N85-16899

INTEGRATED DESIGN CHECKOUT OF SHUTTLE PAYLOAD AVIONICS INTERFACES

JOHN F. MURATORE, 1LT, USAF
6555th AEROSPACE TEST GROUP
SPACE LAUNCH VEHICLE SYSTEMS
CAPE CANAVERAL AFS, FL

KATHY K. WHITCOMB
NASA CARGO OPERATIONS
CITE AND SOFTWARE BRANCH
KENNEDY SPACE CENTER, FL

Orbiter/payload avionics integration testing is a relatively new activity in the shuttle program. Payloads flown to date have shown extensive orbiter interfaces. This paper describes the three modes of testing at Kennedy Space Center and Cape Canaveral Air Force Station used to verify orbiter/payload avionics interfaces. These modes consist of orbiter testing using generic payload simulators, payload testing utilizing the actual payload and a high fidelity orbiter simulator, and interface testing with the actual orbiter and payload. Several special avionics techniques, such as the split flight computer technique have been developed to accomplish this testing. Experience from the first six shuttle cargoes is reviewed with emphasis on problems found in testing that would have hampered mission success.

INTEGRATED DESIGN CHECKOUT OF SHUTTLE PAYLOAD AVIONICS INTERFACES

Many of the challenges that the Space Shuttle Program generated in integrated avionics have been met and conquered by a variety of sophisticated techniques. In this session we have seen presentations of the tremendous challenges associated with flight computer, ground computer, and simulator hardware and software development. The challenges which we are going to discuss in this presentation are very different from these previous areas. The challenges of these other areas have been substantially met and resolved. In the area of checkout of shuttle payload avionics, however, we are just beginning to understand the true nature of the challenge. In this presentation we will discuss our approaches at Kennedy Space Center (KSC) and Cape Canaveral Air Force Station (CCAFS) to perform checkout of the avionics interfaces between the orbiter and payloads. This presentation will not discuss classified defense payload processing. We will, however, discuss integration of the Inertial Upper Stage (IUS) with NASA payloads such as the Tracking and Data Relay Satellite (TDRS). Also, the discussion of Spacelab processing will be limited because we have not yet completed Spacelab processing.

The primary challenge that we face is how to test shuttle and payload avionics so that we have confidence that when the payload arrives on orbit it will successfully function. These interfaces are definitely substantial. Table 1 shows a list of payloads carried to date on-board shuttle missions. Several observations can be made from this table. First, even "simple" payloads like the OSTA and OSS-1 pallets carried on STS-2 and STS-3 have significant avionics interfaces. Second, the amount of interfaces between the orbiter and the payloads are growing as we fly more sophisticated payloads. This trend will probably continue for some time. The Centaur, for example, will have more extensive avionics interfaces than the IUS. Therefore, we have a significant challenge which is growing.

There are many factors which complicate this problem. Some of these are very unique to payload integration. For example, for many of these payloads, the first time the payload avionics is actually connected to orbiter avionics may be as late as 2 weeks before launch. A major part of the challenge of interface checkout has been to define testing to detect problems in the interfaces as early as possible. Another unique complicating factor is that in many cases we are dealing with organizations outside of the Shuttle Program. In some cases, the users have been extensively involved with the shuttle prior to arriving at the launch base; in others, they have not. In almost all cases, the payload users are used to flying expendable boosters where the interfaces between satellite and booster are very limited (e.g., separation indicators). The large involvement of shuttle hardware in payload mission success (providing power, telemetry, attitude control, pointing, commands, etc.) is a new phenomena to most users. Another major complicating factor is that the complex payload support services provided by shuttle are just beginning to mature. The S-Band Payload Communications System, for example, did not arrive at KSC much before its first use by a payload. This lack of experience with this hardware at the launch base has complicated interface verification between cargoes and the orbiter.

In order to meet the challenge of checkout of orbiter payload avionics interfaces, a system has evolved which is based on three modes of interface checkout. This system is still evolving and is very fluid depending on a specific payload user's needs. In general, these three modes are:

MODE A - CHECKOUT OF ORBITER SUPPLIED SERVICES UTILIZING ACTUAL ORBITER AND
         GENERIC PAYLOAD SIMULATORS

MODE B - CHECKOUT OF THE PAYLOAD TO ORBITER INTERFACES USING THE ACTUAL
         PAYLOAD AND A HIGH FIDELITY ORBITER SIMULATOR
         (CARGO INTEGRATION TEST EQUIPMENT (CITE)) AT KSC

MODE C - INTERFACE TESTING UTILIZING ACTUAL PAYLOAD AND ORBITER

This mode of terminology should not be confused with the "levels of integration" used by NASA Cargo at KSC. In using this "mode" terminology we hope to show how a cargo processing through the shuttle launch site really consists of three distinct phases with different techniques and objectives. This mode terminology will not be found in any formal documentation and is intended for clarity in this paper.

The main purpose of Mode A is to verify that the orbiter is properly configured for a given payload and that all of the orbiter support services are functioning.

Mode B verifies the functioning of the major avionics interfaces between the actual payload and an Orbiter Simulator (CITE). In addition to interface verification, several other types of testing are performed in Mode B. Most payloads contain ordnance devices such as separation pyrotechnics which are tested in this mode.

Most payloads have on-orbit control centers. For detached payloads these control centers are usually not at Shuttle Mission Control at JSC. For example during STS-6, the Tracking and Data Relay Satellite (TDRS) was controlled by a control center at White Sands, New Mexico. In order to verify these control centers interfaces to the payload, an "end to end" test is usually part of Mode B. In this test payload telemetry is sent to the control centers and commands are sent from the control centers to the payload.

The other major test that is performed in Mode B is Mission Sequence. Most payloads have extensive mission sequencing of events that involve the interfaces between the orbiter and the payload. A good example of this is the deploy sequencing of PAM Payloads or IUS Payloads. In most cases, it is impossible to test the interfaces involved in this sequencing after the payload has been installed in the orbiter. This is usually due to installation of ordnance or lack of access to test points. In order to test these interfaces, a Mission Simulation is run which includes launch countdown, on-orbit checkout, deploy and post deployment operations. Items that are included in this testing are simulated ordnance firing, power transfers, and umbilical separations.

Mode C checkout is the culmination of all previous testing. Mode C is the type of payload to booster interface testing that has traditionally been performed at the launch site, i.e., post mate testing. Test time for cargo after orbiter mate means consuming time in the orbiter schedule. This time is expensive in terms of support dollars and is becoming less available as we try to reduce turnaround time. Both of the previous modes can be conducted in parallel with other orbiter activities. Mode A testing (orbiter with payload simulator) is generally a low level effort and can be conducted in parallel with other orbiter servicing. Mode B testing (payload with CITE) does not affect the orbiter schedule. By conducting these two previous modes we greatly decrease the risk that in Mode C (post cargo mate) testing we will discover major problems that will delay launch schedule. This saves real dollars in terms of test time and in preventing delays to the shuttle schedule.

In Mode C testing, the major objectives are functional verification of interfaces between the cargo and the orbiter, "end to end" testing with spacecraft control centers, stray voltage verification, and terminal count demonstration. Many of these tests were performed in Mode B testing and experience from this testing is usually very applicable to Mode C. In fact, it is often possible to verify solutions to problems discovered in Mode B during Mode C.

Several unique techniques have been developed to implement these three modes of testing. These techniques are, out of necessity, very fluid and are constantly changing to meet individual payload needs. Some of the more common techniques are described in following paragraphs.

MODE A CHECKOUT

Mode A checkout is performed in the Orbiter Processing Facility (OPF). Mode A checkout is performed after the orbiter has been configured for the mission. Configuring the orbiter for a specific mission usually consists of the following:

Installing Shuttle Mixed Cargo Harnesses (SMCH) Cables for Specific Cargoes

Installing Aft Flight Deck (AFD) Panels

Installing Payload Retention Fittings

Configuring Payload Patch Panel

Installing Interface Electronics

Loading Orbiter Flight Software with Telemetry/Decom Format Loads

Install Remote Manipulator System (RMS), if required

In Mode A checkout generic payload simulators are used for the following tasks:

Sending simulated payload telemetry to Payload Signal Processor, (PSP)/
Payload Data Interleaver, (PDI)/Payload Interrogator, (PI)/Payload
Recorder/Communications Interface Unit (CIU)

Sending commands via orbiter to simulated payload using PI/PSP/CIU/Multiplexer/Demultiplexer (MDM)

Verifying Payload timing buffer outputs to the simulated payload

Verifying Payload power voltage and current at Orbiter connect points

Exercise RMS electrical interface, if required

These generic payload simulators are basically racks of electronics capable of generating simulated payload telemetry streams and receiving payload commands and timing. They are programmable to generate/accept a wide variety of data types and rates. They are not designed as a specific payload simulator (e.g., a TDRS simulator) but rather are designed to checkout generic capabilities. These simulators are hooked into the orbiter at the same places that the payload will be connected. In this way, we are able to perform "copper path" verification of the mission specific cabling used to support the payload.

In addition to copper path testing, functional testing of orbiter support services is performed in the OPF. For example, functioning of the Payload Recorder is performed. Perhaps the most extensive functional testing performed is with the S-Band Payload RF System and the Payload Data Interleaver.

The OPF Communications and Tracking (C&T) Station is used to checkout the S-Band Payload System. Both command transmission via the S-Band Payload System and telemetry reception are checked out in this manner.

MODE B PAYLOAD TO ORBITER SIMULATOR - CITE

The foremost objective of testing a payload with an orbiter simulator like CITE is to minimize the number of surprises encountered by first time testing and mating of the payload and orbiter. To accomplish this objective, the Vertical Processing Facility, the Horizontal Processing Facility and the CITE Control Room were conceived. The major initial design considerations were to use the Orbiter to Payload interface control document in the design of the test stands, to use flight type avionics for the Pulse Code Modulation Master Unit (PCMMU), PDI, MDM, PSP, PI and to use a smaller Launch Processing Unit (LPS) type set for the CITE Control Room. These design considerations allowed procedures and software used to test the payload to be validated prior to orbiter to payload testing.

Since the original CITE did not have a General Purpose Computer (GPC), LPS was designed to include a "GPC Simulator" in CITE. This simulator controlled the acquisition of downlist, loading of the PCMMU, PDI, and the command interface to the payload for uplink and Launch Data Bus (LDB) commands. It in no way attempted to execute the GPC flight software. This design was more than adequate for OSTA, OSS and IUS pathfinder testing.

For OSTA processing, the original concept of processing a payload through CITE was to design an automated sequencer in the LPS which would be patterned after the mission scenario. This concept proved to be an error since the payload requirements and the mission scenario were very dynamic. The LPS software could not be modified as often as required or verified and validated in time for future payloads. A decision was made to return to more manual programs with simple display/command functions. Therefore, the Operation and Maintenance Instructions (OMI's) would control the testing and software redesign would not be necessary if the mission scenario changed. We also discovered the payload community and the flight crew wanted to use the GPC flight software to the maximum extent possible.

As payloads became more complicated and the role of the GPC flight software become more involved with the payload, it became apparent that a GPC, Display Electronics Unit (DEU) and Mass Memory Unit (MMU) were necessary to support payload testing in the CITE facility. A CITE Augmentation System was added which includes the GPC, DEU, MMU and an Eclipse Computer. The Eclipse computer simulates the GNC computer of the Orbiter (e.g., state vector and uplink command routing). By using a GPC in CITE, the CITE procedures could be transferred for use on the Orbiter. It was also determined that if flight software required modification as a result of CITE test, there would be enough time between CITE test and post orbiter/mate of the payload to incorporate the software change.

To support the payload for CITE test, the test stand is configured to support the payload in much the same manner as in preparing the Orbiter in Mode A. Concurrent with the hardware reconfiguration, the LPS Ground Software Development is performed. To validate the Ground Software and Flight Software compatibility, a test prior to payload installation is performed where the payload telemetry stream is simulated to match the Command and Data Annex. This is the first time all the software products are truly merged.

The major tests will be described under Mode C discussion since they are common to both CITE and post orbiter mate testing. The only unique payload test performed in CITE is the Mission Sequence Test. The purpose of this test is to perform a nominal mission flow using the flight data file and to the maximum extent possible exercise the GPC flight software. If no flight software problems are detected, this testing is not performed after the orbiter and payload are mated.

MODE C TESTING

Mode C testing is performed after cargo installation in the orbiter. For horizontally installed payloads, this activity is performed in the Orbiter Processing Facility (OPF). For vertically installed payloads, this activity is performed on the pad. This may be as late as two weeks before launch. Mode C testing consists primarily of functional interface testing, "end to end" testing with spacecraft control center, ordnance stray voltage testing, and terminal countdown demonstration.

After cargo installation and electrical connection to the orbiter, the first major activity is cargo/orbiter interface test. The major purpose of this test is to perform a functional test of the interfaces between the orbiter and cargo. The cabling which connects the orbiter equipment to the payload and the orbiter services should have been checked out in Mode A. The payload side of these interfaces should have been tested in Mode B. In Mode C, we basically perform the Mode B interface testing with the real orbiter.

We face a significant challenge in this area because most of the orbiter support to payloads is only available in on-orbit Orbiter flight software operational sequences (OPS). In some cases to test significant interfaces, the payload must also be moded to on-orbit flight software modes. The challenge we face is that the orbiter/cargo is still on the ground. Things that the orbiter/cargo are programmed to do on-orbit may not work on the ground or even worse may be hazardous to equipment and personnel. Consider the disastrous impacts if the orbiter decided to fire thrusters to correct its attitude or the upper stage decided it was time to separate.

In order to avoid this problem, we have developed several techniques. The central technique is called the split computer, Diagram 1. Under split computer technique, two orbiter computers are activated with different OPS. One computer is loaded with standard ground checkout software, designated GNC9. The other computer is loaded with on-orbit System Management software, designated SM2. The orbiter flight critical buses, which connect the orbiter computers to most of the flight control and other flight critical equipment, are assigned to the GNC9 computer. This effectively safes most of the orbiter because the GNC9 load will not perform autonomous subsystem commanding. The orbiter payload buses which connect the orbiter computers to the payload multiplexer-demultiplexer (MDM), PAM Sequence Control Assembly (SCA), PDI, and other payload equipment are assigned to the SM2 computer. The SM2 computer is thus able to communicate to the payload utilizing the on-orbit software.

There are several problems associated with this configuration. First, there is a significant amount of orbiter equipment attached to "payload" MDMs. Included in the SM2 Computer is "special processes" software. Special processes software consists of automated routines which perform housekeeping of flight systems. For example, the SM2 computer monitors temperatures in the hydraulic system and activates the hydraulic system and the hydraulic circulation pumps to keep the hydraulic fluid from freezing. These are definitely things which could cause havoc during ground test. In order to prevent this functioning, the applicable systems are safed by cockpit switches.

Another problem which occurs is that of downlink bandwidth. With normal vehicle control data being downlisted by the GNC computer, payload interface data being downlisted by the SM computer and Payload data being interleaved into the downlink by the PDI, we have several data sources all competing to fill the 128 kbit downlink bandwidth. The normal on-orbit downlink/downlist formats cannot be used because they do not have the kind of data required for ground operations. More specifically, most on-orbit formats have a very small window for the GNC downlist. The prelaunch checkout (GNC9) load has a large downlist (51.2 kbit) of data. At KSC, we like to monitor this data continuously because these systems are hazardous (e.g., hypergolic systems). The large prelaunch checkout downlist cannot fit into the small on-orbit window. In order to solve this problem, we have designed special downlink formats that have all of the critical downlink information but still have room for SM and the payload data. This is done mostly by decreasing sample rate of high frequency measurements and by specifically planning to drop certain kinds of special test measurements (e.g., range safety) in parallel with payload work.

The final challenge in developing this test configuration is to find a way to command both the orbiter and the payload. Some of the payloads have their own dedicated umbilicals through the orbiter T-0 umbilical (e.g., IUS/TDRS, PAM). This eases the command problems somewhat. Other payloads require use of the launch data bus (LDB) for command. This presents us with a dilemma. We need the launch data bus to control the orbiter flight critical systems as well as the payload. Normally at KSC we try to keep the two launch data buses assigned to only one computer at a time to prevent confusion in the

ground and flight systems. During STS-2 and 3 pallet checkout, we tried to pass the LDB between the GNC9 computer when we needed to command orbiter flight critical systems and the SM2 computer when we needed to command the payload. The result of this technique was a lot of confusion in the control room as to what computer was being sent commands. This is a bad situation.

To solve this problem for later flows, it was decided to split the launch data buses. One data bus is assigned to the GNC9 computer and the other to the SM2 computer. This presented us with some confusion in our ground system with regard to how to route commands. Rather than implement some routing software, it was decided to use two separate LPS sets in two different control rooms. Since the normal control room is designated Firing Room-1, the payload control room is designated Firing Room-X. This system has been used extensively for STS-5 and STS-6 with great success.

An additional fallout of this configuration is that the orbiter uplink can be used to command the payload - either by RF or hardwire. Destination codes of uplink messages can be set to the GNC or SM orbiter computer. The GNC machine receives the uplink messages from the Network Signal Processor (NSP) and sends the messages to the SM machine via the Inter-Computer Communication (ICC) bus, if required. At KSC, we normally use the LDB for most command purposes because it is more versatile than for the uplink for most ground testing. However, because the Orbiter community tends not to use the uplink for ground checkout, the payload community has begun to use it more and more. In hindsight, it makes more sense to use the uplink for payload checkout because payloads are designed to work on-orbit and there is extensive payload support in the orbiter uplink software. We still use the LDB for payload checkout but there is a definite trend to use more uplink commands by the payload community.

Stray Voltage Ordnance Testing simply consists of a demonstration that the orbiter systems do not induce voltages on cargo ordnance lines and vice versa. This is not an electromagnetic compatibility (EMC) test, since EMC testing is usually a design type test/analysis activity. We have performed some minor EMC testing in the Orbiter (Example: Wireless headset compatibility in crew cabin with IUS communications interface unit, CIU) but this is not normal. In stray voltage testing, we configure the orbiter and the payload into prelaunch and on-orbit conditions and perform voltage checks on ordnance lines. This usually involves activating a considerable amount of orbiter and payload equipment such as RF systems, payload bay floodlights, hydraulic circulation pumps, etc.

"End to end" testing is also performed after cargo mate. Initially (STS-2) this test was combined with Orbiter/Mission Control Center Compatibility Testing. As we have gained more experience with the orbiter, the degree of payload to payload control center testing has increased, however, MMC to orbiter testing has decreased. Thus, the payload control center "end to end" tests have been separated from the Orbiter/MCC testing.

These "end to end" tests have become extremely complex affairs. Diagram 2 shows the control center testing. No less than 11 separate centers were processing data. In fact, in one test we shipped data from the IUS to Sunnyvale, CA (SCF) to Greenbelt, MD (GSFC) to Houston, TX (JSC) and then to White Sands, NM (TDRS Control). This data criscrossed the country 3 times.

In "end to end" testing, we ship data from the payload to the user control center. The user control center also formats commands and ships them to the payload. This is a test of the payload, the data network, and the payload control centers.

The purpose of the terminal count demonstration test is to perform a dress rehearsal of the countdown. STS-6 was the first time the payload community was actively involved in the demonstration. This is because the IUS is very active in the terminal count. There are two critical command actions which must take place in the last 9 minutes before launch. AT T-5min30sec, the IUS Inertial Navigation System is commanded to free inertial mode (flight mode). This must be complete before Orbiter APU start at T-5min. It must be commanded as late as possible, however, because due to software/hardware limitations, the IUS navigation system has a limit of time on the ground in the free inertial state. At T-3min59sec, the IUS is switched from orbiter power to Airborne Support Equipment (ASE) batteries. This is to lower electrical power demand on orbiter main C power bus during ascent. The ASE batteries have a limited ground budget so it is important to go on these batteries as late as possible.

In order to properly integrate countdown activities with the orbiter and to train both the shuttle and payload launch teams, the payload functions were included into the STS-6 Terminal Countdown Demonstration Test. During this test, it was discovered that the time required to issue the IUS navigation to flight mode command and the delay until it was verified at the Checkout Station (COS) was very long. In fact, it was so long that the operator at the Checkout Station did not have enough time to call a hold prior to APU start. One of the most highly stressed rules at KSC is "You don't start APU's unless you're ready to go." This is because APU fuel is a limiting factor during any hold. Most important, if all of the APU fuel ground budget is used, it forces a several day reservicing and launch slip. Based directly on the Terminal Countdown Demonstration Test experience, we changed IUS ground software and procedures so that the IUS navigation command was moved back 30 seconds. This way, we

were assured that the IUS navigation mode was completed at T-5min30sec instead of just starting at T-5min30sec. This was a very important lesson to learn and it is possible that we avoided an inadvertent payload hold on the STS-6 launch.

We expect that the STS-6 cargo is the first of many cargoes to have terminal count involvement. Preliminary Centaur High Energy Upper Stage planning indicates the Centaur will have extensive terminal count involvement. The Terminal Countdown Demonstration Test will probably grow to be a very important payload/upper stage test in the future.

RESULTS

Now that we have presented a rather elaborate test program, the obvious question is "Is it worth it?" Based on experience so far, we have found that this program regularly detects problems that could cause launch or mission failures.

One of the major areas in which we have found problems in is flight software. In STS-5, 6 and 7 processing, we have found errors in the Command and Data Annex and the Master Measurement Data Bank (MMDB). Improper command or telemetry/decom formats were in the data base. This could have caused the wrong commands to be issued from MCCH/POCC or wrong interpretations of downlink telemetry. In STS-5 and 6, we also found that the flight software was improperly annunciating payload faults or displaying data improperly on the on-board CRT. In some of these cases, it was possible to make fixes to flight software before flight. In other cases, we were able to brief the crew to ignore certain alarms.

In a similar item, we discovered a CIU to Orbiter incompatibility during Mode A testing. The CIU rejected the GNC State Vector transfer from the orbiter the first time it was enabled. This caused the CIU to turn on a "GPC error" light on the CIU display panel. We found that if the crew simply cleared the first time alarm (by pushing a clear pushbutton) that it caused no further problems.

During STS-5 pad testing of a PAM, a Sequence Control Assembly (SCA) failed. It issued several inadvertent commands to the PAM and satellite. If this had happened in flight we might have damaged flight hardware. The unit was replaced with a redesigned unit to preclude the failure mode.

During the terminal countdown test for STS-6, we found an incompatibility between Shuttle launch procedures and the IUS. At T-11min,, the payload bay purge flow rate is lowered to lower the delta pressure across the quick disconnect on the T-0 umbilical. This is in preparation for "popping" the umbilical at liftoff. In terminal count demonstration test we found that the lower flow rate altered the thermal environment of the IUS Redundant Inertial Measurement Unit (RIMU). This alteration was sufficient to cause large drift in the calculated azimuth of the RIMU. If the first time this had happened had been launch countdown, we certainly would have scrubbed the launch. Having observed this in countdown demonstration test, we were able to research and explain it so it was not a concern on launch day.

A similar item was observed regarding the Orbiter to IUS timing interface. At approximately T-1hour, a final update of the on-board Greenwich Mean Time (GMT) in the Orbiter Master Timing Unit (MTU) is made. During Orbiter to Cargo interface testing (Mode C), we found that an update of the MTU caused the IUS to declare a timing fault and go onto its own internal time. The IUS reset back to orbiter time within a second of the update. This, however, caused a Launch Commit Criteria failure in the IUS ground computer. This would have caused significant problems on launch day if we had not been expecting it, perhaps even a scrub.

There are many other examples of problems we have found that have had an affect on mission success. The above examples are "typical" problems we find.

SUMMARY

We have described the three modes of checkout that have evolved at KSC and CCAFS for Payload to Orbiter Interface Checkout. Mode A consists of copper path checks of mission unique wiring and functional checks of orbiter supplied payload services such as the payload recorder and the S-Band Payload system on the orbiter. Mode B consists of checkout with the real payload and a high fidelity orbiter simulator. Typical testing that occurs in Mode B is CITE/Cargo interface test, "end to end" test with control centers, ordnance test and mission sequence test. Mode C testing is performed after cargo installation in the orbiter. This is similar to traditional booster to spacecraft integration testing, but much more extensive. A full Cargo/Orbiter Interface test, and "end to end" test, Ordnance test and a Terminal Countdown Demonstration test is performed.

These testing modes have found numerous problems with spacecraft processed at KSC and CCAFS to date. This system will continue to evolve as more complex payloads arrive at the launch base with increasingly compressed processing schedules.

PAYLOAD/MISSION	INTERFACE SUMMARY
STS-1/DFI PALLET INTEGRATED ENVIRONMENT CONTAMINATION MONITOR (IECM)	PAYLOAD MDM COMMAND INTERFACE
STS-2/OFFICE OF SPACE AND TERRESTRIAL APPLICATIONS (OSTA-1) PALLET	FLEX MDM INTERFACE FLIGHT SOFTWARE/ON-BOARD CRT DISPLAY ORBITER POWER DATA IN GPC DOWNLIST (SM GPC)/COMMAND BY S BAND PM
STS-3/OFFICE OF SPACE SCIENCE (OSS-1) PALLET	FLEX MDM INTERFACE FLIGHT SOFTWARE/ON-BOARD CRT DISPLAY ORBITER POWER DATA IN GPC DOWNLIST (SM GPC)/COMMAND BY S BAND PM PAYLOAD TELEMETRY BY S BAND FM RMS POWER/DATA INTERFACE
STS-4/DOD 82-1	CLASSIFIED
STS-5/TELESAT ANIK ON PAYLOAD ASSIST MODULE (PAM) AND SATELLITE BUSINESS SYTEM (SBS) ON PAM	SEQUENCE CONTROL ASSEMBLY (SCA) ON PAYLOAD DATA BUS PAYLOAD DATA INTERLEAVER (PDI) S BAND PAYLOAD SYSTEM (SIGNAL PROCESSOR/INTERROGATOR) ORBITER POWER FLIGHT SOFTWARE/AUTOMATIC SEQUENCING ON-BOARD CRT DISPLAY SUBSTANTIAL FUNCTION ON STANDARD SWITCH PANEL (SSP)
STS-6/TRACKING AND DATA RELAY SATELLITE (TDRS-a)/INERTIAL UPPER STAGE (IUS-1)	COMMUNICATIONS INTERFACE UNIT (CIU) TO PAYLOAD MDM ORBITER STATE VECTOR TRANSFER TO PAYLOAD FLIGHT SOFTWARE/ON-BOARD CRT PAYLOAD DATA INTERLEAVER (PDI) S BAND PAYLOAD SYSTEM (SIGNAL PROCESSOR/INTERROGATOR) ORBITER POWER POWER CONTROL PANEL AND STANDARD SWITCH PANEL TDRS COMMAND VIA ORBITER S BAND PM

ALL PAYLOADS HAD PAYLOAD RECORDER AND PAYLOAD TIMING BUFFER INTERFACE

TABLE 1 PAYLOAD INTERFACE SUMMARY

DIAGRAM 1   SPLIT COMPUTER TECHNIQUE

ORBITER
SUBSYSTEM
HARDWARE

FLIGHT
CRITICAL
BUS

GNC
GPC

ICC
UPLINK

SM
GPC

PAYLOAD
BUS

PAYLOAD

ORBITER
SUBSYSTEM
HARDWARE

NETWORK
SIGNAL
PROCESSOR

PCM MASTER UNIT

DOWNLINK

DOWNLINK

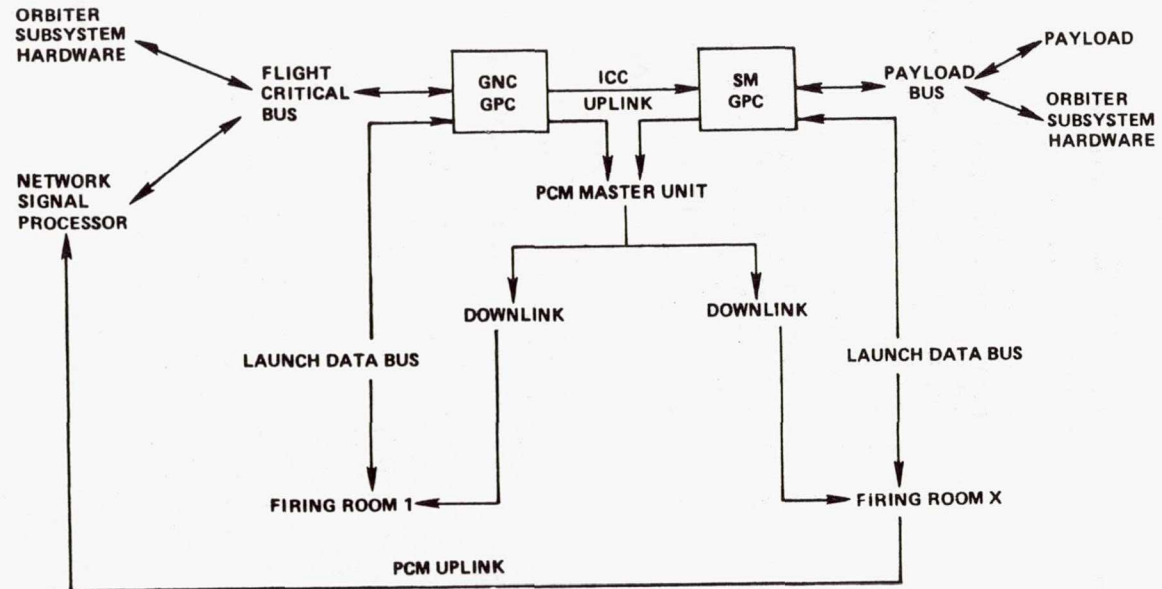LAUNCH DATA BUS

LAUNCH DATA BUS

FIRING ROOM 1

FIRING ROOM X

PCM UPLINK

DIAGRAM 2   STS–6 END TO END TEST (TDRS–A/IUS–1)

DOMSAT

USAF SATELLITE
CONTROL FACILITY
SCF–SUNNYVALE, CA

USAF REMOTE VEHICLE
CHECKOUT FACILITY
RVCF – CCAFS

MILA

SPACECRAFT TRACKING
AND DATA NETWORK
KSC, FLA

GODDARD OPERATIONS
GSFC – GREENBELT, MD

SHUTTLE MISSION CONTROL
JSC – HOUSTON, TX

TDRS WHITE SANDS GROUND TERMINAL
WSGT – WHITE SANDS, N.M.

TDRS BACKUP CONTROL
REDONDO BEACH, CA

IUS CHECKOUT STATION
COS – CCAFS, FL

TDRS SPACECRAFT CONTROL
SCC – CCAFS, FL

FIRING ROOM 1
KSC, FL

FIRING ROOM X
KSC, FL