CP593

C-17 Enhancements to a Flight Test Planning Program

Cheryl Canody
Tybrin Corporation
307 E. Popson Ave, Rm. 415
Edwards AFB CA 93524
Tel (805) 275-4579
Fax (805) 277-6054

Marle D. Hewett
G&C Systems, Inc.
30210 Rancho Viejo Rd, Suite D
San Juan Capistrano, CA 92675
Tel (714) 248-7212
Fax (714) 248-5237

Summary

This paper presents the enhancements incorporated into TEST_PLAN, a commercially available flight test planning program, for the C-17 follow-on test program. TEST_PLAN is a software package for UNIX and VMS based workstation computers that allows flight test engineers (FTE's) to plan and track flight test programs by mapping requirements to test points, flights, and flight test maneuvers.

TEST_PLAN is integrated with the Oracle relational database management system (RDBMS). Test points and requirements are stored in Oracle tables. The software provides automated tools that allows FTE's to query these tables to obtain lists of test points that can be assigned to maneuvers and flights to construct a flight test plan.

The 417th Flight Test Squadron (FLTS) at the Air Force Flight Test Center (AFFTC), Edwards Air Force Base (AFB), funded four major enhancements to TEST_PLAN to support the C-17 follow-on flight test program. These enhancements are:

1. incorporation of the C-17 Test Parameter Requirements (TPR) database into TEST_PLAN;

2. implementation of an Instrumentation Discrepancy Report in TEST_PLAN;

3. integration of FrameMaker with TEST_PLAN to implement an automated flight test card generation facility; and

4. instituting a requirements compatibility matrix using Oracle tables and compatibility definitions provided by the administrator.

The details of these enhancements are presented in this paper.

C-17 Test Parameter Requirements (TPR) database

Each test point has an associated list of instrumentation parameters, a TPR, that are needed for data collection and postflight processing. In TEST_PLAN, the TPR that will provide the necessary data is included in the test point definition when the test point is created. As the FTE plans a mission and assigns the test points within TEST_PLAN, the instrumentation engineer can perform an instrumentation check that will compile a list of all the parameters needed for the mission and check them against the aircraft instrumentation parameters. A system of color coded flags are used to alert the instrumentation engineer of any potential problems with the instrumentation in advance of the mission, allowing for issues to be addressed and remedied.

In addition to the TPR's that are assigned to test points, the instrumentation engineer has the capability to assign TPR's to the mission. This allows the instrumentation engineer to make sure that instrumentation that is necessary for all missions regardless of what the testing is, or for all missions for a certain discipline, is taken into account when the

Paper presented at the AGARD FVP Symposium on "Advances in Flight Testing", held in Lisbon, Portugal, 23-26 September 1996, and published in CP-593.

aircraft instrumentation is setup. Examples of this are safety-of-flight parameters for all missions or the parameters required for airdrop missions only.

All the TPR information is stored in a system of 37 Oracle tables, referred to as the Test Parameter DataBase (TPDB), that contain information such as the instrumentation measurand name, title, applicable software revision, bit configurations, calibration information, and special requirements. TEST_PLAN uses 6 of these tables to gather the TPR information related to each aircraft being tested. For the lists of measurands, the tables contain information such as the TPR name or number, applicable software revisions, list of measurands that make up the TPR, and the applicable aircraft. TPDB allows the number of unique TPR lists to grow as the aircraft is modified, where a TPR's uniqueness is based on either a software configuration update/change, discrete parameter bit changes, aircraft effectivity, or measurand bus location changes. The physical size of TPDB is limited only by the amount of storage space available on the system hosting the data and how much of it is allocated to TPDB.

The TPR data is entered by the responsible project engineer(s) after they have determined if there are any changes to existing measurands or TPRs or if there will be any new instrumentation added to the aircraft that must be defined. New instrumentation can be added to an existing TPR or a new TPR can be created. The project engineer coordinates with the aircraft instrumentation personnel on the accuracy of the information as well as to inform them of any new or changing requirements. After a TPR is created, the project engineer assigns it to a test point. At this point, the instrumentation engineer may perform an instrumentation check from within TEST_PLAN to verify the aircraft configuration will meet the desired testing requirements.

Prior to this capability becoming available from within TEST_PLAN, the instrumentation engineer was running printouts from the contractors TPDB system and manually verifying that the aircraft configuration would support the desired testing. The time required for instrumentation configuration verification has gone from days to hours, a significant saving in man hours, and has resulted in more accurate configuration checkouts.

Instrumentation Discrepancy Report

As part of the TPR process, a report was needed to assist the instrumentation personnel in properly preparing the aircraft for a mission. After the TPR interface with TEST_PLAN was created, the capability to print the results of the instrumentation check was added. An FTE or instrumentation engineer can run an instrumentation check of the TPRs needed for the mission against the current configuration of the aircraft. Then they can then print an Instrumentation Discrepancy Report (IDR) that lists which parameters are unavailable and for what reason or print the entire list of parameters needed for a mission. Figure 1 shows a sample IDR.

To do the IDR, TEST_PLAN uses an additional Oracle table where instrumentation deficiencies are recorded and tracked through a component of TPDB called Report of Unsatisfactory or Defective Instrumentation (RUDI). Instrumentation that is broken or malfunctioning is entered into the RUDI tables. When it is repaired the entry is closed with a description of the fix. When the Instrumentation Discrepancy Report is run by the instrumentation engineer, TEST_PLAN checks the measurands in the TPR against the RUDI table and annotates any that are listed there. This gives the instrumentation engineer advance notice of measurands that may hold up the testing if the measurands' malfunction is not repaired.

TEST_PLAN creates the IDR by pulling the necessary TPR information from the TPDB tables based on the TPR's specified in the flight plan. First, a list of the measurands/parameters necessary to support the test points is created. This list is then compared to an ASCII file that contains the current aircraft configuration with regard to what parameters are available on the aircraft, what sample rate the parameter is available at, what post flight data stream the parameter will be available on, and what telemetry stream, if any, the parameter will be on. When all the parameters in the TEST_PLAN generated list and the ASCII file have been reconciled, TEST_PLAN then checks the parameters against the data in the RUDI table to verify whether the parameter is functioning or not. All this information is then compiled and presented in the IDR with potential problems flagged.

From this report, the instrumentation engineer can start resolving issues so particular test points can be flown, coordinating his/her efforts with the discipline engineer who will need the data.

Since the incorporation of this feature, the time used to obtain these reports has gone from an average of 1 to 2 hours to a matter of minutes. This allows the instrumentation personnel to get to work on the problems indicated in the IDR sooner and run additional IDR's quickly as the aircraft configuration changes due to repairs, installation of additional equipment, or changes to the test points planned for the mission. The time that was spent waiting on reports is now spent getting the aircraft ready sooner, allowing for a faster turn around of the aircraft between missions. Since the discipline engineers also have the capability to run IDR's, they can stay abreast of the aircraft instrumentation setup and not get caught unprepared for a mission.

Previously, this process was performed on the contractors TPDB system by running several different Oracle SQL scripts, printing the results, and manually verifying problems. This could take several hours to get the printouts and identify the instrumentation discrepancies. With the IDR capability provided by TEST_PLAN, the instrumentation engineer has the results in as little as five (5) minutes to as much as thirty (30) minutes, a considerable saving of man hours. The IDR is also more accurate than the previous partially manual method, increasing flight safety - always a desired result.

Automated Flight Card Generation Facility

TEST_PLAN presents flight plans as a sequence of flight test maneuvers with test points assigned to these maneuvers. It is possible to automatically generate a deck of flight cards from this sequence. In the standard commercial version, TEST_PLAN generates these cards using imbedded postscript code with fixed formats that cannot be changed by the FTE. This automatic flight card generation facility has the potential of generating considerable savings in person hours in a flight test program. These savings are realized at a time when FTE's are usually working overtime to generate flight card decks for the test program. The time savings translates to a general improvement in flight safety during the test program because the FTE's have more time to think about flight plans instead of spending time generating cards using a word processor or a form generation software package.

Even though an imbedded flight card generation facility was available in TEST_PLAN, it was tailored to the knee board cards used by fighters and did not lend itself well to the 8 1/2 x 11 format used by the C-17. The enhancements funded by the 417th FLTS for the C-17 program provided for the integration of FrameMaker, a commercially available document generation and management package, with TEST_PLAN. This integration allowed for the generation and use of a large number of flight card formats generated by a local TEST_PLAN administrator in FrameMaker. This capability was used to provide all the existing templates in use and add additional ones as desired.

To generate a flight card deck, FTE's enter and save flight card data by test point in an Oracle table. The flight card database table is a separate table from the test point table, to which the FTE's have read-only privileges. The FTE's have read and write privileges to the flight card table. The FTE then plans a flight in TEST_PLAN by assigning test points to a sequence of flight test maneuvers. At any point during the planning process, the FTE may generate a deck of flight cards by selecting an appropriate menu item with the flight highlighted in TEST_PLAN. TEST_PLAN creates a FrameMaker file in a data interchange format and starts FrameMaker automatically for the user the first time a card deck is built during a planning session. FrameMaker comes up with the appropriate cover page displayed followed by the flight cards as sequenced in the flight. At that point the FTE is free to modify the cards as necessary in FrameMaker and print the deck. As long as the FTE does not quit FrameMaker, it will be available for viewing any other cards that are created or worked on during the session. TEST_PLAN stays aware of whether FrameMaker is running or not and if it is still running will notify the user of such, instructing them on what file to open for viewing. If FrameMaker has been exited, TEST_PLAN will restart it, opening the card file being worked on automatically.

In addition to test points that may be associated with a particular type of maneuver, TEST_PLAN also allows the FTE to assign up to 25 test points to the actual flight block itself. This provides the FTE with the flexibility to include points that may apply for the whole mission or maybe even a portion of a mission without having to try and associate it with a particular maneuver at a particular point in the flight. When all points have been assigned to the flight, either in the flight block or the maneuvers, the FTE may generate the deck for the flight. When doing this, the points assigned to the flight block are first in the deck followed by the points in the maneuvers in the order they were assigned. Each deck is preceded by one or more cover pages that list the test points in the order they are assigned to the mission. Figure 2 presents a typical C-17 flight deck cover page. There are currently four standard cover pages - Ground, Low hazard, Medium hazard, and High hazard. TEST_PLAN determines which cover page to use based on the test points defined hazard levels. Whatever the highest hazard level is in the flight plan is the cover page TEST_PLAN will use. For example, a mission may have 23 Low hazard points, 2 Ground points, and 1 Medium hazard test point. Even though there is only one Medium point, that is the cover page that will be used so as to reflect the highest hazard condition of the mission. By looking at the cover page, one can see what hazard level the points are and see which point(s) make the mission that hazard level.

The integrated package provides the FTE with a time saving device that frees him/her to spend more time in planning and less time in flight card creation. TEST_PLAN also allows an FTE to reuse previously planned missions using cut and paste functions, saving the FTE more time in setting up missions and assigning test points. After completing the changes to a mission plan that the FTE is reusing, the flight deck for the new mission can be created and printed. This is an often used feature for regression testing or for build-up testing where the general mission plan is either the same or very similar to an existing plan. Figure 3 presents a typical C-17 flight test card for a test point.

Requirements Compatibility Matrix

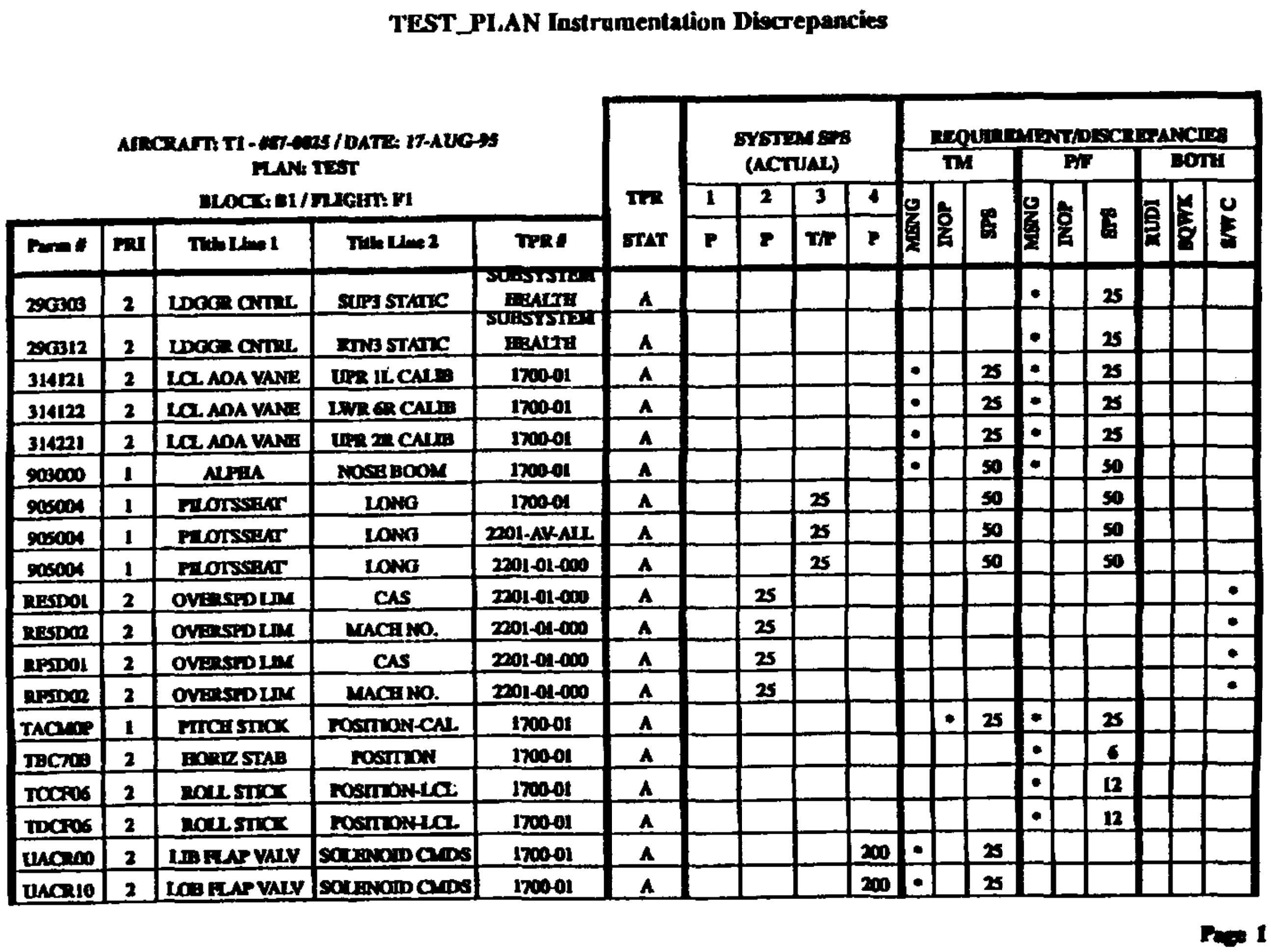
When planning a mission, the FTE must also take into account the aircraft system and support requirements needed to accomplish all desired testing. This often requires consultation with the cognizant engineers to ascertain what support is needed from what organizations and whether the requirements are compatible with each other. Some are obvious - a dry runway requirement and a wet day requirement are incompatible, but others are not so obvious. TEST_PLAN allows users to define ten categories of system configurations, eight categories of support requirements, and a loads configuration category. A typical set of systems configurations might be: flight controls, communications, hydraulics, weapons, navigation, and pneumatic. A typical set of support requirement categories might be: weather, operating area, air support, and ground support. In each system and support category, an FTE can build lists of requirements; each list being composed of items stored in an Oracle table. This table contains all systems and support items in all categories and each item's compatibility (or incompatibility) with all other items in the same category. Once each item is defined, it can be combined into lists with other items in the same category. These lists can then be selected and assigned as part of the test point definition process to indicate what support is required for the point to be accomplished. Figure 4 illustrates a typical compatibility list with compatible items indicated by highlighting.

After the requirements' information has been assigned to the test point, the FTE will get an automatic indication of a conflict of requirements, via color coded flags, when points are assigned to a mission within TEST_PLAN. Then, using the Requirements/Limit violations display available in TEST_PLAN, the FTE can see what the conflict is and make a decision on what point to remove, based on testing priorities, to resolve the conflict. Figure 5 illustrates the Planning Display with test points assigned that have load, systems, and supports requirements incompatibilities. The LOAD, SYSTEMS, and SUPPORTS requirements flags are set to indicate the compatibility problems to the FTE.

The Requirements Compatibility Matrix removes the guess work in determining what support items are needed and their compatibility with each other for the FTE, enabling the FTE to plan missions devoid of conflicts. This feature results in time savings for the FTE in planning the mission and for the cognizant engineer by removing the need of coordinating with the FTE to ensure requirements are met. It also allows for an FTE that may not be intimately familiar with a discipline to plan missions competently, safely, and quickly without having to spend the time to come up to speed on the discipline or having the engineer spend a larger amount of time assisting in the planning of the mission.

Conclusion

In an effort to enhance the quality of flight testing and enable a smaller force to perform the testing, the 417th FLTS invested in TEST_PLAN to provide automated tools for FTE's to construct efficient, safe, and reusable flight test plans in less time. In addition to the generic features available in the software, the 417th also invested in customization work that automated and standardized tasks that were previously done manually or with limited automation. This investment has already begun to reap benefits in reduced man hours to perform these tasks and further benefits are expected as users become more familiar with the system.



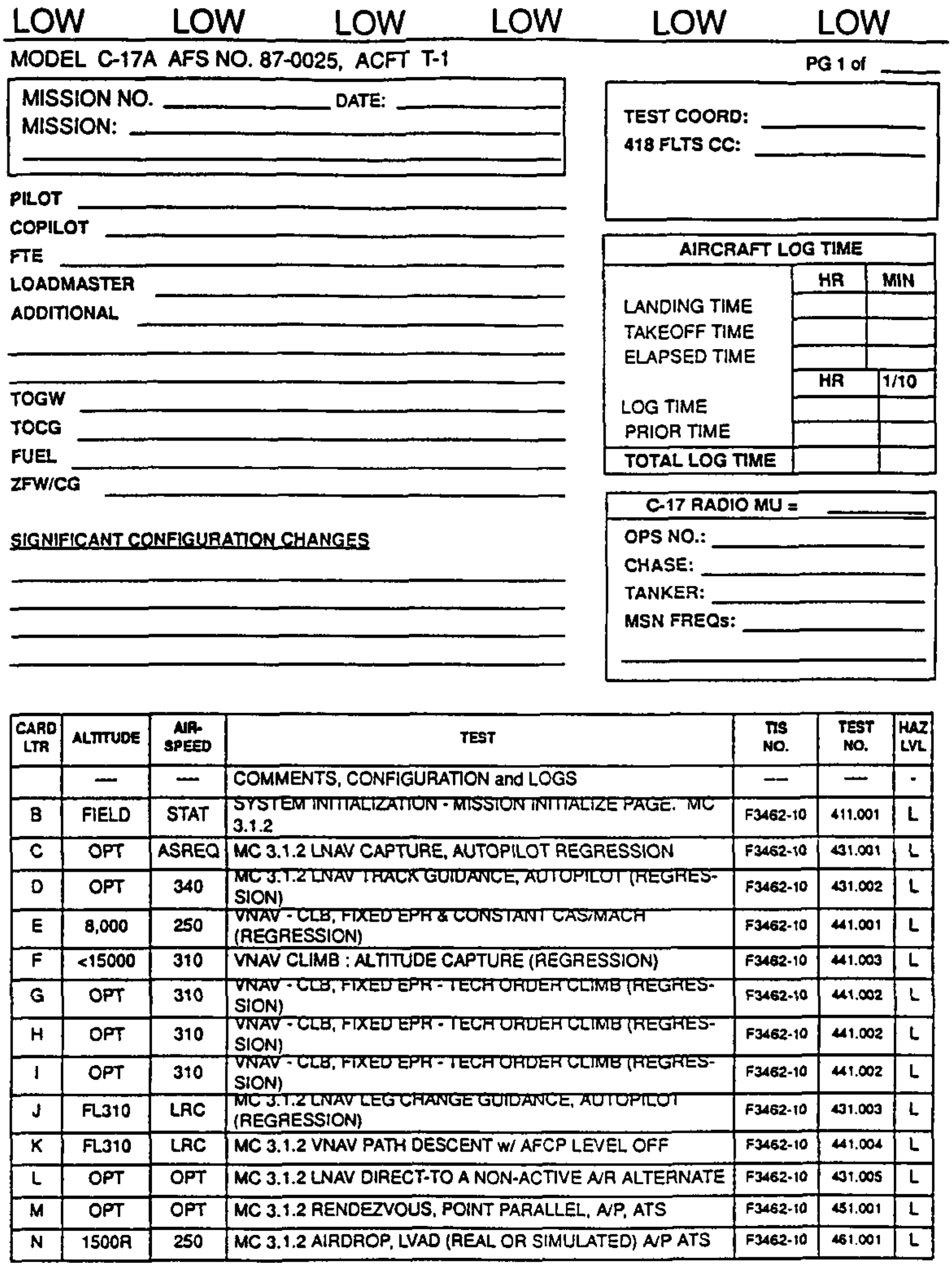
TEST_PLAN Instrumentation Discrepancies

AIRCRAFT: T1 - #87-0025 / DATE: 17-AUG-95
PLAN: TEST
BLOCK: B1 / FLIGHT: F1

						SYSTEM SPS (ACTUAL)				REQUIREMENT/DISCREPANCIES								
										TM			P/F			BOTH		
					TPR	1	2	3	4	MSNG	INOP	SPS	MSNG	INOP	SPS	RUDI	BQWK	S/W C
Param #	PRI	Title Line 1	Title Line 2	TPR #	STAT	P	P	T/P	P									
29G303	2	LDGGR CNTRL	SUP3 STATIC	SUBSYSTEM HEALTH	A								•		25			
29G312	2	LDGGR CNTRL	RTN3 STATIC	SUBSYSTEM HEALTH	A								•		25			
314121	2	LCL AOA VANE	UPR 1L CALIB	1700-01	A					•		25	•		25			
314122	2	LCL AOA VANE	LWR 6R CALIB	1700-01	A					•		25	•		25			
314221	2	LCL AOA VANE	UPR 2R CALIB	1700-01	A					•		25	•		25			
903000	1	ALPHA	NOSE BOOM	1700-01	A					•		50	•		50			
905004	1	PILOTSSEAT	LONG	1700-01	A			25				50			50			
905004	1	PILOTSSEAT	LONG	2201-AV-ALL	A			25				50			50			
905004	1	PILOTSSEAT	LONG	2201-01-000	A			25				50			50			
RESD01	2	OVERSPD LIM	CAS	2201-01-000	A		25											•
RESD02	2	OVERSPD LIM	MACH NO.	2201-01-000	A		25											•
RPSD01	2	OVERSPD LIM	CAS	2201-01-000	A		25											•
RPSD02	2	OVERSPD LIM	MACH NO.	2201-01-000	A		25											•
TACM0P	1	PITCH STICK	POSITION-CAL	1700-01	A						•	25	•		25			
TBC70B	2	HORIZ STAB	POSITION	1700-01	A								•		6			
TOCF06	2	ROLL STICK	POSITION-LCL	1700-01	A								•		12			
TDCF06	2	ROLL STICK	POSITION-LCL	1700-01	A								•		12			
UACR00	2	LIB FLAP VALV	SOLENOID CMDS	1700-01	A				200	•		25						
UACR10	2	LOB FLAP VALV	SOLENOID CMDS	1700-01	A				200	•		25						

Page 1

Figure 1



LOW LOW LOW LOW LOW LOW

MODEL C-17A AFS NO. 87-0025, ACFT T-1

PG 1 of ____

MISSION NO. ______________ DATE: ______________
MISSION: ______________________________
____

TEST COORD: ______________
418 FLTS CC: ______________

PILOT ______________________________
COPILOT ______________________________
FTE ______________________________
LOADMASTER ______________________________
ADDITIONAL ______________________________
____
____

TOGW ______________________________
TOCG ______________________________
FUEL ______________________________
ZFW/CG ______________________________

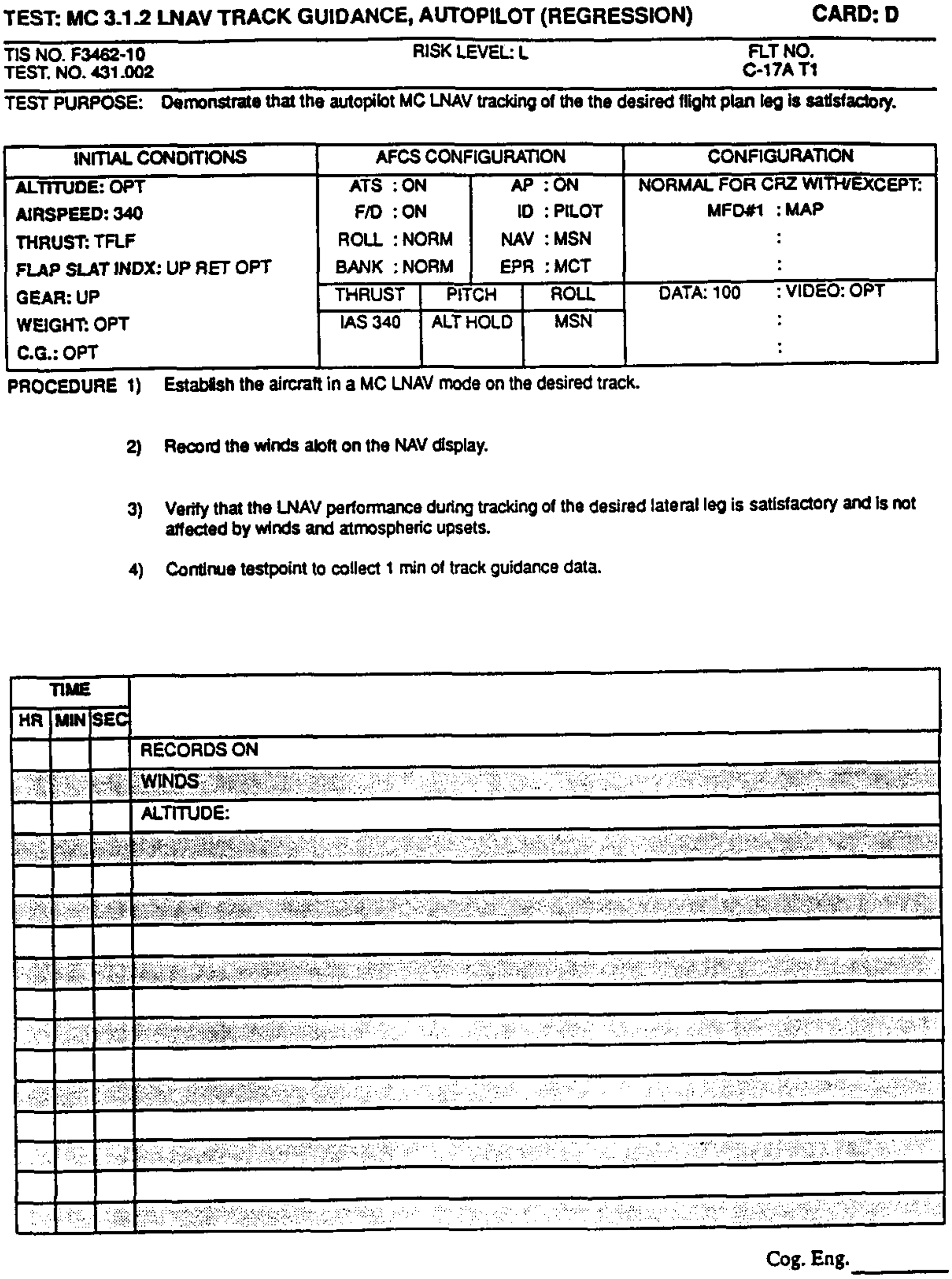
AIRCRAFT LOG TIME		
	HR	MIN
LANDING TIME		
TAKEOFF TIME		
ELAPSED TIME		
	HR	1/10
LOG TIME		
PRIOR TIME		
TOTAL LOG TIME		

SIGNIFICANT CONFIGURATION CHANGES
____
____
____
____

C-17 RADIO MU = ______
OPS NO.: ______________
CHASE: ______________
TANKER: ______________
MSN FREQs: ______________
____

CARD LTR	ALTITUDE	AIR-SPEED	TEST	TIS NO.	TEST NO.	HAZ LVL
	—	—	COMMENTS, CONFIGURATION and LOGS	—	—	-
B	FIELD	STAT	SYSTEM INITIALIZATION - MISSION INITIALIZE PAGE. MC 3.1.2	F3462-10	411.001	L
C	OPT	ASREQ	MC 3.1.2 LNAV CAPTURE, AUTOPILOT REGRESSION	F3462-10	431.001	L
D	OPT	340	MC 3.1.2 LNAV TRACK GUIDANCE, AUTOPILOT (REGRESSION)	F3462-10	431.002	L
E	8,000	250	VNAV - CLB, FIXED EPR & CONSTANT CAS/MACH (REGRESSION)	F3462-10	441.001	L
F	<15000	310	VNAV CLIMB : ALTITUDE CAPTURE (REGRESSION)	F3462-10	441.003	L
G	OPT	310	VNAV - CLB, FIXED EPR - TECH ORDER CLIMB (REGRESSION)	F3462-10	441.002	L
H	OPT	310	VNAV - CLB, FIXED EPR - TECH ORDER CLIMB (REGRESSION)	F3462-10	441.002	L
I	OPT	310	VNAV - CLB, FIXED EPR - TECH ORDER CLIMB (REGRESSION)	F3462-10	441.002	L
J	FL310	LRC	MC 3.1.2 LNAV LEG CHANGE GUIDANCE, AUTOPILOT (REGRESSION)	F3462-10	431.003	L
K	FL310	LRC	MC 3.1.2 VNAV PATH DESCENT w/ AFCP LEVEL OFF	F3462-10	441.004	L
L	OPT	OPT	MC 3.1.2 LNAV DIRECT-TO A NON-ACTIVE A/R ALTERNATE	F3462-10	431.005	L
M	OPT	OPT	MC 3.1.2 RENDEZVOUS, POINT PARALLEL, A/P, ATS	F3462-10	451.001	L
N	1500R	250	MC 3.1.2 AIRDROP, LVAD (REAL OR SIMULATED) A/P ATS	F3462-10	461.001	L

Figure 2



TEST: MC 3.1.2 LNAV TRACK GUIDANCE, AUTOPILOT (REGRESSION) **CARD: D**

TIS NO. F3462-10 RISK LEVEL: L FLT NO.
TEST. NO. 431.002 C-17A T1

TEST PURPOSE: Demonstrate that the autopilot MC LNAV tracking of the the desired flight plan leg is satisfactory.

INITIAL CONDITIONS	AFCS CONFIGURATION			CONFIGURATION	
ALTITUDE: OPT	ATS : ON	AP : ON		NORMAL FOR CRZ WITH/EXCEPT:	
AIRSPEED: 340	F/D : ON	ID : PILOT		MFD#1	: MAP
THRUST: TFLF	ROLL : NORM	NAV : MSN			:
FLAP SLAT INDX: UP RET OPT	BANK : NORM	EPR : MCT			:
GEAR: UP	THRUST	PITCH	ROLL	DATA: 100	: VIDEO: OPT
WEIGHT: OPT	IAS 340	ALT HOLD	MSN		:
C.G.: OPT					:

PROCEDURE 1) Establish the aircraft in a MC LNAV mode on the desired track.

2) Record the winds aloft on the NAV display.

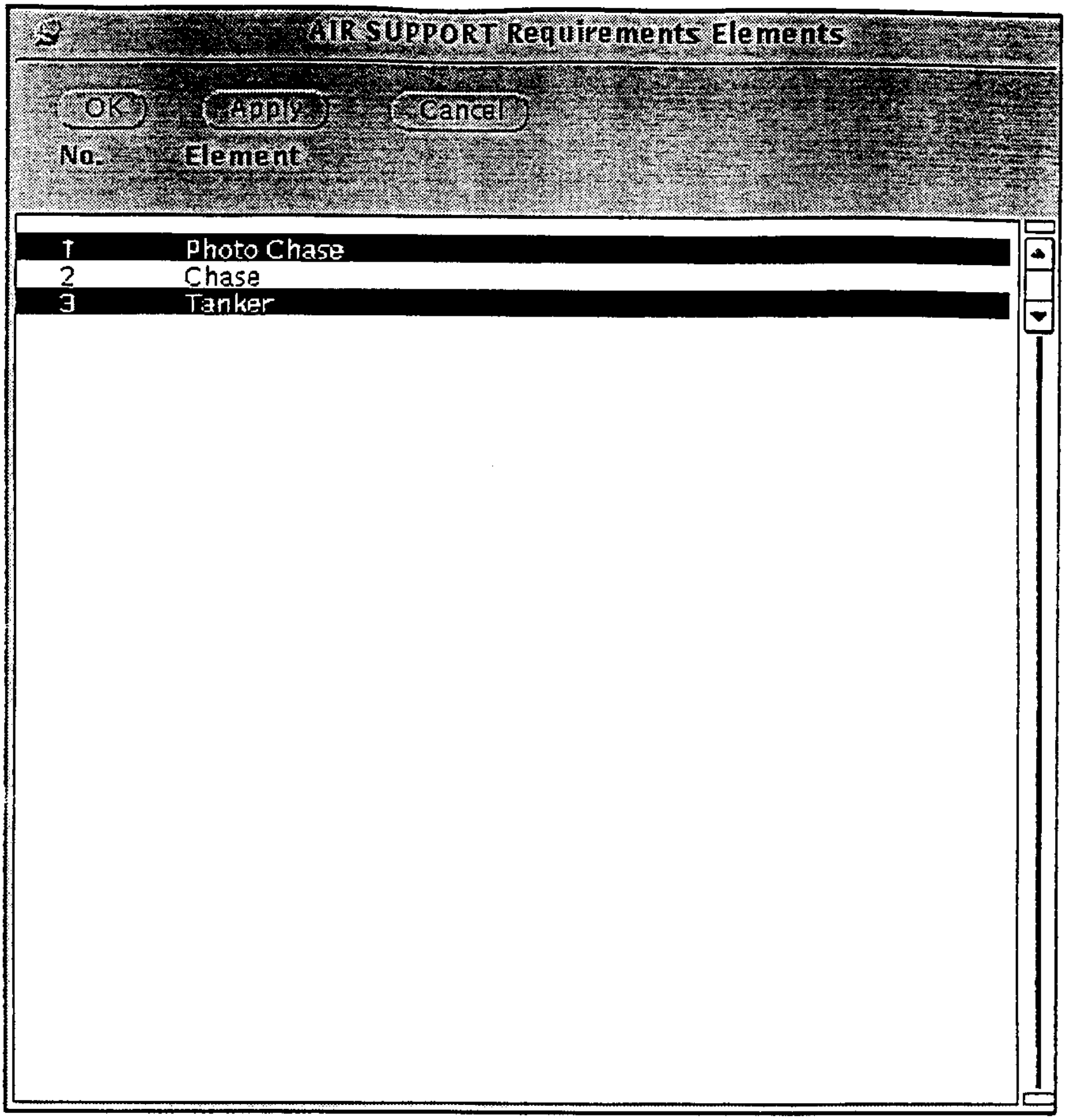
3) Verify that the LNAV performance during tracking of the desired lateral leg is satisfactory and is not affected by winds and atmospheric upsets.

4) Continue testpoint to collect 1 min of track guidance data.

TIME			
HR	MIN	SEC	
			RECORDS ON
			WINDS
			ALTITUDE:

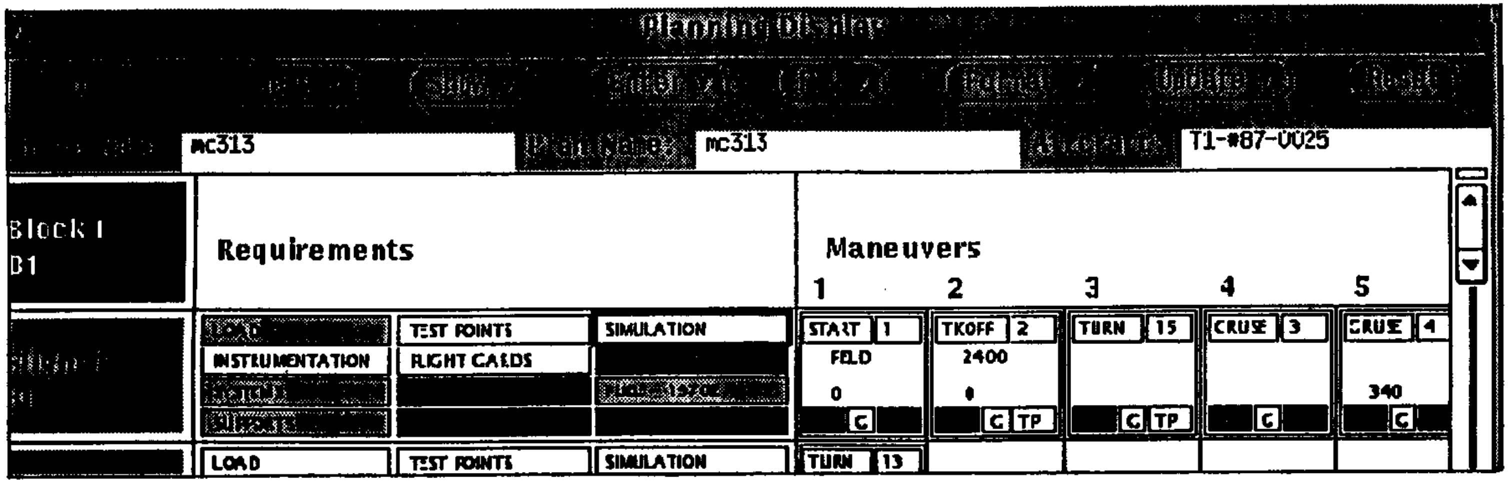
Cog. Eng.________

Figure 3



No.	Element
1	Photo Chase
2	Chase
3	Tanker

Figure 4


Planning Display
Show
Update
Reset
mc313
Plan Name:
mc313
T1-#87-V025
Block I
B1
Requirements
LOAD
TEST POINTS
SIMULATION
INSTRUMENTATION
FLIGHT CARDS
Maneuvers
1
2
3
4
5
START 1
FELD
0
C
TKOFF 2
2400
C TP
TURN 15
C TP
CRUSE 3
C
CRUSE 4
340
C
LOAD
TEST POINTS
SIMULATION
TURN 13