COS DCE BOOT FSW v1.09 Component Test Results
Requirement 5.2.3.2 Echoes for Command Opcode and Parameters

Date:	February 13, 2001
Document Number:	COS-03-0026
Revision:	Initial Release
Contract No.:	NAS5-98043
CDRL No.:	N/A

Prepared By: _____ _____
Tim Swanson, Software Test Engineer, Design_Net Eng. Date

Reviewed By: _____ _____
K. Brownsberger, COS Sr. Software Scientist, CU/CASA Date

Reviewed By: _____ _____
Grant Blue, COS Software & Operations Manager, BATC Date

Approved By: _____ _____
Barry Welsh, FUV Detector Program Manager. UCB Date

Approved By: _____ _____
John Andrews, COS Experiment Manager, CU/CASA Date

Center for Astrophysics & Space Astronomy
University of Colorado
Campus Box 593
Boulder, Colorado 80309

REVISIONS

Letter	ECO No.	Description	Check	Approved	Date
-		Initial Release			

Original Release Name	Date
Drawn: K. Brownsberger	2-13-01
Reviewed:	
Approved:	

THE UNIVERSITY OF COLORADO
At Boulder
The Center for Astrophysics and Space Astronomy

COS DCE BOOT FSW v1.09 Component Test Results
Requirement 5.2.3.2 Echoes for Command Opcode and Parameters

Size	Code Indent No.	Document No.	Rev
A		COS-03-0026	-
Scale: N/A			

Table of Contents

1. INTRODUCTION

1.1 PURPOSE

This document presents the Cosmic Origins Spectrograph (COS) Device Control Electronics (DCE) Flight Software (FSW) certification procedure. The purpose of this procedure is to verify that the FSW satisfies Software Requirements according to the method specified in the DCE FSW Test Plan (STP).

1.2 SCOPE

This test procedure comprises the steps necessary to verify that the FSW satisfies Software Requirements Document (SRD) paragraph 5.2.3.2 — Echoes for Command Opcodes and Parameters.

1.3 LIMITATIONS AND CONSTRAINTS

This test cannot be run in parallel with any other commanding activity directed at the DCE FSW (such as, for example, the periodic transmission of NOOP commands). Test hardware shall be visually inspected, and its configuration noted, prior to conducting this test.

1.4 PROCEDURE OVERVIEW

The procedure requires the **hks** tools running on the Sun SparcStation Electronic Ground Support Equipment (EGSE) whose network IP address is one of

shorty.ssl.berkeley.edu
taiyo.ssl.berkeley.edu
ginger.ssl.berkeley.edu.

Test time shall be scheduled in advance. The Test Conductor must be logged into the Unix system as user **eagcos**, and be commanding from the appropriate directory. This directory contains both the test script file and the shell script file; these two files control test execution. The test is conducted by invoking the shell script. This shell script in turn invokes the Perl 5 program **UniScript.pl**, which resides in its own distinct directory. The test procedure steps have been pre-recorded in the test script file, and are executed interpretively by the **UniScript** program. The shell script and test script are attached to this document as appendices. As **UniScript** executes the test script it sends results to the operator console and to two report files, which are also placed in the current directory. After completion of the test script, the Test Conductor can certify successful test

execution by examining the contents of the report files and determining that required outputs are present in them. Printed copies of the report files are attached to the manually completed checklist (Paragraph 4 below) as documentation of the test.

1.5 THEORY OF TEST

The script synthesizes an **LFDNOOP** command whose five parameters, rather than being 0x0000, as in a normal no-op, are, respectively, 0x0010, 0x0020, 0x0030, 0x0040, 0x0050. This syntactically correct (but semantically deviant) command is uplinked to the DCE. The HK data from this command are examined: the image, **LFDCBUF**, is compared, word-by-word, with the contents of Buffer 1, into which the deviant no-op was generated.

1.6 TEST SCRIPT IMPLEMENTATION

1.6.1 Test Script Arguments

The script is not parameterized.

1.6.2 Test Script Coding

The script uses standard **UniScript** commands and directives.

2. SPECIAL INSTRUCTIONS

2.1 QUALITY ASSURANCE

QA support is required to verify the configuration and setup environment as well as monitoring test steps and verifying results.

2.2 SAFETY

2.2.1 Personal Safety

To ensure the safety of the test personnel during test execution the guidelines contained in Paragraph 3.4, Reference [1] will be adhered to.

2.2.2 Test Article and Equipment Safety

- If access within one (1) meter of COS bench electronics is necessary, wrist straps attached to technical ground shall be used by all personnel involved in handling of any COS test article. Overcurrent and overvoltage shall be set to remove power if nominal limits are exceeded.

- Emergency Power Shutdown — If, during the COS DCE FSW test, power is ON and a severe test equipment failure results in the power system exceeding specified limits, the Test Conductor shall direct or perform shutdown of power.

2.3 CONTAMINATION

All flight hardware shall be handled with clean latex gloves; it shall be covered with clean ESD material and/or stored in a clean flow-bench.

3. SUPPORT REQUIREMENTS

3.1 PERSONNEL

Execution of the COS DCE FSW certification procedure requires the following personnel (to be completed at the Test Readiness Review (TRR):

Test Director: _____

Test Conductor: _____

Test Technician: _____

QA: _____

3.2 TOOLS, EQUIPMENT, AND MATERIALS

The following is a list of tools, equipment, or materials required in this test. Record manufacturer and model, metrology, or property numbers of equipment used, where appropriate. Record calibration due dates where appropriate.

Boot Mode ROM: schematic **27C256**

Engineering Ground Support Equipment (see paragraph 1.4). Indicate specific configuration:

EGSE			DCE		
taiyo	**shorty**	**ginger**	**ETU**	**DCE #1**	**DCE #2**
	X			X	

3.3 DATA/SOFTWARE

The following files must be present:

Table 3-1: Required Program and Data Files

EGSE (shorty) Directory	File	Description
\disks\galex\users\galex\tcs\uniscript	**UniScript.pl**	**UniScript** interpreter
\disks\galex\users\galex\tcs\uniscript\stp5_2_3_2	**u**	Shell script for this procedure
Ditto	**stp5_2_3_2.tst**	Test script for this procedure (Appendix B)

In addition, the **hks** tools must be active. Directions for activating **hks** are given in UCB-COS-DOC-1118 (Paragraph 3.4, Reference [4]).

3.4 REQUIRED DOCUMENTATION

Reference	Document Number	Title
1	NHB 1700.1(V1-A)	*NASA Basic Safety Manual*
2	COS-03-0026	*DCE FSW Test Procedure 5.2.3.2* (this document)
3	UCB-COS-008	*COS FUV Detector Software Test Plan*
4	UCB-COS-DOC-1118	*COS EGSE Startup Procedure*

4. PROCEDURE/TASK STEPS

4.1 PRE-OPERATION ACTIVITIES

4.1.1 Make Sure that **hks** Tools Are Active

Follow the procedure given in Paragraph 3.4, Reference [4].

4.1.2 Make Sure that the Proper ROM Is Installed

Visually verify that the ROM under test is installed: if EEPROM, in U18: if PROM, in U2 and U7.

4.1.3 Log In to the EGSE

Step	QA	Operator Entry/System Response	Description
1		C:\tcs\us> **telnet shorty.ssl.berkely.edu**	Establish connection to shorty via Telnet client program
2		Login: **tcs** Password:	Using telnet window, login as user **tcs**

4.1.4 Set Current Directory

Step	QA	Operator Entry/System Response	Description
3		tcs@shorty% **cd ~galex/tcs** tcs@shorty% **pwd** /disks/galex/users/galex/tcs	Change current directory as shown

4.1.5 Slogin as eagcos

Step	QA	Operator Entry/System Response	Description
4		tcs@shorty% **slogin –l eagcos shorty.ssl.berkeley.edu** eagcos@shorty.ssl.berkeley.edu's password: ***(get from SSL personnel)*** Last login: Sat Oct 7 10:41:05 2000 from auntem.ssl.berke Sun Microsystems Inc. SunOS 5.8 Generic February 2000 You have mail. COS EGSE software version: devel	slogin as **eagcos**; get password from SSL personnel

4.1.6 Set Current Directory

Step	QA	Operator Entry/System Response	Description
5		eagcos:shorty% **cd /disks/galex/users/galex/tcs/uniscript/stp5_2_3_2** eagcos:shorty% **pwd** /disks/galex/users/galex/tcs/uniscript/stp5_2_3_2	Change current directory as shown

4.1.7 Ensure that Proper Files are Present

Step	QA	Operator Entry/System Response	Description
6		eagcos@shorty% **ls –l** Total 12 -rw-r--r-- 1 tcs eag 1398 Oct 8 18:03 stp5_2_3_2.tst -rw-r--r-- 1 tcs eag 62 Oct 9 17:44 u	List files; the **.tst** file and the shell script **u** should be present

4.2 OPERATION EXECUTION

4.2.1 Establish Initial Test Conditions

Step	QA	Operator Entry/System Response	Description
7		eagcos:shorty% **set path=($path ~dbb/scripts/bin)**	Set path as shown to enable access to hks tools

4.2.2 Execute the Script

Step	QA	Operator Entry/System Response	Description
8		eagcos:taiyo% **more < u** #!/bin/sh pkill cosnoopy perl ../UniScript.pl stp5_2_3_2 "0,0,0,0,0,0,0,0" cosnoopy& eagcos:taiyo% **sh u** $pstring=0,0,0,0,0,0,0,0 Parameters are: Script File: stp5_2_3_2 #0: 0 #1: 0 #2: 0 #3: 0 #4: 0 #5: 0 #6: 0 #7: 0 Report file >/disks/galex/users/galex/tcs/uniscript/stp5_2_3_2/stp5_2_3_2.rp1	Shell to **u**. You should see the accompanying output as **UniScript** executes

Step	QA	Operator Entry/System Response	Description
		successfully opened. Report file >/disks/galex/users/galex/tcs/uniscript/stp5_2_3_2/stp5_2_3_2.rp2 successfully opened. Script file /disks/galex/users/galex/tcs/uniscript/stp5_2_3_2/stp5_2_3_2.tst successfully opened at level 0. "Sending POR" LFDNOOP WAIT 0: HKV0=1; HKV1=25; wc=5 "Transmitting synthesized LFDNOOP command with five non-zero parameters" "Waiting for HK data" WAIT 0: HKV0=2; HKV1=0; wc=1 WAIT 1: HKV1=2; wc=0 "Test stp 5.2.3.2 completed successfully"	

4.3 POST-OPERATION ACTIVITIES

4.3.1 Copy Reports to PC Files and Print Them

Using an FTP client, copy the **u**, **stp5_2_3_2.tst**, **stp5_2_3_2.rp1**, and **stp5_2_3_2.rp2** files to appropriate PC files. Include these files as Appendices A, B, C, and D with this completed form.

4.3.2 Complete The Test Procedure Form

Ensure that all blank fields in this report are completed correctly and submit the completed report to QA.

SUMMARY SHEET

OPERATION TITLE:_____ WOA#_____

TEST ARTICLES IDENTIFICATION (including serial and/or part numbers):

_____

TASKS/STEPS COMPLETED: _____

_____

LOCATION: _____

TEST STARTED: TEST TERMINATED

TIME: _____ Hr/Min TIME: _____ Hr/Min

DATE:_____ DATE:_____

LOGS USED: _____

ANOMALY REPORTS GENERATED: _____

_____

COMMENTS:_____

_____

_____

_____

_____

TEST CONDUCTOR: _____

Signature/Date

QA REPRESENTATIVE:_____

Signature/Date

Appendix A. Shell Script u

```
#!/bin/sh
pkill cosnoopy
perl ../UniScript.pl stp5_2_3_2 "0,0,0,0,0,0,0,0"
cosnoopy&
```

Appendix B. Test Script stp5_2_3_2.tst

```
; ********************************************************************************
; * STP 5.2.3.2                                                                  *
; * ---------------------------------------------------------------------------- *
; * Echoes for Command Opcodes and Parameters                                    *
; * ---------------------------------------------------------------------------- *
; * Send a valid Command String with Serial Number and five parameters to the DCE. *
; * Verify that the resulting HK packet contains the complete and proper Command   *
; * String echo.                                                                 *
; * ---------------------------------------------------------------------------- *
; * NOTE that since no actual DCE command requires or permits 5 parameters, we use *
; * a special command synthesized in Buffer 1, that is in fact a NOOP with para-   *
; * meters 0010, 0020, 0030, 0040, 0050. The S/N is 0001.                        *
; ********************************************************************************
;
SYM      DELTA=1
ECHO     2
;
DTG      3,"(0) Sending POR"
WTO      "Sending POR"
POR
WAIT     1
LFDNOOP
WAIT     5,HK
;
; *************************
; * Synthesize goofy NOOP *
; *************************
;
DATA     1, 0,8,CONST=0x045AFFAF_04580050
DATA     1, 8,8,CONST=0x0456FFBF_04540040
DATA     1,16,8,CONST=0x0452FFCF_04500030
DATA     1,24,8,CONST=0x044EFFDF_044C0020
DATA     1,32,8,CONST=0x044AFFEF_04480010
DATA     1,40,8,NEXT
DATA     1,48,8,CONST=0x04427F7F_04408080
;
LOG      1,1
;
; **********************************
; * Transmit the command to the DCE *
; **********************************
;
DTG      3,"(1) Transmitting synthesized LFDNOOP command with five non-zero parameters"
WTO      "Transmitting synthesized LFDNOOP command with five non-zero parameters"
XCMD     1
;
DTG      3,"(2) Waiting for HK data"
WTO      "Waiting for HK data"
WAIT     5,HK
LOG      1,LFDCBUF,LFDDIAGS
;
; ****************
; * Verify match *
; ****************
;
CHECK    1,($LFDCBUF[ 0] == hex(substr($B1,108,4)))
CHECK    1,($LFDCBUF[ 1] == hex(substr($B1,100,4)))
CHECK    1,($LFDCBUF[ 2] == hex(substr($B1, 92,4)))
CHECK    1,($LFDCBUF[ 3] == hex(substr($B1, 84,4)))
CHECK    1,($LFDCBUF[ 4] == hex(substr($B1, 76,4)))
CHECK    1,($LFDCBUF[ 5] == hex(substr($B1, 68,4)))
CHECK    1,($LFDCBUF[ 6] == hex(substr($B1, 60,4)))
CHECK    1,($LFDCBUF[ 7] == hex(substr($B1, 52,4)))
CHECK    1,($LFDCBUF[ 8] == hex(substr($B1, 44,4)))
CHECK    1,($LFDCBUF[ 9] == hex(substr($B1, 36,4)))
CHECK    1,($LFDCBUF[10] == hex(substr($B1, 28,4)))
CHECK    1,($LFDCBUF[11] == hex(substr($B1, 20,4)))
```

```
CHECK    1,($LFDCBUF[12] == hex(substr($B1, 12,4)))
CHECK    1,($LFDCBUF[13] == hex(substr($B1,  4,4)))
;
DTG      3,"(3) Test stp 5.2.3.2 completed successfully"
WTO      "Test stp 5.2.3.2 completed successfully"
```

Appendix C. Test Report stp5_2_3_2.rp1

```
                                          55555          222          333
222
                                          5             2   2         3   3         2
2
                          ssss  ttttt  pppp   555              2            3
2
                         s        t    p   p     5             2            3
2
                         sssss    t    pppp       5           2            3
2
                             s    t    p      5   5          2           3   3          2
                         ssss     t    p       555   _____  22222  _____   333   _____
22222

Ver 01.09  Fri Dec  1 00:31:40 2000    "(0) Sending POR"

LFDNOOP

Len  CRC  Buffer         Data
---- ---- ------         ----
0038 8E73 1              04 5A FF AF 04 58 00 50 04 56 FF BF 04 54 00 40 04 52 FF CF 04
50 00 30 04 4E FF DF 04 4C 00 20
                         04 4A FF EF 04 48 00 10 04 46 FF FD 04 44 00 02 04 42 7F 7F 04
40 80 80

Ver 01.09  Fri Dec  1 00:31:41 2000    "(1) Transmitting synthesized LFDNOOP command with
five non-zero parameters"

Ver 01.09  Fri Dec  1 00:31:41 2000    "(2) Waiting for HK data"

Addr Addr HK-Name        Value
---- ---- -------------- -----
1664-167F LFDCBUF        8080  7F7F  0002  FFFD  0010  FFEF  0020  FFDF  0030  FFCF  0040
FFBF  0050  FFAF

1780-179F LFDDIAGS       011B  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000
0000  0000  0000  0000  0000
17A0-17BF                0000  0000  0000  0000  0000  0000  0000  0000  0000  0000  0000
0000  0000  0000  0000  0000

CHECK:   ($LFDCBUF[ 0] == hex(substr($B1,108,4)))
eval:    (0000[ 0] == hex(substr(045A...,108,4)))

S U C C E S S

CHECK:   ($LFDCBUF[ 1] == hex(substr($B1,100,4)))
eval:    (0000[ 1] == hex(substr(045A...,100,4)))

S U C C E S S

CHECK:   ($LFDCBUF[ 2] == hex(substr($B1, 92,4)))
eval:    (0000[ 2] == hex(substr(045A..., 92,4)))

S U C C E S S

CHECK:   ($LFDCBUF[ 3] == hex(substr($B1, 84,4)))
eval:    (0000[ 3] == hex(substr(045A..., 84,4)))

S U C C E S S

CHECK:   ($LFDCBUF[ 4] == hex(substr($B1, 76,4)))
eval:    (0000[ 4] == hex(substr(045A..., 76,4)))

S U C C E S S

CHECK:   ($LFDCBUF[ 5] == hex(substr($B1, 68,4)))
eval:    (0000[ 5] == hex(substr(045A..., 68,4)))
```

```
S U C C E S S

CHECK:   ($LFDCBUF[ 6] == hex(substr($B1, 60,4)))
eval:    (0000[ 6] == hex(substr(045A..., 60,4)))

S U C C E S S

CHECK:   ($LFDCBUF[ 7] == hex(substr($B1, 52,4)))
eval:    (0000[ 7] == hex(substr(045A..., 52,4)))

S U C C E S S

CHECK:   ($LFDCBUF[ 8] == hex(substr($B1, 44,4)))
eval:    (0000[ 8] == hex(substr(045A..., 44,4)))

S U C C E S S

CHECK:   ($LFDCBUF[ 9] == hex(substr($B1, 36,4)))
eval:    (0000[ 9] == hex(substr(045A..., 36,4)))

S U C C E S S

CHECK:   ($LFDCBUF[10] == hex(substr($B1, 28,4)))
eval:    (0000[10] == hex(substr(045A..., 28,4)))

S U C C E S S

CHECK:   ($LFDCBUF[11] == hex(substr($B1, 20,4)))
eval:    (0000[11] == hex(substr(045A..., 20,4)))

S U C C E S S

CHECK:   ($LFDCBUF[12] == hex(substr($B1, 12,4)))
eval:    (0000[12] == hex(substr(045A..., 12,4)))

S U C C E S S

CHECK:   ($LFDCBUF[13] == hex(substr($B1,  4,4)))
eval:    (0000[13] == hex(substr(045A...,  4,4)))

S U C C E S S

Ver 01.09  Fri Dec  1 00:31:42 2000    "(3) Test stp 5.2.3.2 completed successfully"
```

Appendix D. Test Report stp5_2_3_2.rp2

```
                                        55555          222           333
222
                                        5             2   2         3   3         2
2
                         ssss  ttttt  pppp   555              2             3
2
                        s        t    p   p     5             2             3
2
                        sssss    t    pppp       5           2             3
2
                            s    t    p      5   5          2            3   3          2
                        ssss     t    p       555   _____  22222  _____   333   _____
22222

Ver 01.09  Fri Dec  1 00:31:40 2000    "(0) Sending POR"

-------------------------------------------------------------------------
                          P O R   P A C K E T
-------------------------------------------------------------------------
80000000
-------------------------------------------------------------------------

-------------------------------------------------------------------------------------------
                              C O M M A N D   P A C K E T
-------------------------------------------------------------------------------------------
            PARM4             PARM3             PARM2             PARM1             PARM0
045AFFFF 04580000 0456FFFF 04540000 0452FFFF 04500000 044EFFFF 044C0000 044AFFFF 04480000
-------------------------------------------------------------------------------------------
               SN            OPCODE
0446FFFE 04440001 04427F7F 04408080
-------------------------------------------------------------------------------------------


Ver 01.09  Fri Dec  1 00:31:41 2000    "(1) Transmitting synthesized LFDNOOP command with
five non-zero parameters"

-------------------------------------------------------------------------------------------
                              C O M M A N D   P A C K E T
-------------------------------------------------------------------------------------------
            PARM4             PARM3             PARM2             PARM1             PARM0
045AFFAF 04580050 0456FFBF 04540040 0452FFCF 04500030 044EFFDF 044C0020 044AFFEF 04480010
-------------------------------------------------------------------------------------------
               SN            OPCODE
0446FFFD 04440002 04427F7F 04408080
-------------------------------------------------------------------------------------------


Ver 01.09  Fri Dec  1 00:31:41 2000    "(2) Waiting for HK data"

-------------------------------------------------------------------------------------------
                              C O M M A N D   P A C K E T
-------------------------------------------------------------------------------------------
            PARM4             PARM3             PARM2             PARM1             PARM0
045AFFFF 04580000 0456FFFF 04540000 0452FFFF 04500000 044EFFFF 044C0000 044AFFFF 04480000
-------------------------------------------------------------------------------------------
               SN            OPCODE
0446FFFC 04440003 04427F7F 04408080
-------------------------------------------------------------------------------------------


Ver 01.09  Fri Dec  1 00:31:42 2000    "(3) Test stp 5.2.3.2 completed successfully"
```