD, L, R RCS OXID QTY FWD, L RCS LOWEST QTY SPI: ELEVON POS SPD BRK POS CMD RUD POS OFF FLAG	

* In INPUT & OUTPUT columns, A or B indicates dual redundancy; 1, 2, 3 or 4 indicates triple or quad redundancy

LRU Power Problems

Some equipment, such as air data transducer assemblies (ADTAs), IMUs, network signal processors (NSPs), and star trackers, is attached to serial I/O. A power failure with the piece of equipment or a serial card problem can result in a fault message, such as BCE STRG 1 ADTA or BCE STRG 3 NSP, unique to the piece of equipment. This equipment is listed under the BCE column of the tables already discussed in the Reference Data Book (table 10-2).

Crew procedures treat these fault messages as if they are MDM-related problems and attempt an I/O RESET, port mode, power cycle, etc. If in fact the message is a result of an LRU power problem, these steps will not do any good as far as recovering the equipment. If a flight controller can identify a failure of the equipment as a power problem, he should tell the crew and avoid having the crew run through a lot of unnecessary and potentially harmful steps.

Output Cards

Output cards are the cards in the MDM that send data from the MDM to the Space Shuttle systems (figure 10-8). Unlike input cards, there are not any fault messages for output card failures. The Reference Data Book in the Flight Data File also has information about output commands from the MDM. Table 10-2 in the last section contains the output commands from FF 1 under the title GPC OUTPUT. The output commands are listed by MDM and not by each individual card. The DPS flight controllers have detailed listings of outputs broken down by card, channel, and bit.

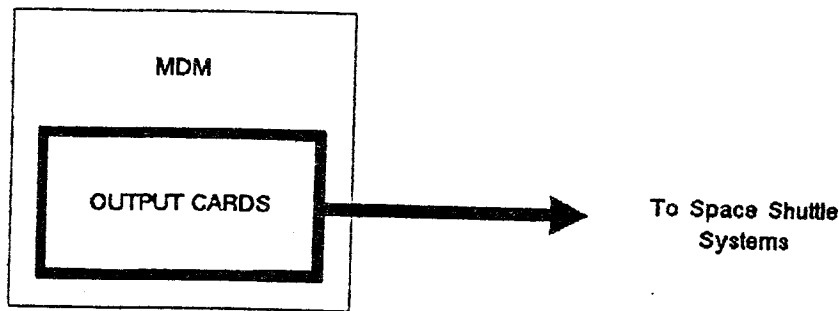


Figure 10-8.—MDM output cards.

Output card failures can only be identified by piecing together a puzzle of various Space Shuttle systems output commands that are not functioning as expected. Some problems, such as unexplained jet failures, are easily identified, and other problems are more difficult to identify. The DPS flight controllers have the necessary information to piece this puzzle together, but they must rely on other users to bring each piece of the puzzle to them. If a couple of output commands that are missing or incorrect are on the same output card, then DPS has enough information to suspect an output card problem.

Once the DPS flight controller suspects an output card problem, he may request the crew to power cycle or port mode the MDM in an attempt to fix the problem. I/O RESETs do not affect output from the GPC/MDM to the systems, only inputs.

Channels and Bits

Each MDM card has several channels. Discrete cards can have up to three channels numbered 0, 1, and 2. Each channel on a discrete card has 16 bits. Each bit on the channel is attached to a unique part of the Orbiter. Figure 10-9 illustrates the data that comes from discrete input card 6, channel 0 on FF 1. Notice that all of the bits were not used on this channel. Analog cards can have up to eight channels and are not further broken down into bits. Serial I/O cards can have up to four channels.

As far as failures are concerned, it is possible that one channel could fail on a card and the rest of the card still be functional. This type of failure is difficult to isolate since there is not a fault message annunciated with input or output card channel or bit failures. The Shuttle computer analysis program (SCAP) has detailed information about MDM channelization.

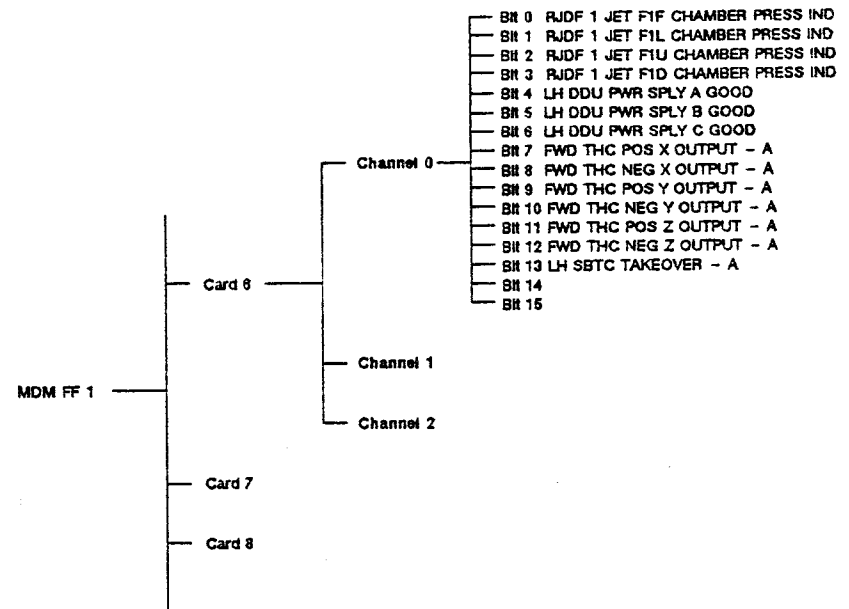


Figure 10-9.- MDM cards, channels, and bits.

Section 11: Port Mode

MDMs have two separate ports that connect the MDM to the GPC. Figure 11-1 illustrates the two ports on a typical MDM. Each port is functionally identical and can be thought of as an alternate path through the MDM. Port 1 is usually called the primary port, and port 2 is called the secondary port. The exception to this terminology is for PL MDM 2 (PF2 or PL2); its primary port is port 2.

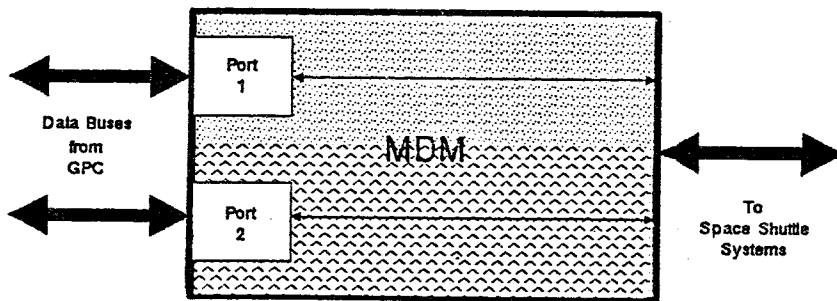


Figure 11-1.- MDM ports - two different data paths through an MDM.

Figure 11-2 shows in more detail how each port is connected within an MDM. As you can see, the MDM I/O cards are the same regardless of which port is selected. There is no redundancy of the cards.

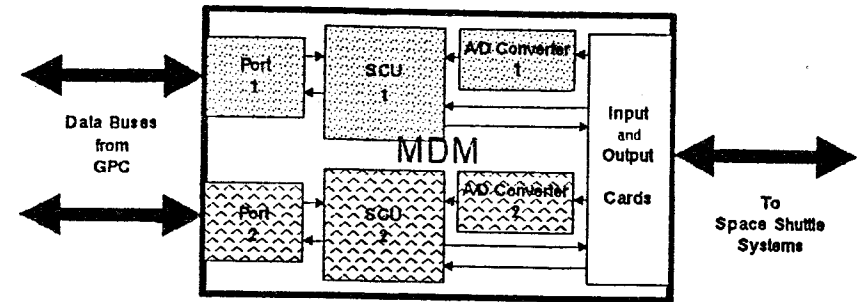


Figure 11-2.- MDM internal components.

Figure 11-3 illustrates a specific example of how data flows for string 1 on primary ports. Data flows from the commanding GPC through FC 1 to the primary port of MDM FF 1. Data also flows from that GPC through FC 5 to the primary port of MDM FA 1. Notice that both MDMs have data flowing through their primary ports.

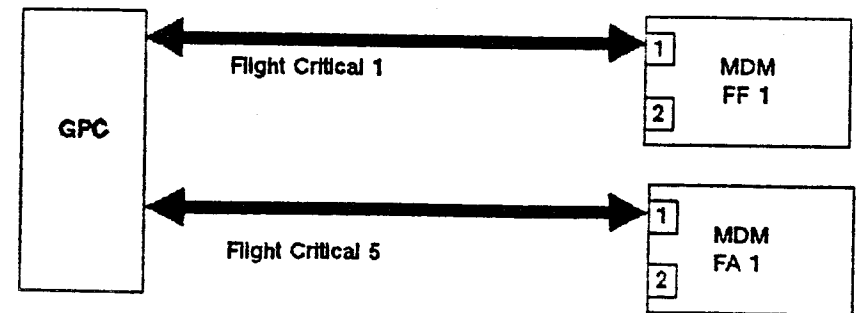


Figure 11-3.- Flight critical MDM data flow with primary ports selected.

Figure 11-4 illustrates how data flows between the GPC and the MDMs when secondary ports are selected. Data flows from the commanding GPC through FC 1 to the secondary port of FA 1 and flows from that GPC through FC 5 to the secondary port of FF 1. Notice that the flight critical data buses have swapped in respect to the MDM to which they are talking.

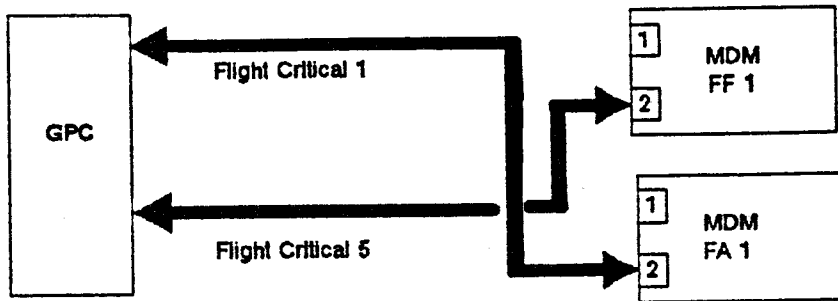


Figure 11-4.- Flight critical MDM data flow with secondary port selected.

38

As discussed in section 10, payload MDMs are wired differently than flight critical MDMs. When you have primary ports selected for the payload MDMs, payload 1 data bus is attached to port 1 of MDM PF 1, and payload 2 data bus is attached to port 2 of MDM PF 2. Port 2 of MDM PF 2 is actually its primary port. Figure 11-5 illustrates data flow for the SM GPC and the payload MDMs with primary ports selected.

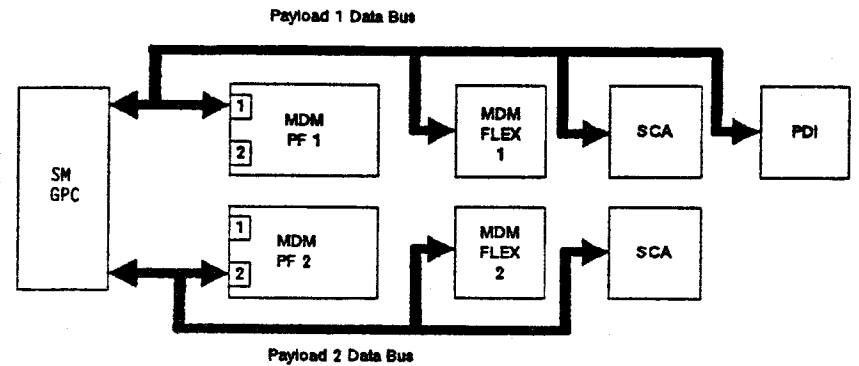


Figure 11-5.- Payload MDM data flow with primary ports selected.

Figure 11-6 illustrates data flow for the SM GPC and the payload MDMs with secondary ports selected. Notice that payload 1 data bus is attached to port 1 of MDM PF 2, and payload 2 data bus is attached to port 2 of MDM PF 1.

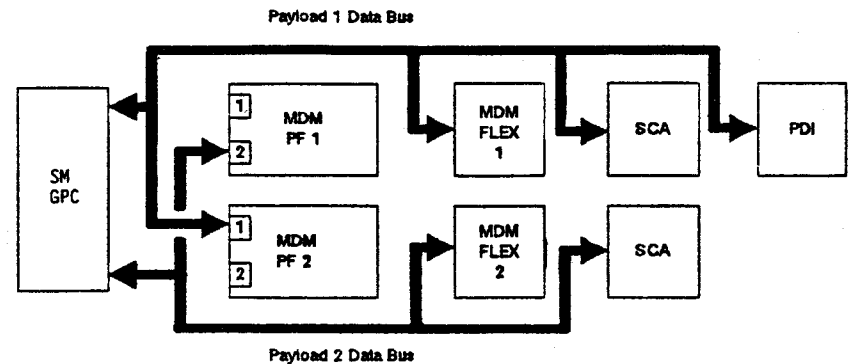


Figure 11-6.- Payload MDM data flow with secondary ports selected.

The methods used to perform a port mode vary depending on the flight phase. An automatic I/O RESET is performed to the affected data bus pair by flight software when a port mode is done, so there is no need for the crew members to enter one from the keyboard. If the original message for the failed MDM does not recur, and the M's or down arrows for the affected parameters go away, then the port mode fixed the problem.

Ascent

During ascent, the crew does not have the capability on-board to port mode, so the MCC must do it for them. If the ground uplinks a port mode, they do not have the option of port moding back to the primary port as the crew could in other flight phases. It might be desirable to port mode back if a hidden failure is found on the secondary port. The BFS does not recognize a PASS port mode and will bypass any string that is port moded. An I/O RESET EXEC to the BFS will cause it to start tracking the string that was port moded.

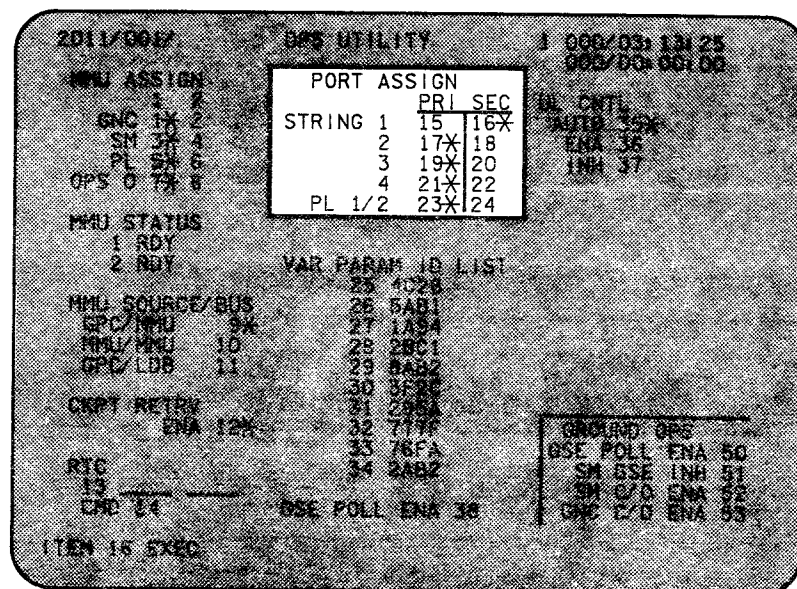
39

Uplink a Port Mode

As discussed in the preceding paragraph on ascent, the crew cannot port mode during ascent. The MCC can perform a port mode using an uplink. The uplink is actually a read/write command to SPEC 0 that enables the routine to execute the port mode. Once a port mode is performed, the MCC cannot send another uplink up to switch back to primary ports.

Orbit

On orbit, port moding is done using items 15 through 24 on SPEC 1, DPS UTILITY, as illustrated in figure 11-7. In order to mode string 1 to secondary ports, the crew member would enter ITEM 16 EXEC. SPEC 1 is not available in OPS 1, 3, or 6.

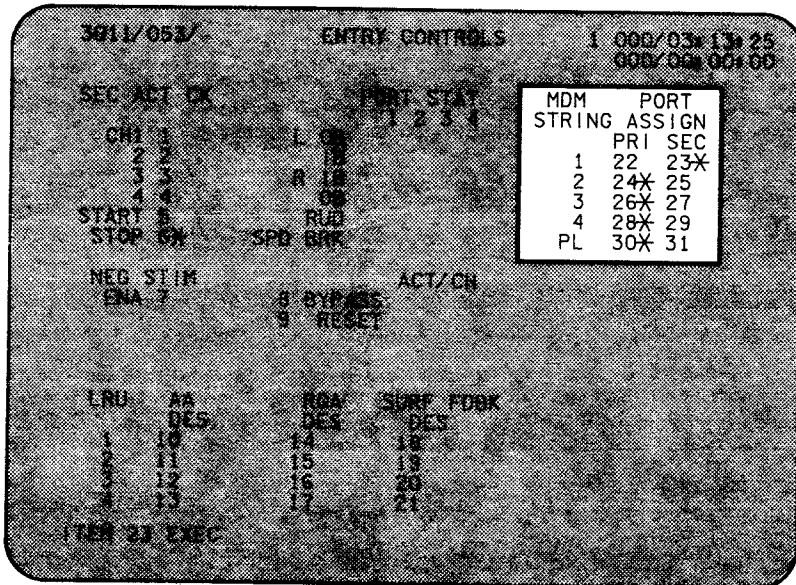


TD3871121. ART: 2

Figure 11-7.- Port mode on SPEC 1.

Entry

During entry, port moding is performed on SPEC 53, ENTRY CONTROLS, using items 22 through 31. SPEC 53 is illustrated on figure 11-8 where the crew members have entered ITEM 23 EXEC to port mode string 1.



TD3871131. ART, 2

Figure 11-8.- Port mode on SPEC 53.

Section 12: Read/Writes

A read/write (R/W) is a software change made to GPC memory. These are sometimes referred to as GPC memory writes (GMEMs). R/Ws can be performed by the crew using SPEC 0 or can be uplinked by the MCC. Normally the MCC will perform the R/Ws for the crew because they can perform more at a time and there is less chance of error. Most types of R/Ws will not be done during dynamic flight phases because there is a chance that they could cause GPC failures if done incorrectly. During an MCC uplink of a R/W, SPEC 0 is called up and must not be dropped until the R/W is complete.

R/Ws fall into several categories: approved and verified, approved and unverified, and "hip pocket."

Approved and Verified Read/Writes

The flight rules state that "only those write procedures which have completed software development verification, crew procedural verification, and been documented in a controlled manner will be considered for use." Approved R/Ws must be verified before every flight. It is expensive to verify R/Ws, and therefore the number of preflight approved R/Ws is limited. The following list is an example of R/Ws that are preflight approved and verified.

- OPS 3 overlay uplink (PASS)
- Flight critical MDM port moding (PASS)
- Heads up roll
- OMS tank mixed crossfeed (PASS)
- IMU BITE mask (PASS)
- Select alternate MMU area 2 without IPL (PASS)
- Fourth LRU substitution RGA, AA, and aerosurface feedback (BFS)
- IMU CAL terms (PASS and BFS)
- SRB rate gyro deselect (PASS and BFS)
- TAL alternate area 3 MMU
- Enable BFS PLBD outputs
- Disable BFS PLBD outputs

The most current list of preflight approved R/Ws is contained in section 7 of the Flight Rules book.

Another point needs to be made about approved R/Ws. Once an approved R/W is entered into a GPC memory, additional R/Ws must be reverified to make sure that they do not impact the changes that have already been made.

Approved and Unverified Read/Writes

Approved and unverified R/Ws are the second category of R/Ws. These R/Ws have been preflight approved but were not officially verified prior to the flight. It takes a minimum of 24 hours for the software developer to verify a R/W. The following list is an example of approved and unverified R/Ws.

- Aft RCS jet priority change (PASS)
- Entry memory uplink/ORB GPC
- RGA/AA compensation bias
- GRTLs yaw jet priority
- Reset aerosurface PFB
- Prelaunch jet select
- RCS RM override
- SSME position prediction

The most current list of approved and unverified R/Ws is contained in section 7 of the Flight Rules book.

"Hip Pocket" Read/Writes

Generally, there are R/Ws that people know about or think might be able to fix a problem but have never been approved or officially verified. These are called hip pocket R/Ws, or real-time GMEMs, and must be verified prior to use.

24-Hour Approval

All unverified and hip pocket R/Ws must be verified prior to being used. The software developer (IBM for PASS and Rockwell for BFS) requires at least 24 hours to verify one of these R/Ws.

Section 13: Restringing

As of this writing, the concept of restringing is very controversial and subject to change. This section attempts to explain restringing without being controversial.

In order to understand the concept of restringing, the definition of a string is needed. A string is a group of equipment hardwired to a pair of FC data buses that is interchangeable as a group between GPCs. If the GPC that has command of the string fails, the equipment on that string is lost until the string is assigned to a good GPC. This assignment is called restringing. One important dedicated note is that equipment like the master events controllers (MECs), engine interface unit (EIU), dedicated display unit (DDU), and heads up display (HUD) can receive data from more than one flight critical data bus, and that equipment will not necessarily be lost when a string goes down.

43

Figure 13-1 illustrates typical equipment that is considered part of string 1.

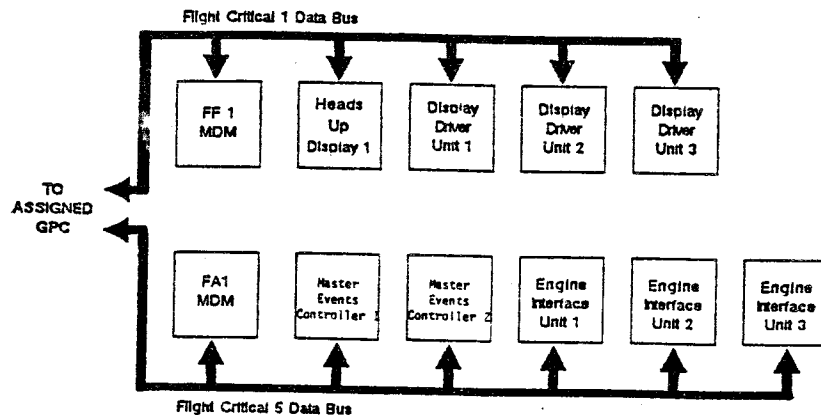


Figure 13-1.- Equipment attached to string 1.

Restrining Post-MECO

Restrining post-MECO is done only during the ascent flight phase to regain critical systems redundancy. If a computer fails, the string that was assigned to that computer is lost. The remaining equipment, such as HUDs and DDUs, can be quickly reconfigured to get data from a different string; however, the input and output from equipment attached to the FF and FA MDMs is lost completely.

The following is an example of some of the equipment that is attached to FF 1 and FA 1 MDMs that would be lost if GPC 1 failed on ascent (GPC 1 is normally assigned string 1).

FF 1 --- ADTA 1, AA 1, IMU 1, tactical air command and navigation system (TACAN) 1, MLS 1, NSP 1, and reaction jet driver forward (RJDF) 1B

FA 1 --- RGA 1, aerosurface servo actuator (ASA) 1, left (L) OMS control valve 1, right (R) OMS control valve 1, and L OMS primary gimbal (P GMBL)

The crew can restringe (assign the string to a good GPC) to recover all of the equipment, but restringing in powered flight is unwanted. There is a slight chance the other computers could fail during restringing, which is not desirable during this time because it could cause the vehicle to lose control. Flight rules require the crew to wait until after MECO before they attempt restringing. By recovering equipment at the earliest opportunity, future BFS engage for loss of equipment, such as IMUs or FCS channels, may be avoided.

Once a GPC has failed, the crew will normally set up the NBAT assigning the failed GPC's string to the highest remaining GPC and implementing the change after MECO; hence the term "restring post-MECO."

Stringing Philosophy

If all of the GPCs are running, stringing is a simple decision. However, if there are any computers which have failed, the proper stringing configuration can be a nightmare. The DPS flight controller has the ultimate responsibility to come up with the proper string configuration, but relies on the knowledge of other affected flight controllers to come up with the optimum configuration.

According to the flight rules, restringing can be done post-MECO to regain critical systems redundancy and during entry to recover data paths lost due to GPC failures to recover strings 1, 2, or 3.

For deorbit burn and entry, the strings will be distributed to minimize the loss of critical functions for a subsequent GPC failure. The following is a partial list of the equipment priority that is considered in restringing with the highest priorities listed first:

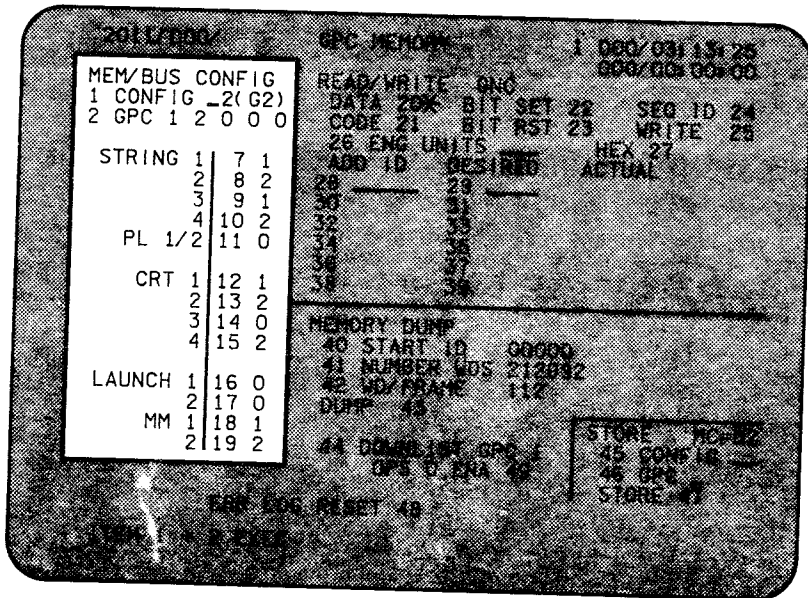
- One IMU (transient GPC concern only)
- Two ASAs
- Two RGAs
- Two AAs
- Two IMUs
- Two yaw jets
- Pitch jets (one-fault tolerant)
- RHC contacts (one-fault tolerant)
- Three ASAs
-
- ...
- ..

The complete list of priorities is contained in the Flight Rules book. In this example, protecting one IMU is considered more important than protecting two ASAs.

NBAT

NBAT stands for nominal bus assignment table, but sometimes the NBAT is called the memory configuration table. The NBAT is contained on SPEC 0 and is the tool the crew uses to change the DPS stringing. Figure 13-2 illustrates SPEC 0 with the NBAT highlighted. In order to change the string configuration, the crew enters items 7 through 11 with the desired stringing changes and then does an OPS transition or an OPS mode recall to implement the table.

The terminology used in restringing can sometimes be confusing. Normally, a new string configuration is called up to the crew as, in an example of a GNC OPS 2 restring, "restring one-two-two-one." The crews all understand this to mean that GPC 1 should be assigned to strings 1 and 4 and GPC 2 should be assigned to strings 2 and 3. The crew calls up the NBAT for GNC OPS 2 (figure 13-2) and enters ITEM 9 + 2 + 1 EXEC to change the string assignment from 1212 to 1221. Then an OPS mode recall, OPS 201 PRO, is done to implement the change.



TD3871331. ART. 2

Figure 13-2.- NBAT on SPEC 0.

Section 14: Systems Management Checkpoint

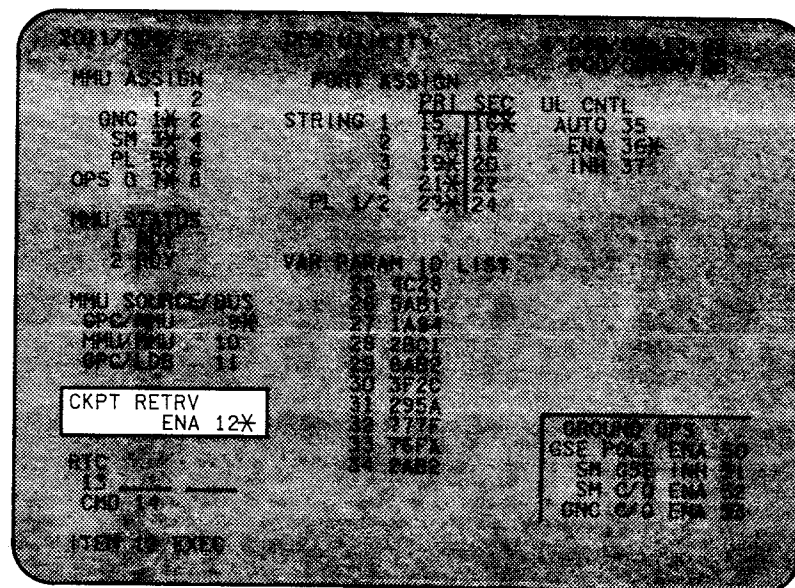
The SM checkpoint is used by the crew to save a copy of the current SM upper and lower C&W alert limits. Crew procedures require the crew to save the SM checkpoint to both MMUs.

You might be wondering why you would want to save an SM checkpoint. If the SM GPC should fail, flight software will lose all of the alert limit changes that have been entered. If the crew previously saved an SM checkpoint, they can bring up a new SM GPC and it can retrieve the SM checkpoint that was saved to the MMU. If the crew did not have the capability to retrieve an SM checkpoint, they would have to manually reenter all of the alert limit changes on SPEC 60, SM TABLE MAINTENANCE, or the MCC would have to uplink a table maintenance block update (TMBU) (see section 15). Obviously, the SM checkpoint can potentially save a lot of time.

46

Checkpoint Retrieve Enable

In order for a checkpoint to be retrieved when the crew brings up a new SM GPC, they must have checkpoint retrieve enabled on SPEC 1 (figure 14-1). This enable is performed with an ITEM 12 EXEC. This item is a toggle, and the crew can tell it is properly enabled if an asterisk is next to the number 12.



TD3871411. ART, 2

Figure 14-1.- Checkpoint retrieve enable on SPEC 1.

If checkpoint retrieve is not enabled, the new SM GPC will load SM software but will not retrieve the checkpoint. The crew can get out of this predicament by taking the SM GPC back to OPS 0, enabling checkpoint retrieve, and then bringing the GPC back to OPS 201.

Section 15: SPECs

SPECs are commonly referred to as the displays that the crew calls up to perform different tasks. The term SPEC can sometimes be confusing because it is used to describe both SPECs and DISPs. SPECs are interactive, and DISPs are used to display data only.

Two-SPEC Limit

Crew members have said that they could not call up a certain SPEC because they had violated the two-SPEC limit. This is an unusual predicament brought about because there is a limit of two SPECs (not DISPs) that can be called up in any given major function (GNC, SM, or PL). What really causes problems for the crew is a DEU failure when a SPEC is already called up on it. That SPEC still counts as one of the two SPECs in that major function. There are two things that DPS can do to get the crew out of this predicament. DPS can uplink a RESUME to the affected DEU, assuming there is still a GPC assigned to it. The other option is to perform an OPS transition in that major function, which will automatically RESUME the SPEC.

47

Roll-In SPECs

A roll-in SPEC is a display that has its background format stored on the MMU. In order to call up one of these displays, the crew must have at least one MMU available. The background format is stored on the MMU because there is not enough room available to store it in the DEU or in the GPC.

There is a two-SPEC buffer for roll-in SPECs that stores the last two SPECs called up (actually, it is a little more complicated than that). Basically, if the crew already has the roll-in SPEC that they need called up and the MMUs fail, they can still call up the display because it is in the buffer.

When one MMU fails, the flight rules call for the crew to minimize use of the remaining MMU which includes limiting calling roll-in SPECs as much as possible.

Roll-in SPECs are basically DISPs numbered 66-98 (excluding SPEC 78 and 79, which are SM SYS SUMM 1 and 2). Roll-in SPECs may also include some 200 series SPECs that are payload unique.

Section 16: TMBU

TMBUs are prepared by the DPS flight controllers based on inputs from other flight controllers. TMBUs are basically the same as changes in SM upper and lower alert limits that the crew could make on SPEC 60, SM TABLE MAINTENANCE. The DPS flight controllers can prepare these changes for the crew and thus save them a lot of keystrokes. Some flights, like Spacelab, require a lot of TMBUs. One tricky thing about TMBUs is that TMBUs are rejected if SPEC 60 is already called up, and if the crew calls up SPEC 60 while TMBUs are being uplinked, they will get ILLEGAL ENTRY for any items they try to enter.

Section 17: Variable Parameters

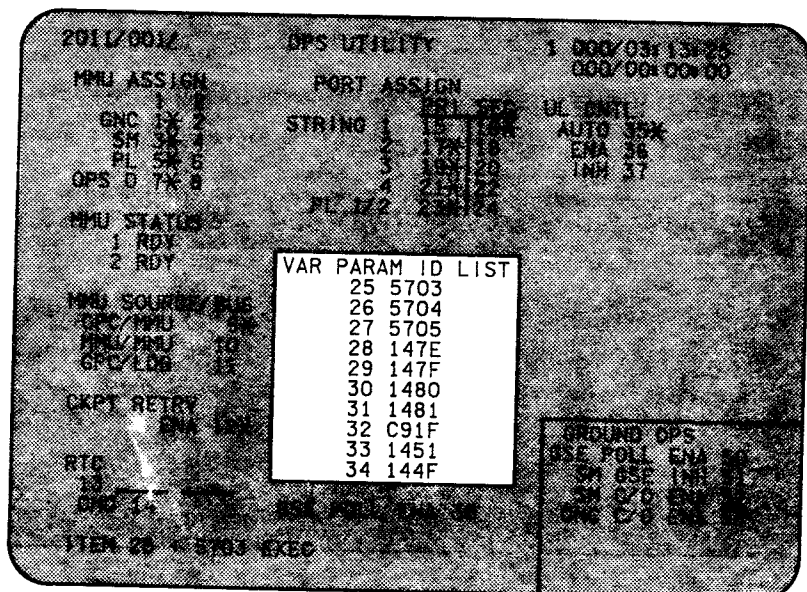
Variable parameters are used to display the contents of any 10 GPC addresses in each major function to the ground. The crew can change these addresses on SPEC 1 (figure 17-1), or the ground can uplink variable parameters. The crew cannot see the contents of the variable parameters on this display. They can only see which addresses are selected for variable parameters. Normally the variable parameters are set for things on which the flight controllers do not have telemetry.

Table 17-1 is an example of the GNC variable parameters that might be used on ascent. Variable parameters are also loaded before uplinking a R/W (see section 12) to verify the contents of the address before and after the uplink.

Table 17-1.— Typical ascent variable parameters.

Slot	Address	Description
1	5703	FRCS jet select status
2	5704	ARCS jet select status
3	5705	ARCS jet select status
4	147E	NBAT string 1
5	147F	NBAT string 2
6	1480	NBAT string 3
7	1481	NBAT string 4
8	C91F	Alternate TAL site
9	1451	MMU assigned GNC MF
10	144F	PRI/SEC port status

49



TD3871701. ART. 1

Figure 17-1.— Variable parameters on SPEC 1.

Appendix A: Acronyms and Abbreviations

AA	accelerator assembly	GMEM	GPC memory write
ADI	attitude direction indicator	GN&C, GNC	guidance, navigation, and control
ADTA	air data transducer assembly	GPC	general purpose computer
AOS	acquisition of signal		
APU	auxiliary power unit	HISAM	hardware initiated standalone memory
ASA	aerosurface servo actuator	HUD	heads up display
BCE	bus control element	I/O	input/output
BFC	backup flight controller	IBM	International Business Machines
BFS	backup flight system	ICC	intercomputer communication
		IFM	in-flight maintenance
C&W	caution and warning	IMU	inertial measurement unit
CAM	computer annunciation matrix	INCO	instrumentation and communications officer
CRT	cathode ray tube		
		IOP	input/output processor
DDU	dedicated display unit	IPL	initial program load
DEU	display electronic unit		
DFL	data format load	L	left
DK	display/keyboard	LOS	loss of signal
DPS	data processing system	LRU	line replaceable unit
EIU	engine interface unit	MCC	Mission Control Center
EMU	entry memory uplink	MDM	multiplexer/demultiplexer
		MEC	master events controller
FA 1	flight aft 1	MECO	main engine cutoff
FC	flight critical	MF	major function
FCS	flight control system	MLS	microwave landing system
FF 1	flight forward 1	MM	major mode
FTH	fail to halt	MMU	mass memory unit
FTS	fail to sync		

NAV navigation
NBAT nominal bus assignment table
NSP network signal processor

OI operational instrumentation
OMS orbital maneuvering system
OPS operations

P primary
PASS primary avionics software system
PCMMU pulse-code modulation master unit
PDI payload data interleaver
PF payload forward
PL 1 payload 1

R right
R/W read/write
RGA rate gyro assembly
RJDF reaction jet driver forward
RM redundancy management

SCA sequence control assembly
SCAP Shuttle computer analysis program
SM systems management
SRB solid rocket booster

TACAN tactical air command and navigation system
TAL transatlantic abort landing
TDRS tracking and data relay satellite
TFL telemetry format load
TIG time of ignition
TMBU table maintenance block update
TVC thrust vector control

51 END