41849-G159-026

TRW

1023

GROUND SOFTWARE MAINTENANCE FACILITY (GSMF) USER'S MANUAL

FINAL - APPENDICES

26 FEBRUARY 1986

(NASA-CR-178806) GROUND SOFTWARE MAINTENANCE FACILITY (GSMF) USER'S MANUAL. APPENDICES NASA-CR-178806 NAS 1.26:178806 REPT-41849-G159-026-APP HC A05/MF A01 Final Report (TRW Defense and Space Systems Group) G3/61 N86-25144 Unclas 42925

TRW DEFENSE SYSTEMS GROUP
HUNTSVILLE OPERATIONS
213 WYNN DRIVE
HUNTSVILLE, ALABAMA 35805

em Development Division
nse Systems Group

213 Wynn Drive
Huntsville, Alabama 35805
205-837-2400

SN41849.000
G002.VOB.86-6068
28 February 1986

McDonnell Douglas Astronautics Co.
Huntsville Division
P.O. Box 1181
Huntsville, Alabama 35807

ATTENTION: MR. W. R. LOKKEN

SUBJECT: SUBCONTRACT 83916018, GSMF DELIVERABLES,

Dear Mr. Lokken,

Per Supplemental Agreement 04, paragraph 4.2 of the June 1984 revised Statement of Work, enclosed are the following final reports and baselined software:

Detailed Design Document (Final)
Users Manual (Final)
System Manual (Final)
GSMF Baselined Software
Acceptance Review Demonstration Package

If you should have any questions, please contact Mr. Garry Griffith at MSFC, Building 4708/B26A or telephone (205) 453-3910.

TRW Inc.
Defense Systems Group

V O'L Bain

V. O'L. Bain
Contracts Manager
System Development Division

Encl: As Stated

cc: V. O. Bain, N. L. Blanks, F. P. Herring, M. E. Hollowich, G. M. Lide, G. C. Hitt, TRW File (2), V. A. Wulff

Delivery Package Received by:

_____ _____

W. R. Lokken, Branch Chief — Date
Software and Ground Computer Systems
MDAC-HSV

41849-G159-026

TRW

GROUND SOFTWARE MAINTENANCE FACILITY (GSMF) USER'S MANUAL

FINAL - APPENDICES

26 FEBRUARY 1986

PREPARED BY:

V. Aquila
D. Derrig
G. Griffith

APPROVED BY:

M. E. Hollowich
Project Manager

J. E. Irby, GSMF
Review Authority

TRW DEFENSE SYSTEMS GROUP
HUNTSVILLE OPERATIONS
213 WYNN DRIVE
HUNTSVILLE, ALABAMA 35805

CONTENTS

ILLUSTRATIONS

TABLES

PREFACE

The GSMF User's Manual consists of a "Final" and "Final-Appendices". These two volumes are bound separately for the convenience of the reader.

APPENDIX A

GSMF POWER-UP SEQUENCES

APPENDIX A

GSMF OPERATING PROCEDURES

The following is a list of operating procedures that shall be used to:

1) Start the GSMF 3250

2) Back up the system

3) Shut down the GSMF 3250 system

4) Save disk to tape

5) Perform a disk check and close any open files

6) Perform selective disk-to-tape

7) Initialize disk pack

8) Dump tape to printer.

A list of CSS Procedures is presented on page A-6 that will allow the user to assemble tasks, link, compile, backup the system, generate/establishment/print display pages, cancel tasks in memory, and to TET an assembly task without having to enter the .CSS commands every time. Examples are given for establishing, generating, and printing display pages (A-6, A-7).

PROCEDURE FOR STARTING THE GSMF 3250

SIGN THE UTILIZATION LOG BOOK

1. If the GSMF 'RELEASE' disk is not in the drive and "READY".

   1A. If the "RELEASE" disk is loaded.
   - A. Press the button under "START".
   - B. The drive will spin up and the "READY" light come on.

   1B. 1B. If another disk is loaded.
   - A. If the "READY" light is on, mark the disk off (MA MSM:, OFF) press the button under "START". The disk will spin down and the "READY" light will go out.
   - B. Be sure the "READY" light is out.
   - C. Open the door on the top of the disk drive.
   - D. Place the upper dust cover for thte disk presently mounted over the disk.
   - E. Turn the handle counter-clockwise until it clicks.
   - F. Remove the disk and dust cover assembly.
   - G. Place the lower dust cover on the disk.
   - H. Store the disk on a shelf in the disk rack.

   1C. If no disk is loaded:
   - A. Open the door on the top of the disk drive.
   - B. Remove the lower dust cover by squeezing the lock on the lower cover.
   - C. Place the disk with the upper dust cover still on over the spindle.
   - D. Turn the handle clockwise until it locks firmly.
   - E. Lift off the upper dust cover.
   - F. Place the upper cover neatly over the lower cover and store it on the back of the drive enclosure.
   - G. Press the button under 'START".
   - H. The drive will spin up and the "READY" light come on.

2. Move the key to "ON".

3. Move the IPL switch to "ENABLE".

4. Momentarily press the "INIT" switch. A list of devices will appear on the system console if you have made it so far. If not, try again or get help.

5. Move the IPL switch to "DISABLE".

6. Move the key to 'LOCK'.

7. Enter "DS67" at the prompt for device.

8. Enter 'GSMF3250.OS" at the prompt for file.

9. Memory test will run. Wait for it to end.

10. Enter time and date when prompted as 'SET TIME MM/DD/YY,HH:MM:SS"

11. Enter '$CONTINUE'

12. BACKUP PROCEDURES - First IPL of the day only!!!!!!!

MAKE SURE THE 'PE/NRZI' LIGHT IS LIT ON THE TAPE DRIVE

The printer must be ready and a tape loaded for the backup to run. If problems persist with one or the other, restart the backup to preclude 'HIDDEN" problems.

12A. Monday - Thursday backup procedure:

A. Mount the daily tape for the proper day of the week.

Mount the reel and thread the tape as indicated on the tape drive.

Wind 3 or 4 turns on the lower reel.

Press "LOAD". The 'ONLINE' and "LOAD" lights will come on. If not, try again or get help

B. "DAILY" wait until backup finishes.
C. Dismount and store the tape.

Press "RESET" then "REWIND". Holding "REWIND" will wind the entire tape back on the upper reel.

D. Put the listing in the book marked "DAILY".

12B. Friday backup:

A. Check the log sheet to see if a Friday or total backup is due this date.

If Friday:

Load the proper Friday tape as indicated on the log sheet.

"DAILY" wait until backup finishes.

Dismount and store the tape.

Put the listing in the book marked "FRIDAY".

If total:

Load the first reel of the proper total tape as indicated on the log sheet.

"TOTAL" wait until backup finishes. This will require mounting several reels as prompted by the procedure.

Dismount and store the tapes.

Put the listing in the book marked "TOTAL".

B. A-F-T-E-R the backup is run reset the "DAILY" procedure:
'EDIT'
'G DAILY.CSS"
Change the date following the keyword "SINCE" to the Friday date on Line 11 of the CSS.

13. "UP"

PROCEDURE FOR SHUTTING THE GSMF 3250 DOWN

1. Get everyone clear and signed off.
2. "CLEANUP"
3. 'MA MSM:, OFF"
4. "D D" - Verify that the devices are off line.
5. Press the "START" button on the disk drive. The "READY" light will go out.

SIGN THE UTILIZATION LOG BOOK

TO SAVE THE FULL DISK TO TAPE:

1. Turn Error Log off with 'ERROR LOG,OFF".
2. Load a blank tape on right hand drive.
3. 'LOAD B,BACKUP,48"
4. "T B"
5. "ST ,IN=MSM: ,OUT=MAG1: ,LI=PR: ,VERIFY

TO PERFORM A DISK CHECK AND CLOSE ANY OPEN FILES:

1. "MA MSM: ,ON,P"
2. "LOAD D,DISCHECK
3. "TA D"

4. "MA MSM:,OFF"

5. "ST ,MSM: ,CON:,CL"

6. "MA MSM:,ON"

TO PERFORM SELECTIVE SAVE DISK-TO-TAPE:

1. 'LOAD BKUP,BACKUP,48"

2. 'TASK BKUP"

3. "ST ,IN=MSM: ,OUT=MAG1: ,LI=PR: .VE,SEL=COON:"

4. The COPY task will ask for the names of files to be saved to tape. The input is terminated by a "/*" keyboard entry.

TO INITIALIZE A DISK PACK:

1. Load a disk that contains "FASTCHEK" into the disk drive.

2. "MA MSM: ,ON,,CD=ALL"

3. "LO FASTCHECK"

4. "O R"

5. "MA MSM: ,OFF"

6. Unload previous disk and load the disk to be initialized.

7. "TA FASTCHEK"

8. "ST ,C=CON:,LI=PR:

9. Respond to interactive questions.
   - Desired operation - 'INIT=MSM:"
   - Volume name - "MSM1"
   - Fastchek mode - "Fill=BBDBDBD"
   - Fastchek: OK to Run - "Yes"

10. "CA FASTCHEK"

11. 'MA MSM:,ON,,CD=ALL"

DUMP TAPE CONTENTS TO THE PRINTER:

1. "LO COPY 32,,30"

2. "ST"

3. COPY 32 > "IN MAGX:,3000,VAR"

4. COPY 32 > "LI PR:"

5. COPY 32 > "DI *,NRECS=1"

AVAILABLE CSS PROCEDURES

1. MACROASM.CSS - Assembles a CAL file with macro expansion (FILE.ASM)

2. TET.CSS - TET an assembly task

3. LNKALL.CSS - LINK a task (User modifications necessary)

4. FORT.CSS - Compile a FORTRAN subroutine

5. DAILY.CSS - Backup procedure (User modifications may be necessary)

6. RETET.CSS - LINK an assembly tsk (Must already exist)

7. CLEANUP.CSS - Cancels tasks in memory

8. GSMF3.CSS - Brings up the GSMF system

9. PRTDSPLY.CSS - Enables printing of display pages

10. DSKDSPLY.CSS - Generation of display pages

11. ESTDSPLY.CSS - Establishment of display pages

12. MODBCD [COMP=NO],[LINK=NO],[CLEAN=NO]
    COMPILE (OPTIONALLY) AND LINK (OPTIONALLY)
    CLEAN = NO RETAINS ALL WORKFILES SUCH AS THE .MAP
    INPUT: FILE.FTN (SOURCE)
           FILE.CMD (LINK COMMANDS)
    OUTPUT: FILE.LST (NORMAL)
            FILE.TSK (IF LINK=NO NOT SPECIFIED)

NOTES ON THE GSMF SYSTEM

1. Start system - "UP"

2. Generate and establish display pages

   "DSKDSPLY DP _____"

   "ESTDSPLY DP _____"

3. Print display page

   "PRTDSPLY"

   > "0001-0001"

   > "END"

4. After new display pages have been created,

   "DELETE DFTEMP.LOG"

   before starting GSMF system.

APPENDIX B

ACRONYMS AND TERMINOLOGY

APPENDIX B
ACRONYMS AND TERMINOLOGY

ATE	Automatic Test Equipment. (ATE computer is used to designate Mitra GCOS host)
BITE	Built in Test Equipment
BSR	Bite Status Request
CID	Computer Interface Device
CDR	Critical Design Requirements
CMS	Conversational Monitor System (IBM)
CRT	Cathode Ray Tube
DBGM	Data Base Generation Management
DMA	Direct Memory Access
ECOS	Experiment Computer Operating System
ECP	Engineering Change Proposal
FIFO	First In - First Out
GCID	Ground Computer Interface Device
GCOS	Ground Computer Operating System
GFE	Government Furnished Equipment
GOAL	Ground Operations Aerospace Language
GSMF	Ground Software Maintenance Facility
HAL/S	Houston Aeronautical Language/Spacelab
HOL	High Order Language
HOST	This is a reference to the P-E computer that is the host to the GSMF operation. The reference may also accommdate the SDF-Host references for the Integrated mode descriptions
IPS	Instrument Pointing System
KSC	Kennedy Space Center
MACI	Monitor, Access, and Control Interface

Major Cycle	One second basic time frame under which host computer (P-E. for GSMF) is required to service data requiremennts of ATE/GCID.
MDTSCO	McDonnell Douglas Technical Services Company
Mitra	A computer upon which different operating systems execute (ECOS, GCOS, and SCOS) for the test equipment processing
Module	A portion of software in a computer system which may be indepedently described, designed, coded and tested. If may be a subset of a larger set or superset (i.e., a task or a functional unit).
MSE	Measuring and Stimuli Equipment
MSFC	Marshall Space Flight Center
NASA	National Aeronautics and Space Administration
PCU	Payload Checkout Unit
PDR	Preliminary Design Review
P-E	Perkin-Elmer
PIOL	Periodic Input/Output Loop
PPI	Processor to Processor Interface (A P-E Communication Device)
SCOS	Support Computer Operating System
SDF	Software Development Facility
SL	Spacelab
SLDB	Spacelab Data Base
SMID	Simulation Identification
SVC	Supervisor Call
SWID	Software Identification
Task	A process in a computer which can perform its function independent of other processes. It may depend on other process for data or scheduling.
TMB	Telemetry Buffer
UDF	Unit Development Folder
VDU	Visual Display Unit
VM	Virtual Memory

APPENDIX C

SWID RELATION FORM GENERATION PROCEDURE

APPENDIX C
SWID RELATION FORM GENERATION PROCEDURE

The SWID FORTRAN Program extracts ATE subtable data located in the Spacelab Data Base (SLDB) which resides on the IBM 4341. This data will be recorded by SWID for each ATE subtable and transcribed to a SEID relations form.

The ATE subtables to be extracted from the SLDB are defined in Figure C-1.

To extract the ATE subtables and generate the aforementioned SWID Relations forms, the user will exercise step 1 described in Figure F-1 of Appendix F. This VM/CMS procedure will prompt the user to all ATE subtables to be addressed and generate this corresponding SWID Relations form.

SUBTABLE NAME	SUBPART
A/AIDA	1
A/AISC	1
A/AIMS	1
A/AOSA	1
A/DIGDA	1
A/DIGM	1
A/DIGS	1
A/DISDA	1
A/DISM	1
A/DORM	1
A/DOSM	1
A/GIDA	1
A/GIMS	1
A/GISC	1
A/GOMS	1
A/GOCL	1
A/SOCD	1

Figure C-1. Subtables and Subparts

APPENDIX D

STIMULI/MEASUREMENT SWID PAIRS GENERATION

APPENDIX D

STIMULI/MEASUREMENT SWID PAIRS GENERATION

The SWID relations file will contain appended information which defines the measurement SWIDs affected by a given stimuli SWID (for this reason, the SWID relations file is also referred to as the stimuli/measurement SWID pairs file in the GSMF documentation). This information will be categorized into four parts (i.e., SWID, measurement SWID(s), count, and behavior function. The information, which resides within the stimuli/measurement SWID Pairs File, is collected and compiled through a series of steps involving manual processes of matching and mapping data. The first process is to manually locate a stimuli SWID. Once a stimuli SWID is located, information such as group number, subtable name, and SNI will be used to yield a reference to other data pertaining to the stimuli SWID, namely, measurement SWID(s), measurement SWID count, and behavior function.

With reference to Figure D-1, Stimuli SWID Number 9245 is one of many SWID(s) in the host database. It is manually found by scanning the database report.

Stimuli SWID 9245 is mapped by retaining:

1) The predefined group number, which is 11 in this case

2) The subtable name, which is SOCD

3) The SNI, which is "Clear TLM Error Counters."

NOTE: "Counters" indicates more than seven measurement SWIDs will be affected by SWID 9245.

After obtaining the stimuli SWID information, a scan of the database listing is made for the affected measurement SWID. Referring to Figure D-2:

1) A search for an identical SNI (which is "Clear TLM Error Counters" in this case) predefined in its given group number (which is 11 in this case) is found.

2) It is found that three SWIDs, 8197, 8198, and 8199 resulted from the mapping and cross reference of stimuli SWID Number 9245. Thus, the measurement SWID count is 3 and the measurement SWIDs which were effected by Stimuli SWID Number 9245 are 8197, 8198, and 8199.

	FUNCTION DESIGNATOR	SW ID	SW CS	SUB TABLE NAMES	HLCW	SHORT FORM OF SNI	ARAT	SART REFRESH VALUE	HARDWARE ADDRESS 1	PP 2	PP 3	GRPG	BIT FUNC	INTLK	INSTRUMENT RANGE LOW 0V	INSTRUMENT RANGE HIGH +5 V	SYMBOLIC NAME OF INSTRUMENT (SNI)
1	S11K0148P	9225	1	SOCD	1	TCBSI LP/PO	3224										TLC LOOP/POLARITY MODE
1	S11K0149P	9226	1	SOCD	1	TCBSI INPCD	3304										TLC INPUT CODE
1	S11K0150P	9227	1	SOCD	1	TCBSI ACQLP	3401										TLC ACQUISITION LOOP
1	S11K0151P	9228	1	SOCD	1	TCBSI INPSR	3510										TLC INPUT SOURCE
1	S11K0152P	9229	1	SOCD	1	TCBSI INPLP	3600										TLC INPUT LOOP
1	S11K0173P	9230	1	SOCD	1	TCFSI AUTO	4001										TLC AUTO MODE CRITERIA
1	S11K0174P	9231	1	SOCD	1	TCFSI MSGFR	4100										TLC NUMBER OF MESSAGE FRAMES
1	S11K0175P	9232	1	SOCD	1	TCFSI WD/FR	4234										TLC WORDS PER FRAME (TENS)
1	S11K0176P	9233	1	SOCD	1	TCFSI BIT/W	4315										TLC BITS PER WORD
1	S11K0177P	9234	1	SOCD	1	TCFSI SY0/1	440F										TLC LENGTH OF SYLLABLE 0/1
1	S11K0178P	9235	1	SOCD	1	TCFSI SY2/3	4500										TLC LENGTH OF SYLLABLE 2/3
1	S11K0179P	9236	1	SOCD	1	TCFSI ACQCR	4600										TLC ACQUISITION CRITERIA
1	S11K0180P	9237	1	SOCD	1	TCFSI SYNC	4700										TLC FRAME SYNCHRONIZATION
1	S11K0181P	9238	1	SOCD	1	TCFSI WDPOS	4800										TLC ID-WORD POSITION
1	S11K0182P	9239	1	SOCD	1	TCFSI BITS	4900										TLC NO OF BITS PRECEDING ID
1	S11K0183P	9240	1	SOCD	1	TCFSI WDTYP	4A00										TLC ID-WORD TYPE
1	S11K0184P	9241	1	SOCD	1	TCFSI CODE1	4BC7										TLC FRAME SYNC CODE (1)
1	S11K0185P	9242	1	SOCD	1	TCFSI CODE2	4CE3										TLC FRAME SYNC CODE (2)
1	S11K0186P	9243	1	SOCD	1	TCFSI CODE3	4D01										TLC FRAME SYNC CODE (3)
1	S11K0187P	9244	1	SOCD	1	TCFSI CODE4	4E00										TLC FRAME SYNC CODE (4)
1	S11K0122P	9245	1	SOCD	1	TM CLR EC	5170										CLEAR TLM ERROR COUNTERS
1	S11K0123P	9246	1	SOCD	1	TM SET STOP	5701										SET STOP MODE (TLM)
1	S11K0124P	9247	1	SOCD	1	TM SET TEST	5704										SET TEST MODE (TLM)
1	S11K0125P	9248	1	SOCD	1	TM CLR STOP	5702										CLEAR STOP MODE (TLM)
1	S11K0126P	9249	1	SOCD	1	TM CLR TEST	5708										CLEAR TEST MODE (TLM)
1	S11K0127P	9250	1	SOCD	1	TM SEL DIAC	5101										SELECT DIAGNOSTIC DATA
1	S11K0128P	9251	1	SOCD	1	SEL TM DATA	5102										SELECT TELEMETRY DATA
1	S11K0136P	9256	1	SOCD	1	TC CLR FLS	5807										CLEAR ERROR FLAGS
1	S11K0161P	9257	1	SOCD	1	INITMSE C/O	6001										INITIALIZE MSE FOR C/O

Figure D-1. ATE SWID Report (Stimuli)

	FUNCTION DESIGNATOR	SW ID	SW CS	SUB TABLE NAMES	HLC W	SHORT FORM OF SNI	ART	SART REFRESH VALUE	HARDWARE ADDRESS 1	2	3	GRPFG	BIT PNUM	LNG	INTLK	INSTRUMENT RANGE LOW OV	HIGH +5 V	SYMBOLIC NAME OF INSTRUMENT (SNI)
1	S11K0162P	9258	1	SOCD	1	INITMSE RPL	6004											INITIALIZE MSE FOR REPLAY
1	S11K0166P	9259	1	SOCD	1	MET STOP	700F											MET SET STOP
1	S11K0173P	9266	1	SOCD	2	MET INITIAL	7500						7400					MET PRESET - INITIALIZE MET TIME
2							7300						7200					
3							7100											
1	S11K0674P	9267	1	SOCD	1	MET SET RUN	77F0											SET RUN
1	S11K0533P	9268	1	SOCD	1	PSSU SELCH1	C120											SELECT CHANNEL 1 ON LINE
1	S11K0534P	9269	1	SOCD	1	PSSU SELCH2	C220											SELECT CHANNEL 2 ON LINE
1	S11K0535P	9270	1	SOCD	1	PSSU SELCH3	C320											SELECT CHANNEL 3 ON LINE
1	S11K0536P	9271	1	SOCD	1	PSSU SELCH4	C420											SELECT CHANNEL 4 ON LINE
1	S11K0537P	9272	1	SOCD	1	PSSU SELCH5	C520											SELECT CHANNEL 5 ON LINE
1	S11K0538P	9273	1	SOCD	1	PSSU SELCH6	C620											SELECT CHANNEL 6 ON LINE
1	S11K0539P	9274	1	SOCD	1	PSSU SELCH7	C720											SELECT CHANNEL 7 ON LINE
1	S11K0540P	9275	1	SOCD	1	PSSU SELCH8	C820											SELECT CHANNEL 8 ON LINE
1	S11K0541P	9276	1	SOCD	1	PSSU SELCH9	C920											SELECT CHANNEL 9 ON LINE

1SLD3 VERSION DATA: 84.08.22 TIME: 16:04 MISSION: 3 SUBPART: ATE

:: MDTSCO :: SPACELAB DATA BASE REPORT ON CONFIGURATION ----SL03 : GROUND :: DOC NO:
:: V1303 :: SLDB DATE: 84.08.22 TIME: 16:04 SOFTWARE LIST : COMMANDS :: ISS: DATE: 11/9/84
:: CHANGE FLAG: 840423 :: MISSION: 3 SUBPART: ATE SORTED BY ID : :: PAGE: 5 (PART: 9)

Figure D-1. ATE SWID Report (Stimuli) (Continued)

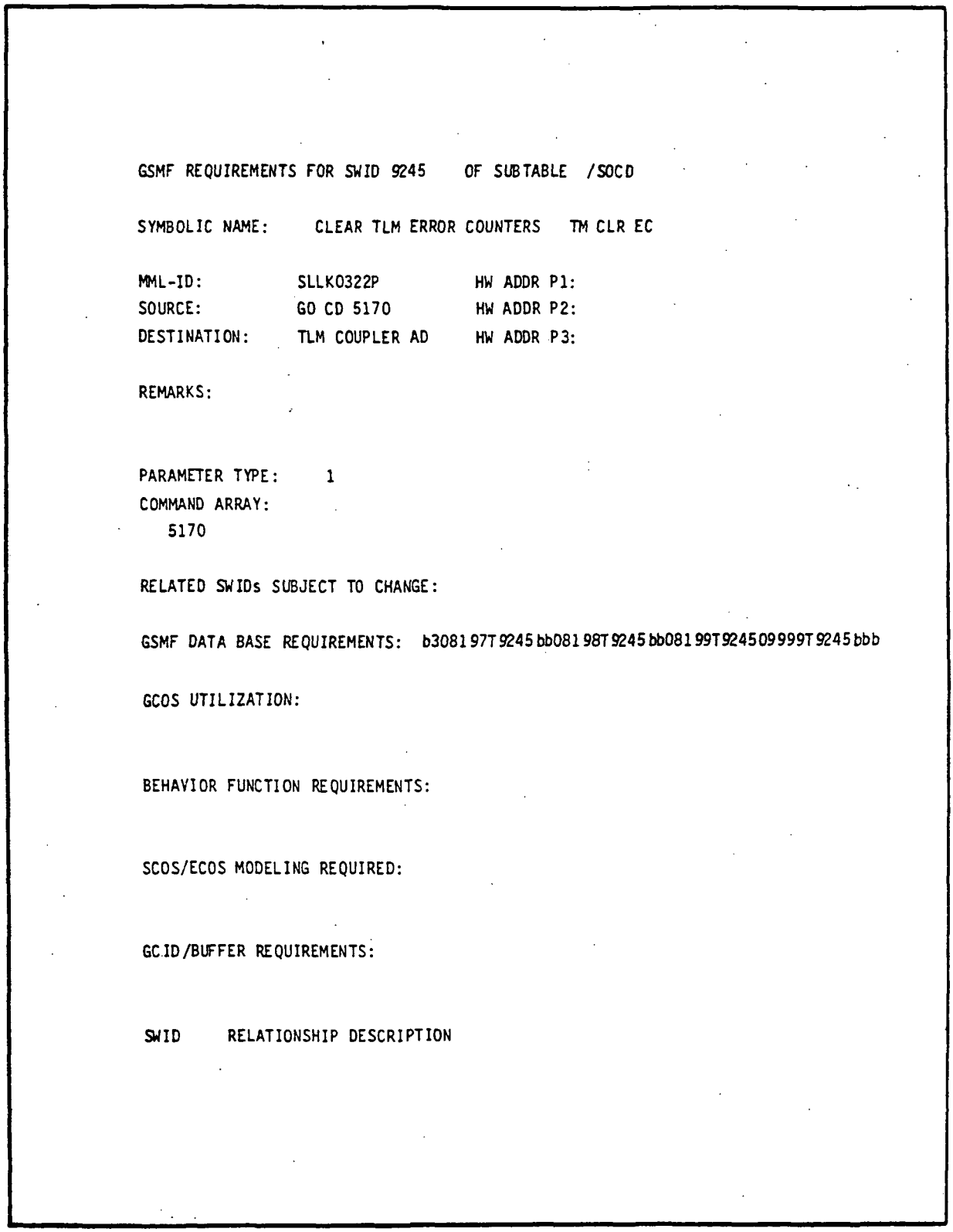
```
S11X0213X  8167 1 DIGS  TRUE    TMBS PH      2  4  10      FALSE  TRUE    TLM BIT SYNCHR PHASE IN LIMIT3
S11X0214X  8168 1 DIGS  TRUE    TMBS FREQ    2  4  11      FALSE  TRUE    TLM BIT SYNCHR FREQU IN LIMIT3
S11X0215X  8169 1 DIGS  OFF     TMBS PWRSU   2  4  12      ON     OFF     TLM BIT SYNCHR POWER SUPPLY
S11X0216X  8170 1 DIGS  OFF     TMBS INPSQ   2  4  13      ON     OFF     TLM BIT SYNCHR INPUT SIGNAL
S11X0230P  8171 1 GISC          TMFS STAT    2  5   0 16                  TLM FRAME SYNCHRONIZER STATUS
S11X0231X  8172 1 DIGS  ON      TMFS HI      2  5   0      OFF    ON      TLM FRAME SYNCHR HEALTH IND
S11X0232X  8173 1 DIGS  ON      TMFS NOAK    2  5   1      OFF    ON      TLM FRAME SYNCHR NON-ACKN CMD IND
S11X0233X  8174 1 DIGS  ON      TMFS MAN     2  5   2      OFF    ON      TLM FRAME SYNCHR MANUAL MODE IND
S11X0234X  8175 1 DIGS  ON      TMFS LOS     2  5   3      OFF    ON      TLM FRAME SYNCHR LOSS OF SYNCH IND
S11X0235X  8176 1 DIGS  OFF     TMFS AUTO    2  5   8      ON     OFF     TLM FRAME SYNCHR AUTO MODE
S11X0236X  8177 1 DIGS  ON      TMFS LOCK    2  5   9      OFF    ON      TLM FRAME SYNCHR LOCK PHASE
S11X0237X  8178 1 DIGS  ON      TMFS CNTL    2  5  10      OFF    ON      TLM FRAME SYNCHR CONTROL PHASE
S11X0238X  8179 1 DIGS  ON      TMFS SEAR    2  5  11      OFF    ON      TLM FRAME SYNCHR SEARCH PHASE
S11X0239P  8180 1 GISC          TMFS FR ERR  2  5  12  4                  TLM FRAME SYNCHR FRAME ERRORS
S11X0265P  8184 1 GISC          TMCA STAT    2  6   0 16                  TLM COUPLER ADAPTOR STATUS
S11X0266X  8185 1 DIGS  ON      TMCA HI      2  6   0      OFF    ON      TLM COUPL ADAPT HEALTH IND
S11X0267X  8186 1 DIGS  OFF     TMCA MSTRCK  2  6   2      ON     OFF     TLM COUPL ADAPT MASTER CLOCK
S11X0268X  8187 1 DIGS  ON      TMCS DATAFI  2  6   3      OFF    ON      TLM COUPL ADAPT DATA FAIL IND
S11X0269X  8188 1 DIGS  ON      TMCA DIALFI  2  6   4      OFF    ON      TLM COUPL ADAPT DIALOG FAIL IND
S11X0270X  8189 1 DIGS  OFF     TMCA MSTRCK  2  6   8      ON     OFF     TLM COUPL ADAPT MASTER CLOCK
S11X0271X  8190 1 DIGS  OFF     TMCA STOP    2  6   9      ON     OFF     TLM COUPL ADAPT STOP MODE
S11X0272X  8191 1 DIGS  OFF     TMCA TEST    2  6  10      ON     OFF     TLM COUPL ADAPT TEST MODE
S11X0273X  8192 1 DIGS  OFF     TMCA INPMO   2  6  11      ON     OFF     TLM COUPL ADAPT INPUT MODE
S11X0274X  8193 1 DIGS  OFF     TMCA INPSQ   2  6  12      ON     OFF     TLM COUPL ADAPT INPUT SIGNAL
S11X0275X  8194 1 DIGS  OFF     TMCA DIAC    2  6  13      ON     OFF     TLM COUPL ADAPT DIAGNOSTIC MODE
S11X0276X  8195 1 DIGS  FALSE   TMCA SSB     2  6  14      TRUE   FALSE   TLM COUPL ADAPT SS BASE SELECTED
S11X0277X  8196 1 DIGS  OFF     TMCA DMAVCT  2  6  15      ON     OFF     TLM COUPL ADAPT DMA VECTOR 200
S11X0280P  8187 1 AISC  UNITS   TM MSQ EC    2  7                        TLM MESSAGE ERROR COUNT
S11X0284P  8198 1 AISC  UNITS   TM SS WD EC  2  8                        TLM SS WORD ERROR COUNT
S11X0285P  8199 1 AISC  UNITS   TM EX WD EC  2  9                        TLM EXP WORD ERROR COUNT
S11X0292P  8200 1 GISC          TM DMA VCTR  2 10   0 16                  TLM DMA VECTOR (2E16 - 2E01)
ISLDS VERSION DATA:  84.08.22  TIME:  16:04  MISSION:  3 SUBPART:  ATE

:: MDTSCO              :: SPACELAB DATA BASE REPORT ON CONFIGURATION ----SL03  : GROUND MEAS.    :: DOC NO:
:: V1303               ::   SLDS DATE:  84.08.22  TIME:  16:04  SOFTWARE LIST  : INSTRUMENTATION  :: ISS:     DATE:  11/9/84
:: CHANGE FLAG:  840423 ::   MISSION:    3    SUBPART: ATE       SORTED BY ID   :      DATA        :: PAGE: 8  (PART: 3)
```

Figure D-2. ATE SWID Report (Measurement)

Referring to the SWID relations form for SWID 9245 in Figure D-3, the measurement SWID information obtained would be entered as GSMF data base requirements (i.e., 3 for the measurement SWID count, 8197, 8198, and 8199) for the measurement SWIDs and T9245 as the behavior function name. This would constitute the Stimuli/measurement SWID pairs for Stimuli SWID 9245.

The format for this STIMULI/SWID relationship would be indicated in Figure D-3. When SWID measurements effected are more than three, continuation is started directly beneath the start of the previous line (indicated in Figure D-3 as SWID 9999). It must be noted that all measurement SWIDS are defined in a five character field right justified and behavior functions in a eight character field left justified. A maximum of fourteen measurement SWIDs can be accommodated for each Stimuli SWID. (A blank character for purpose of clarity is indicated by b.)



GSMF REQUIREMENTS FOR SWID 9245 OF SUBTABLE /SOCD

SYMBOLIC NAME: CLEAR TLM ERROR COUNTERS TM CLR EC

MML-ID:	SLLK0322P	HW ADDR P1:
SOURCE:	GO CD 5170	HW ADDR P2:
DESTINATION:	TLM COUPLER AD	HW ADDR P3:

REMARKS:

PARAMETER TYPE: 1

COMMAND ARRAY:

5170

RELATED SWIDs SUBJECT TO CHANGE:

GSMF DATA BASE REQUIREMENTS: b308197T9245bb08198T9245bb08199T924509999T9245bbb

GCOS UTILIZATION:

BEHAVIOR FUNCTION REQUIREMENTS:

SCOS/ECOS MODELING REQUIRED:

GCID/BUFFER REQUIREMENTS:

SWID RELATIONSHIP DESCRIPTION

Figure D-3. SWID Relations Form

APPENDIX E

PCU OPERATING SYSTEM SERVICES

APPENDIX E

PCU OPERATING SYSTEM SERVICES

1. OVERVIEW

The Display Control software utilized by the GSMF system has been obtained from the Payload Checkout Unit (PCU) software. Hence, the following documentation has been extracted from the Payload Checkout Unit Application Software User's Guide (IBM Document Number 7940054B, IBM Federal Systems Division) with minor revisions to reflect functional usage with the GSMF system.

2. DISPLAY SUPPORT

Display Support includes on-line and off-line services that are provided specifically for use with the CRT consoles attached to the P-E 3220.

Subsections 2.1, 2.2, and 2.3 which follow describe the support provided (i.e., display services) from an operational standpoint, display generation, which is an off-line preparation service, and display communications, which is an application task real-time service.

2.1 Display Services

2.1.1 Overview

The GSMF operator(s) interfaces with system and application software (and vice-versa) from the P-E 1200s. These devices, each consisting of a keyboard and CRT screen, are referred to as test consoles (TC) and allow two-way communication utilizing the following elements:

a) Preformatted display pages

b) Compose fields

c) Menus

d) Fill-in fields

e) 0-line messages

f) 1-line commands

g) Error numbers.

These elements and their general usages are discussed in the following subsections. In addition, a layout of the TC keyboard is provided (Figure E-1).

2.1.2 Preformatted Display Pages

The GSMF system is built around a set of predefined display pages. These pages will be titled (first/second lines) based on their function. Other elements that may be present are descriptive text, menu options, compose fields, and fill-in fields. The Display Control software also maintains GMT and MET/CDT in line 1, columns 1-16 and 61-80, respectively, for all active TC screens.

2.1.3 Compose Fields

Compose fields, areas on the screen denoted by underscores (_), provide the mechanism for the operator to enter data destined for the controlling software. The following items are applicable to compose fields:

a) The entire compose field must be filled prior to depressing any one of three SEND keys.

b) Data entered into a compose field is retained across other displays. Compose data may be resent by re-entering as little as one character.

c) The TAB and BACK-TAB (|→) keys are used to position cursor to individual compose fields (including 1-line).

d) The cursor must be positioned past the compose field to be sent prior to depressing SEND key. This is performed automatically if data is entered into the entire field.

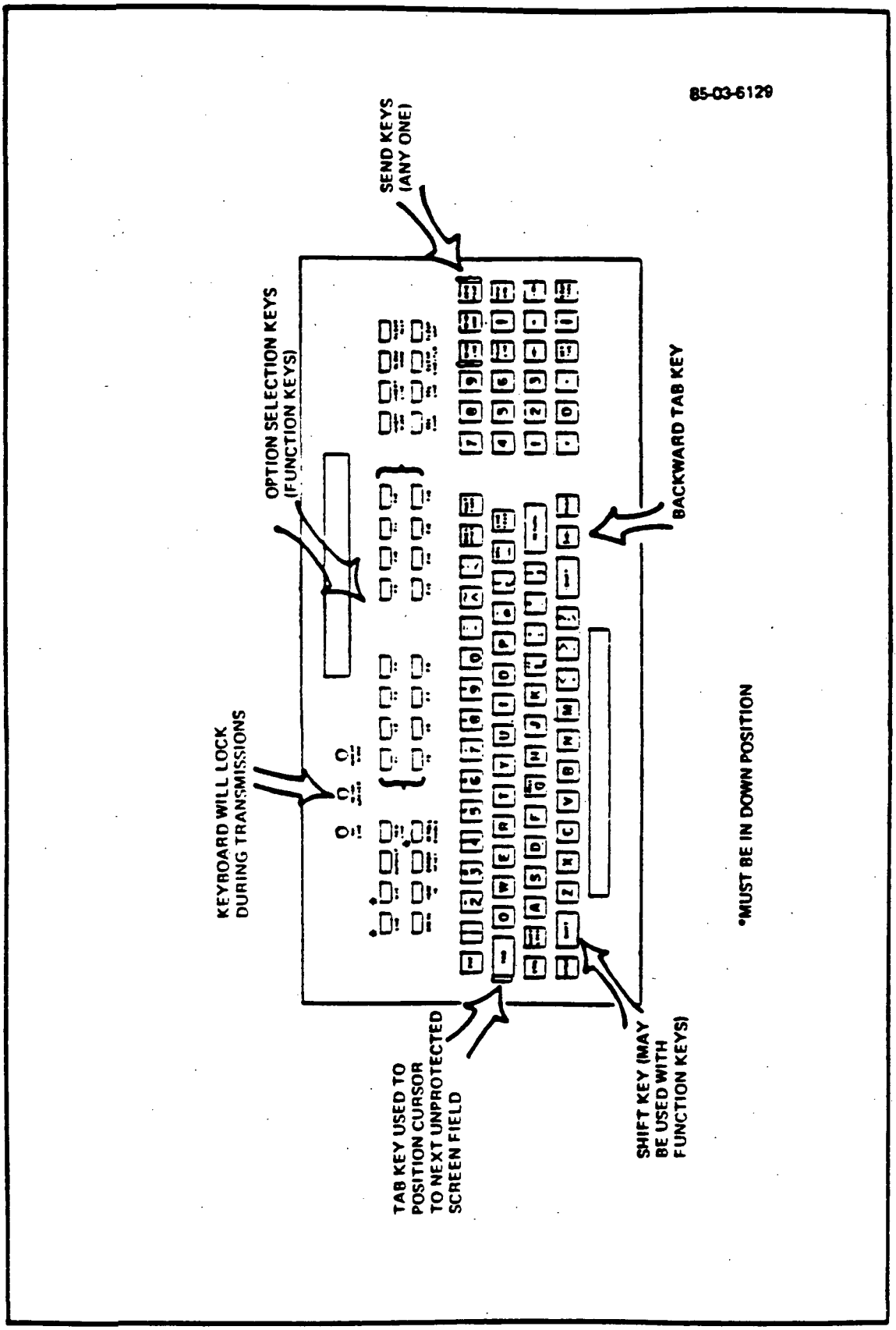


Figure E-1. Test Console Keyboard

e) Compose fields (data) must be sent individually, i.e., one at a time.

f) The audible alarm will be heard if data is sent. If the alarm is not heard, depress SEND key again. If unsure, depress the SEND key again since entry will be ignored if field was already sent.

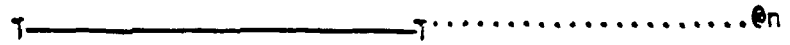
g) If data entered is invalid, a message indicating such will be displayed within the O-line. Reenter and resend the data.

h) The sending of compose field data may or may not cause a new display to be brought up.

i) Any attempt to enter data into any position on the screen other than a compose field (and I-line), will result in a keyboard lock condition.

2.1.4 Menus

A menu is defined to be a group of 1 to 16 options selectable for a given display with each option defined textually on one and only one line. In general, an option line takes on the following form:



Where T is the text characters describing the option and n is a number (1-16) identifying the menu option number. Function keys one (1) to sixteen (16) are used to select individual options; therefore, n also indicates the function key. Usually a new display is brought up on option selection but not necessarily.

2.1.5 Fill-in Fields

Fill-in fields provide one method by which GSMF software communicates with the operator. These predefined fields (up to 125 per display page) may be updated with function dependent variable data at any time regardless of whether the page is being viewed. Fill-in data is always broadcast to all active TCs containing a copy of the target display page. Note that all fill-in field data must be sent as ASCII data.

2.1.6 O-Line Messages

Two classes of messages are output to the dedicated O-line (line 23) by the GSMF software. Information conveyed by a given message is function dependent and self explanatory. The particular message class (A or B) is always indicated at the beginning of the message.

2.1.6.1 Class A Messages

Class A messages normally indicate an operational problem that requires operator intervention. Class A messages must be specifically acknowledged (Ref. 2.1.7.2) and are output on a FIFO basis. A Class A message will override a Class B message on the O-line. Also Class A messages are always broadcast to all active TCs.

2.1.6.2 Class B Messages

Class B messages are provided for operator informational purposes or to indicate a non-critical error. Class B messages are output on a FIFO basis and are acknowledged indirectly upon depressing any function or send key.

2.1.7 I-Line Commands

I-line commands are provided to communicate a specific operator request regardless of the display page being viewed. The command is entered by tabbing to the I-line, typing a one to four character mnemonic, an optional operand, and depressing a send key. Individual I-line commands are described in the following paragraphs.

2.1.7.1 Error Number Acknowledge (NAC)

The NAC command is used to acknowledge the currently flashing error number (Ref. 2.1.8). No operand is involved and shifted function key one duplicates this command.

2.1.7.2 Message Acknowledge (MAC)

The MAC command is used to acknowledge the currently displayed class A message. No operand is involved and shifted function key #2 duplicates this command. A Class A message may only be acknowledged from the TC (1-5) identified by the number just preceding the Class A indicator. This is the broadcasting TC (i.e., the original target TC of the applications task's request).

2.1.7.3 Identify Current Error Number (ID)

The ID command is used to identify the currently flashing error number (Ref. 2.1.8). Entering this command results in display skeleton DP0000 overriding the current display and error number related text being filled into an associated fill-in area. No operand is involved and shifted function key three duplicates this command.

2.1.7.4 Display (D)

The D command is used to refresh the screen with a specified display page, i.e., override the current page. The operand, a 1-4 digit number, specifies the display page.

2.1.7.5 Load a Task (LO)

The LO command is used to load the task named in the 1-8 character operand. The named task must be a task file (TSK extension) and contained on volume MSM1.

2.1.7.6 Start a Task (ST)

The ST command is used to start a task named in the 1-8 character operand. The named task must have been previously loaded (Ref. 2.1.7.5).

2.1.7.7 Cancel a Task (CA)

The CA command is used to cancel a task named in the 1-8 character operand.

2.1.7.8 Suspend a Task (SUS)

The SUS command is used to suspend (pause) a task named in the 1-8 character operand.

2.1.7.9 Release a Task (REL)

The REL command is used to release a task named in the 1-8 character operand. The named task must have been previously suspended (Ref. 2.1.7.8).

2.1.7.10 Send Message to Task (SEN)

The SEN command is used to send a 1-64 character message (2nd operand) to the task named in the 1-8 character 1st operand. The target task must be in memory and capable of receiving messages.

2.1.8 Error Numbers

Abnormal conditions detected by GSMF software are normally conveyed to the operator via a 3-digit error number. Operator intervention may or may not be required; however, the number must be acknowledged (Ref. 2.1.7.1). If the operator does not know the meaning of a specific number, the ID command may be used - for a detailed explanation (Ref. 2.1.7.3). Error numbers are placed into a dedicated flashing field on line 2, column 70-72 of the TC screen. Error numbers are not broadcast.

2.1.9 Hierarchical Display Support

The Display Control software supports a hierarchical paging structure whereby a next page, the same page, or the previous page may be associated with either a menu option line or a compose field. This association is made at display generation time and basically, allows previously displayed pages to be recalled in the reverse order that they were orginally displayed.

2.1.10 Multiple SCS Support

The display control software in reality is one to five independent tasks (reentrant) (DSPLCTL1 - DSPLCTL3) with each supporting TC1 - TC3, respectively. Keyboard actions at a given TC result in parameter blocks being built that contain the specific TC ID (1-3). These parameter blocks are passed to and interpreted by the associated application tasks. In response and in general, such application tasks should append the TC ID passed, to the Display Control task name (DSPLCTLN) required within the service request block when communications back to the TC is required. In doing so, the application has performed its only requirement in support of multiple TCs.

2.2 Display Generation

2.2.1 Overview

Utilizing a set of supplied macros, the applications programmer defines the text and control information necessary for building the display page (see related macros). These macros then become input to the P-E 3220 Macro Processor which in turn generates a source file suitable for input to the CAL assembler. In addition, an assembly listing of the generated assembler statements reflecting the defined display is also produced. Error messages may or may not be embedded within this listing. If present, such messages will be self-explantory.

Also, the display must be 'established' in order to be recognized as a viable display. This is accomplished through the P-E 3220 Task Establishing Task (TET). The object file created as output from the assembly phase becomes input to TET. The output file produced from this phase constitutes an established display and may be placed on the TC screen and/or the printer utilizing the PRTDSPLY off-line utlity (Ref. Sub-section 6.12).

Two CSS procedures are available to control the entire process of display generation and establishment. Figure E-2 shows the use of these procedures

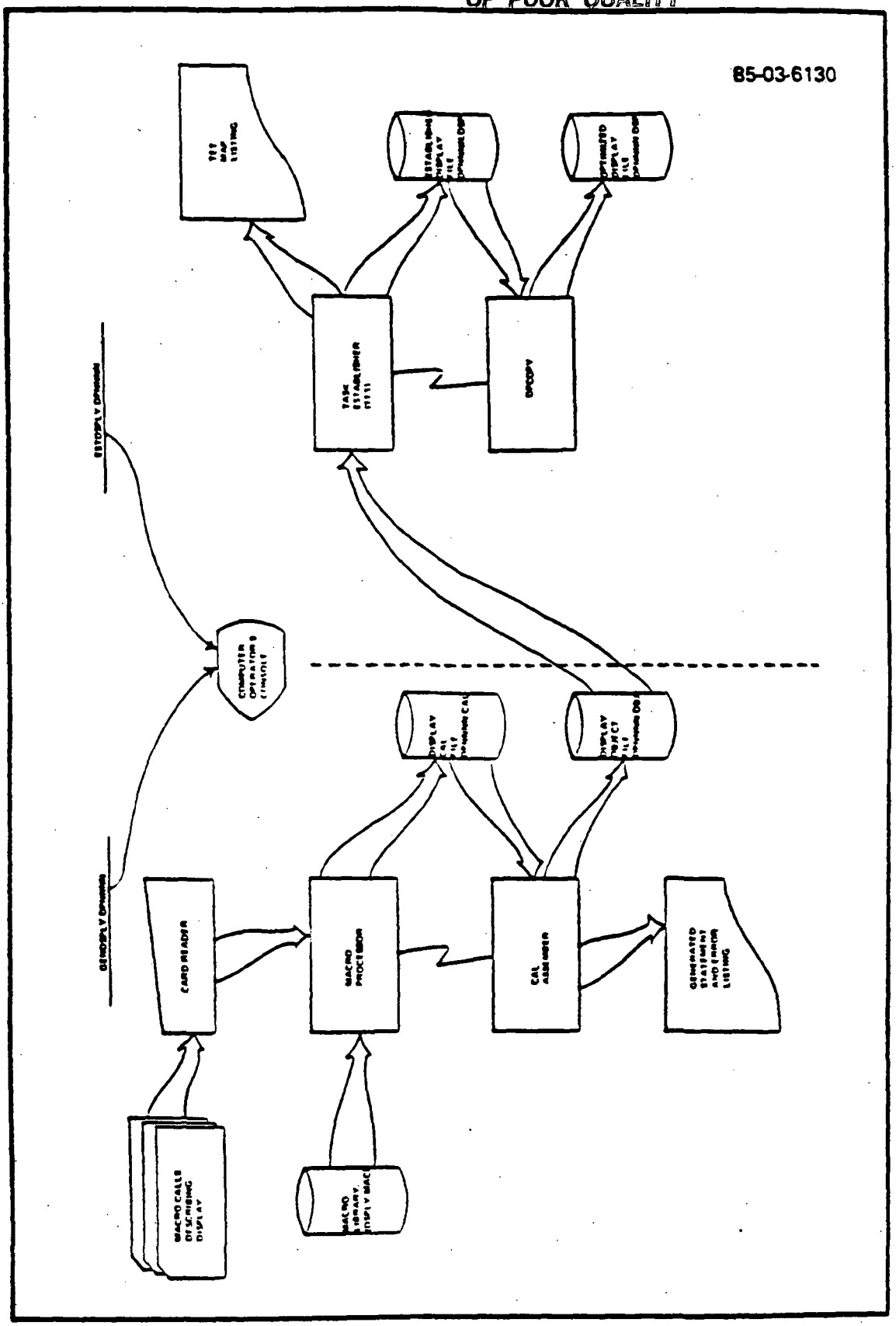


Figure E-2. Display Generation Overview

in the Display Generation and establishment process. It should be noted that the display must be established on volume MSM1 in order to be accessed by Display Control software.

2.2.2 Display Generation Macros

Six (6) macros are provided in support of Display Generation. These macros, described in subsequent paragraphs, as as follows:

a) TEXT macro

b) TEXTEND macro

c) OPTION macro

d) COMPOSE macro

e) LEGAL macro

f) DSPLYEND macro

2.2.2.1 TEXT Macro

One or more (up to 22) TEXT statements are used to communicate the text portion of a display. A TEXT statement has the following format:

```
C.C.1 TEXT  LN=N,LINE='———ASCII TEXT CHARACTERS———'
```

The line number (LN=N) is optional and may contain a value of 1-22. If LN=N is specified, it must be at least one larger than the previous line number (explicit or implied). If LN=N is not specified, then the implied line number is one greater than the previous line number. The implied line number of the first TEXT statement is one (1).

Text characters specified in the "LINE=" parameter must be enclosed in apostrophes and may include any valid ASCII characters with the exception

of apostrophes. Certain ASCII characters however, are used as special indicators. These characters and their usage is as follows:

- Underscore (_) - Indicates compose field
- At sign (@) - Indicates option line.
- Exclamation point (!) - Indicates fill-in field
- Dollar sign ($) - Indicates blinking field.

For the purpose of clarification, the definition of compose field, fill-in field, and option line is presented here along with appropriate rules/restrictions regarding the usage of such.

a) Compose Field

An area within a text line where variable TC keyboard data may be entered with the intention of passing such data to a GSMF applications task. Two or more consecutive underscore symbols shall be required of which the first shall be unavailable for keyboard entry. That underscore shall be overridden with an attribute character and shall be displayed as a blank. In addition, the character immediately following the last underscore shall be overridden for the same purpose as well. Finally, only one compose field per line is allowed and sixteen (16) per display.

b) Option Line

Any line in which the @ symbol is detected. The purpose of an option line is to indicate that an associated TC function key (1-16) may be used to signal a GSMF applications task and/or signal Display Control to refresh the TC screen with another display. The specific function key number (1-16) is computed based on the numerical order of prior option lines and relative to one (1). This number is then inserted into the line immediately following the '@' symbol. Either one or two text characters will be overridden. The '@' symbol should not appear in positions 78-80 of the line. Also, the maximum number of option lines permitted in a display is sixteen and compose fields are not allowed in option lines.

c) Fill-In Fields

Fill-In Fields are defined to be fields that application tasks may request the placement of fill-in data. Such a field is identified by arranging one or more consecutive '!' symbols in the proper position of the text line (i.e., through the "LINE=" operand). Multiple fill-in fields may be arranged on a single line or an entire line (80 characters) may encompass a single fill-in field. Also, there is no restriction with regard to compose fields or option lines. The maximum number of fill-in fields per display is 125. Fill-in fields will be addressed by applications tasks through a fill-in field number. Such numbers will be assigned by the display generation process and will be 1-N depending on numerical sequence of the specific fill-in field with regard to prior fill-in fields. Finally, unfilled fill-in fields will be displayed as blanks on the TC screen.

The '$' symbol shall be used within a text line(s) to indicate the beginning and end of a blinking area (field) on the screen. One byte will be (for each $) dedicated for the generated control code (blinker on/blinker off) and shall be displayed as a blank. There are no restrictions on the '$' usage except as follows:

1. No compose fields should be defined within a blinking field.
2. The '$' symbols must be paired.

2.2.2.2 TEXTEND Macro

The purpose of the TEXTEND macro is to indicate the end of TEXT macro processing for the generation of a given display.

The TEXTEND macro statement must immediately follow the last TEXT macro statement and has the following format:

```
C.C.10
    TEXTEND
```

No operands are required for this macro. Its primary function is to cause the generation of required control information such as total text area length and fill-in field addresses and lengths.

2.2.2.3 OPTION Macro

The purpose of the OPTION macro is to communicate the 'next task' and/or 'next display' that is to be associated with a particular option line (function key) within the display text.

There must be one OPTION macro specified for each option line within the text of the display. Such statement(s) must also immediately follow the TEXTEND macro. In addition, the OPTION macro statement(s) must be specified in the same numerical sequence for which the associated option line was positioned. That is, the first OPTION macro is associated with the first option line, the second OPTION macro with the second option line, etc.

The OPTION statement has the following format:

```
C.C.10
     OPTION TASKNAME,DSPL
```

The first operand is optional and if specified, must be the name of the task to receive control upon depression of the associated TC function key. The second operand is required and may be SAME, PREV, or a one to four digit display number. SAME may be specified to indicate to display control that the current display image is to remain on the TC screen. PREV indicates the previous display image is to replace the current image.

2.2.2.4 COMPOSE Macro

The purpose of the COMPOSE macro is to communicate the 'next task' and/or the 'next display' that is to be associated with a particular compose field (keyboard entered data) within the display text.

There must be one COMPOSE macro specified for each compose field within the text of the display. In addition, each COMPOSE macro statement must be paired with a corresponding LEGAL macro statement (see LEGAL macro). The COMPOSE/LEGAL pair grouping must immediately follow any OPTION statement(s) that may be present. Each COMPOSE/LEGAL pair must also be specified in the same numerical sequence for which the associated compose field was positioned. That is, the first COMPOSE/LEGAL pair is associated with the first compose field, the second COMPOSE/LEGAL pair with the second compose field, etc.

The COMPOSE statement has the following format:

```
C.C.10
     COMPOSE TASKNAME,DSPL
```

where the first operand is required and must be the name of the task to receive control upon TC keyboard data being entered into the associated compose field. The second operand is required and may be SAME, PREV, or a one to four digit display number, i.e., next display. SAME may be specified to indicate to Display Control that the current display image is to remain on the TC screen. PREV indicates the previous display image is to replace the current image.

2.2.2.5 LEGAL Macro

The purpose of the LEGAL macro is to indicate compose field parameter limits and compose field validation and conversion requirements.

There must be one LEGAL macro specified for each compose field within the text of the display. In addition, each LEGAL macro statement must be paired with, i.e., immediately follow, an associated COMPOSE statement (see COMPOSE macro). Also, the COMPOSE/LEGAL pair must be specified in the same numerical sequence for which the associated compose field was positioned in the text.

A compose field, for the purpose of validation and conversion, consists of one or more parameter fields. For the purpose of validation, each parameter may consist of one or more sub-parameter fields. With this in mind, the LEGAL statement then requires the following format:

C.C.10

LEGAL $(P_1),(P_2),\ldots,(P_n)$

P_1 is required and represents the operand describing the validation and conversion necessary for the first (or only) parameter field within the associated compose field. P_2-P_n would describe additional parameter fields.

Each P operand has the following format:

$$(\alpha,\beta_1,n_1,[\gamma_1][,\beta_2,n_2,[\gamma_2]\ldots])$$

where: α is the type of parameter to be passed to the application task.

β is the type of data entered (sub-parameter type) for this α field.

n is the number of characters in a sub-field.

γ is the legality check data for a sub-field.

α may be:

- C for characters.
- F for a fixed point number (integer only).
- E for a single precision floating point number.
- D for a double precision floating point number.
- H for hexadecimal digits 0-9 and A-F.

β may be:

- A for alphabetic characters
- B for binary digits (0 or 1).
- D for decimal digits (0-9).
- M for all type characters.
- H for hexadecimal digits 0-9 and A-F.

γ is either δ or $(\delta_1;\delta_2)$

δ is an acceptable field contents.
δ_1 is the lower limit of an acceptable field contents.
δ_2 is the upper limit of an acceptable field contents.

Display Control shall perform validation of sub-field data entered based on γ data. This validation for a field of proper type shall be of the form:

LEGALITY CHECK = $(\gamma_1[,\gamma_2] \ . \ . \ .[,\gamma_n])$

If the field passes one of the content or range checks, it shall be accepted.

Finally:

1 + Σn = number of '_' (underscore) symbols in a given compose field.

An example of a LEGAL statement is:

```
C.C.10                                                                C.C.72
     LEGAL (C,A,1,(A,B)),(C,A,1,(H,F,Z)),(C,M,2,(S1,S2)),                  Z
    (E,M,1,(+,-),D,3,,M,1,,.,D,1,,M,2,(E+,E-),D,2),                        Z
    (C,M,2,(T1;T7,X1;X7))
```

This LEGAL statement defines the validation/conversion required for a compose field with five parameters. The first three are character fields, the fourth is a floating point number, and the fifth is a character field. The first parameter is one alphabetic character that can be either A or B. The fourth parameter describes a floating point number and consists of seven sub-fields. A typical entry for this parameter might be:

+123.4E+11

The fifth parameter is character data that can be within the range T1 to T7 or X1 to X7.

2.2.2.6 DSPLYEND Macro

The purpose of the DSPLYEND macro is to indicate the end of the Display Generation process.

The DSPLYEND macro statement must be the last macro statement within the group. The statement has the following format:

```
C.C.10
      DSPLYEND
```

No operands are required for this macro. Its primary function is to cause initiation of final validation on the overall Display Generation process. Specifically, the number of option lines vs. number of OPTION statements and the number of compose fields vs. number of COMPOSE/LEGAL pairs are checked.

2.2.3 Display Generation Example

The following example illustrates the necessary control statement required to generate a typical test console display.

```
                                          At least one blank
C.C.10   OPERATION  .  OPERAND(S)

         MLIBS         7              Defines macro library L.U.

         TEXT       .  '...TEXT...'   } Defines text, etc.
                    .                 } (1-22 statements)
                    .
         TEXT          '...TEXT...'   }

         TEXTEND                      Defines end-of-text

         OPTION        TASK,DSPL
                    .                }
                    .                } One statement for each
                    .                } option line
                    .                }
         OPTION        TASK,DSPL

         COMPOSE       TASK,DSPL
         LEGAL         (LEGAL DATA)
                    .                }
                    .                } One compose/legal statement
                    .                } pair for each compose
                    .                } field
                    .                }
         COMPOSE       TASK,DSPL
         LEGAL         (LEGAL DATA)


         DSPLYEND                     Defines end-of-display

         END                          Required for assembler
```

It should be noted that the very first (MLIBS) and very last (END) statements are not display generation macros; however, such statements are required input for proper operation.

2.3 Display Communications

2.3.1 Overview

Display communications in GSMF involves two distinct areas. These are:

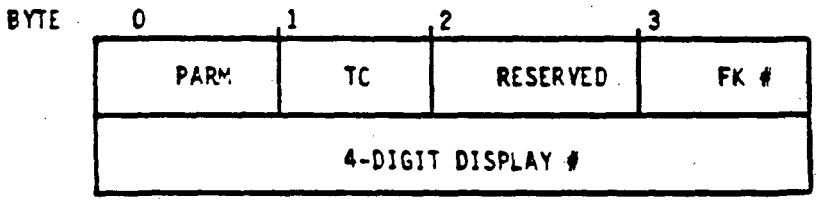
a) Communicating TC keyboard entries to associated target application tasks.

b) Communicating TC screen request made by application tasks.

Item 'a' above may be initiated by either selecting a menu option (function key) or sending compose field data. In both cases, a parameter list is constructed and the address passed to the application task via the SVC 6 Q-PARM facility. It is up to the application task to decode and act accordingly on the passed parameter list. Item 'b' above is initiated by the application task through the same facility and involves the passing of a request block to a specific Display Control task. These Display Control communications services are described in detail in subsequent paragraphs.

2.3.2 Menu Option Handling

Upon fielding the menu option selection (function key), Display Control S/W examines associated display page control information to determine whether communications with a target application task is required. If not, then 'next page' processing is performed and no interface to an application task is established. If an application task is associated however, then a 'function key' parameter list, as described in Figure E-3, is constructed and passed to this task. It is assumed that the applications task is capable of receiving a parameter list (i.e., conforms to the rules of a queue service interrupt handling task as described in Section 3 of the Interdata OS/32-MT Program Reference Manual, Pub. No. 29-613).



FIELD DESCRIPTIONS:

1. PARM ID. - X'02', identifies parameter list type.
2. TC ID. - ID (1-n) of associated test console.
3. FK # - Binary number (1-16) of function key that was depressed.
4. 4-DIGIT DISPLAY # - Display identifier for which this parameter list is associated.

BB-03-6131

Figure E-3. FUNCTION Key Parameter List

If an error is detected on attempting to pass the parameter, the Display Control S/W assumes the associated task is not in memory. In this case, Display Control S/W will attempt to load, start the task, and again pass the parameter. If an error is detected during this phase, an appropriate Class B message is output and no further action is taken.

2.3.3 Compose Field Data Handling

As with menu option handling, the communications to the application task for a compose field entry is performed in exactly the same manner. However, prior to construction of the associated parameter list, validation and conversion is performed on the entered compose data according to the LEGAL data specified at display generation time.

Figure E-4 illustrates the parameter list generated and passed to the applications task. In addition, Tables E-1 and E-2 are provided to further clarify the automatic validation and conversion performed on a compose field entry.

Both parameter lists (compose data and function key) are built in storage obtained from CMPOOL (Ref. 5.3.2). It is the application task's responsibility to properly release this storage once processing is complete.

2.3.4 Display Applications Services

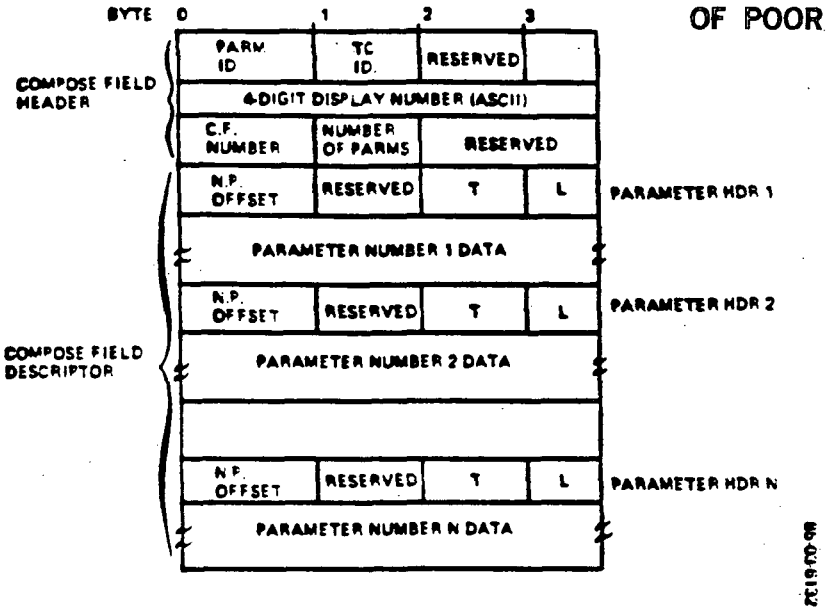
The Display Control software supports a variety of display related and application task request services. These include the following:

1) Fetch display page to memory

2) Display specified page image

3) Update display page (single fill-in field)

4) Update display page (multiple fill-in fields)



FIELD DEFINITIONS:

1. PARM ID. - X'01', identifies parameter list type.
2. TC ID. - ID (ASCII 1-n) of associated test console.
3. 4-DIGIT DISPLAY # - DISPLAY IDENTIFIER (ASCII 0000-1999)
4. C.F. # - Compose field number (binary 1-n) with respect to numerical sequence of entire set of compose fields in display.
5. # PARMS - Number of parameters (binary 1-n) comprising associated compose field.
6. N.P. OFFSET - Next parameter offset (binary), i.e., value to be added to address of this field in order to address next parameter hdr. A zero (X'00') value indicates this is last parameter for associated compose field.
7. T (TYPE) - Type of parameter (ASCII); may be C-character, F-fixed point, D-Dbl prec. floating pt., E-Sgl prec. floating pt., H-Hexadecimal.
8. L (LENGTH) - Number of bytes (binary) in parameter.
9. PARAMETER DATA - Keyboard entered data in format designated by conversion requirements stipulated when display was generated.

Figure E-4. COMPOSE Field Entry Parameter List

Table E-1. Display Compose Field Sub-parameter Validation

TYPE	VALIDATION
A-Alphabetic	Each character of input sub-parameter is tested for alpha (A-Z) restricted.
B-Binary	Each character of input sub-parameter is tested for binary (0 or 1) restricted.
D-Decimal	Each character of input sub-parameter is tested for decimal (0-9) restricted.
H-Hexadecimal	Each character of input sub-parameter is tested for hexadecimal (0-F) restricted.
M-Mixed	All characters are valid. No automatic validation is performed unless an actual or range comparator(s) was specified when display was generated.

89-0316133

Table E-2. Display Compose Field Parameter Conversions

TYPE	CONVERSION
C-Character	No conversion done. Character string aligned on fullword boundary and passed.
F-Fixed Pt. * (Integer only)	Value must be an integer with an optional sign only. The value is converted to a 2's complement binary number and passed in a fullword.
E-Single-Precision Flt. Pt.	Value may be a signed integer, fraction, or mixed number. Optionally, exponent may be specified. The value is converted to a floating pt. number (characteristic and fraction) and passed in a fullword.
D-Double - Precision Flt. Pt.	Same as "E" except converted value is a 64-bit number and is passed as a double-word.
H-Hexadecimal	Each input character is translated to its corresponding hex value; i.e., 0-F=00-0F. The 4 MSBS of each character are then removed, the string packed and passed aligned on fullword boundary.

89-03-0134

* Fixed point fractions are not supported, i.e., only integers.

5) Initialize fill-in fields (all)

6) Output Class A message

7) Output Class B message

8) Output condition/error number

9) Compose field initialize (single)

10) Compose field initialize (all)

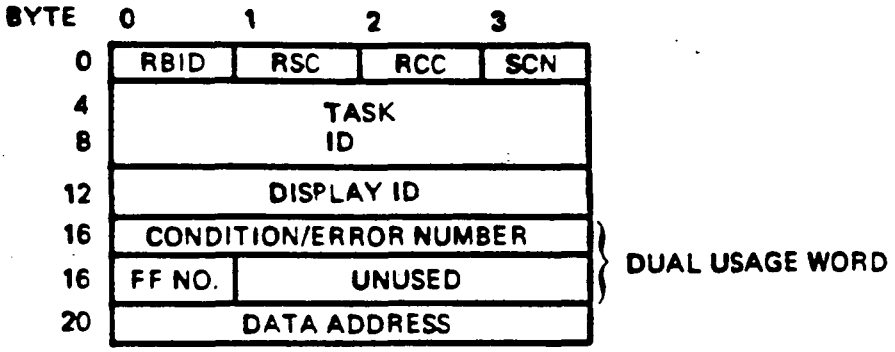
As previously mentioned, all requests are made by passing an appropriately formatted request block. This request block must be acquired from CMPOOL. Therefore, the application task must be properly linked with the CMPOOL task common. Subsequent paragraphs describe each of the various services in detail. In addition, Figure E-5 and Table E-3 describe the required request block. It should be noted that the modifier bit (bit 1) of request service code (RSC) is used for requests 3, 4, 6, and 7. If set to a 1, it will indicate that Display Control S/W is to free the associated data area (message or fill-in data area) pointed to by the R.B. data addr. field (RB+20).

2.3.4.1 Fetch Display Page to Memory

Execution of this service will cause the specified display page to be fetched into the display stack area of memory unless it is already in the area. Use of this service is limited since it is performed automatically on all other service requests involving a specific display page.

2.3.4.2 Display Specified Page

Execution of this service will cause the specified display page to be fetched to stack area (if not already there) and placed on the TC screen being controlled by the target display control task (i.e., overrides the existing image).



FIELD		MEANING
1.	RBID	- X'03', identifies this control block as a Display Control request block
2.	RSC	- Request service code. Identifies specific Display Control function to be performed (see Table E-3).
3.	RCC	- Request completion code. Indicates completion type (successful, unsuccessful) of request.
4.	SCN	- Service completion notification.
5.	TASK ID	- Eight-byte name of requesting task. Required if TC indicates queue and/or release.
6.	DISPLAY ID	- Four-digit ID (name) of target display (required for all requests except 6, 7, and 8).
7.	CONDITION/ERROR #	- Four-digit ASCII error number. Used to indicate specific errors or conditions detected by applications.
8.	FF #	- Fill-in field number, binary 1-N. (Also used to indicate single compose field to be reinitialized.)
9.	DATA ADDRESS *	- Address of fill-in data or one-line message.

* As with this request block, the data address if used must point to an area within the global common COMPOOL. Furthermore, the area must be fullword aligned with the first two bytes containing the length (binary) of the succeeding data.

Figure E-5. Display Control Request Block

Table E-3. Display Service Request Block Cross-Reference

SERVICE	RSC CODE	R.B. FIELDS REQUIRED: DSPLY ID	C/E#	FF #	DATA ADDR	WHEN REQUESTER NOTIFIED
BITS	01234567					
• Fetch Display Page	0M000001 (1)	X				After display page fetched
• Display Specified Page	0M000010 (2)	X				After display page is on TC screen
• Update Display Page (single fill-in field)	0M000011 (3)	X		X	X	After display page on TC screen updated including any broadcast
• Update Display Page (multiple fill-in fields	0M000100 (4)	X		X	X	Same as single F.F. update above
• Initialize Fill-in Fields	0M000101 (5)	X				Same as single F.F. update above
• Output Class A Message	0M000110 (6)				X	After message is acknowledged by operator (MAC command)
• Output Class B Message	0M000111 (7)				X	After message is acknowledged by operator (any K.B. action)
• Output Cond/ Error Number	0M001000 (8)		X			After number is acknowledged by operator (MAC command)
• Compose Field Initialize (single)	0M001001 (9)	X		X		After display page on TC screen updated including any broadcast
• Compose Field Initialize (all)	0M001010 (A)	X				Same as single C.F. initialize above

85-03-6136

2.3.4.3 Update Display Page (Single Fill-in Field)

Execution of this service will cause the specified display page to be fetched to stack area (if not already there). The specified field (FF#) is then updated with the data pointed to by the request block (data addr.+2). If the page is also being viewed on any active TC screens, then those images are updated as well (i.e., broadcast). Validation is performed prior to the update operation to insure the specified F.F. # and data length (1st H/W of data area) corresponds to a valid fill-in field number or length for the display. The request block RCC field is set to X'04' or X'08' respectively if the validity of these items cannot be established and assuming TC indicated post RCC as part of service complete notification.

2.3.4.4 Update Display Page (Multiple Fill-in Fields)

Execution of this service is handled exactly as with a single fill-in field update request with the following exceptions:

a) F.F.# in request block is used to indicate the 1st of N contiguous fields.

b) Data length (1st H/W of data area) specifies total composite length of contiguous fields to be updated. This length must be the exact length and be followed by the exact amount of data required to fill all N fields.

2.3.4.5 Initialize Fill-in Fields (All)

Execution of this service is performed for the specified display page much like the multiple fill-in field update service. However, no F.F. # nor data address is required in the request block. All TC screens with target display page being viewed will be affected (i.e., service is broadcast). Also, no single fill-in field initialize service is provided since this can be accomplished with the single fill-in field update request by simply supplying ASCII blanks for the data.

2.3.4.6 Output Class A Message

Execution of this service will cause the specified message to be scheduled for output to the O-lines of all active TC screens. Scheduling is on a FIFO basis; however, any active Class B message will be overridden. If the message is greater than 72 bytes it is truncated. As with fill-in data, the message length and succeeding message is addressed through the data address field of the request block.

2.3.4.7 Output Class B Message

Execution of this service is performed exactly as with a Class A message except for the broadcast feature. The Class B message is output only to the TC screen being controlled by the target display control task.

2.3.4.8 Output Condition/Error Number

Execution of this service will cause the specified error number (Ref. 2.1.8) to be scheduled for output on a FIFO basis to the TC screen being controlled by the target display control task. If a user intends to use this service in a new application task, then consultation with the system programmer is required since the number must be assigned and message text must be placed into the off-line file (Ref. 2.4).

2.3.4.9 Initialize Compose Field (Single)

Execution of this service will cause the compose field, as specified by the F.F. # field of the request block, to be reset to underscores (_) for the target display page. If this page is being viewed on any TC screen, then it is reset on those screens as well (i.e., broadcast). The compose field specified must be a valid compose field for the specified display page. RCC=X'04' is posted if not and assuming post RCC was specified by the SCN field.

2.3.4.10 Initialize Compose Field (All)

Execution of this service is handled by the Display Control software exactly as in 2.3.4.9 except that all compose fields for target display page are reset. The compose field number (F.F. # field) in request block is not required. Use of this service as well as the single field reset service is limited since resetting a compose field is not required before reentry of compose data in that field. Also, the operator can reset compose fields on a page being viewed at any time via shifted function key four.

2.4 Error Number Message Text Generation

This sub-section defines the method by which error (or unusual condition) message text is generated to coincide with display error numbers (Ref. 2.1.8).

These messages, intended to provide meaning for error numbers, are generated and maintained in a disk file on volume MSM1: and named ERRORMSG.TXT.

This file has the following characteristics:

a) File has been initialized to 1000 records (blocks), each 256 Bytes in length.

b) Each record contains message text for a corresponding error number in the range of 000-999.

c) Records that have not been updated contain text indicating NOT DEFINED.

d) Message text from record will be used as fill-in data for display 'DP0000' by Display Control in response to an ID I-line command, Ref. 2.1.7.3 and Figure E-6.

A utility update program (MFUPDATE) is provided on the P-E 3220 in conjunction with the CSS procedure MFUP to perform ERRORMSG.TXT file

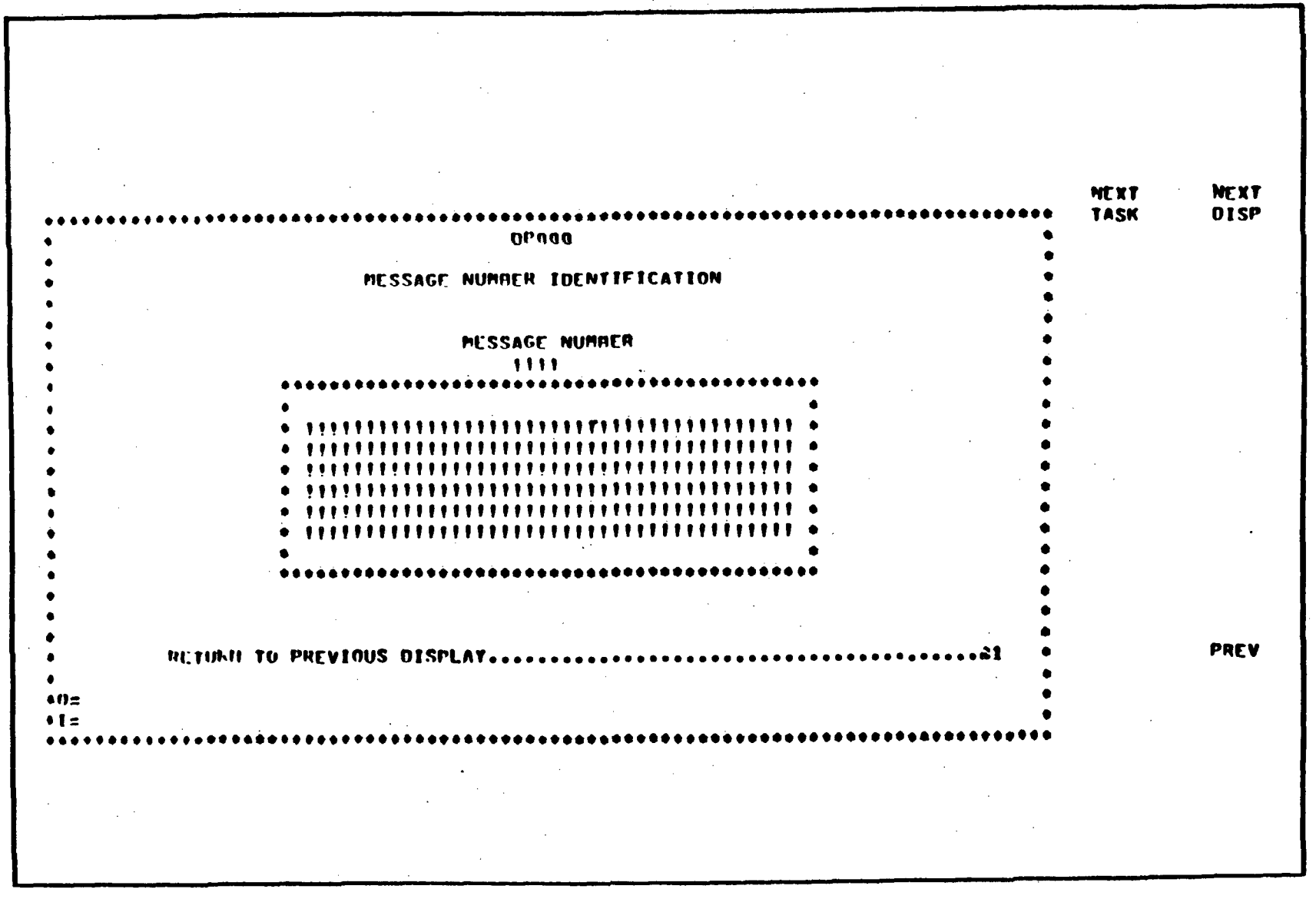
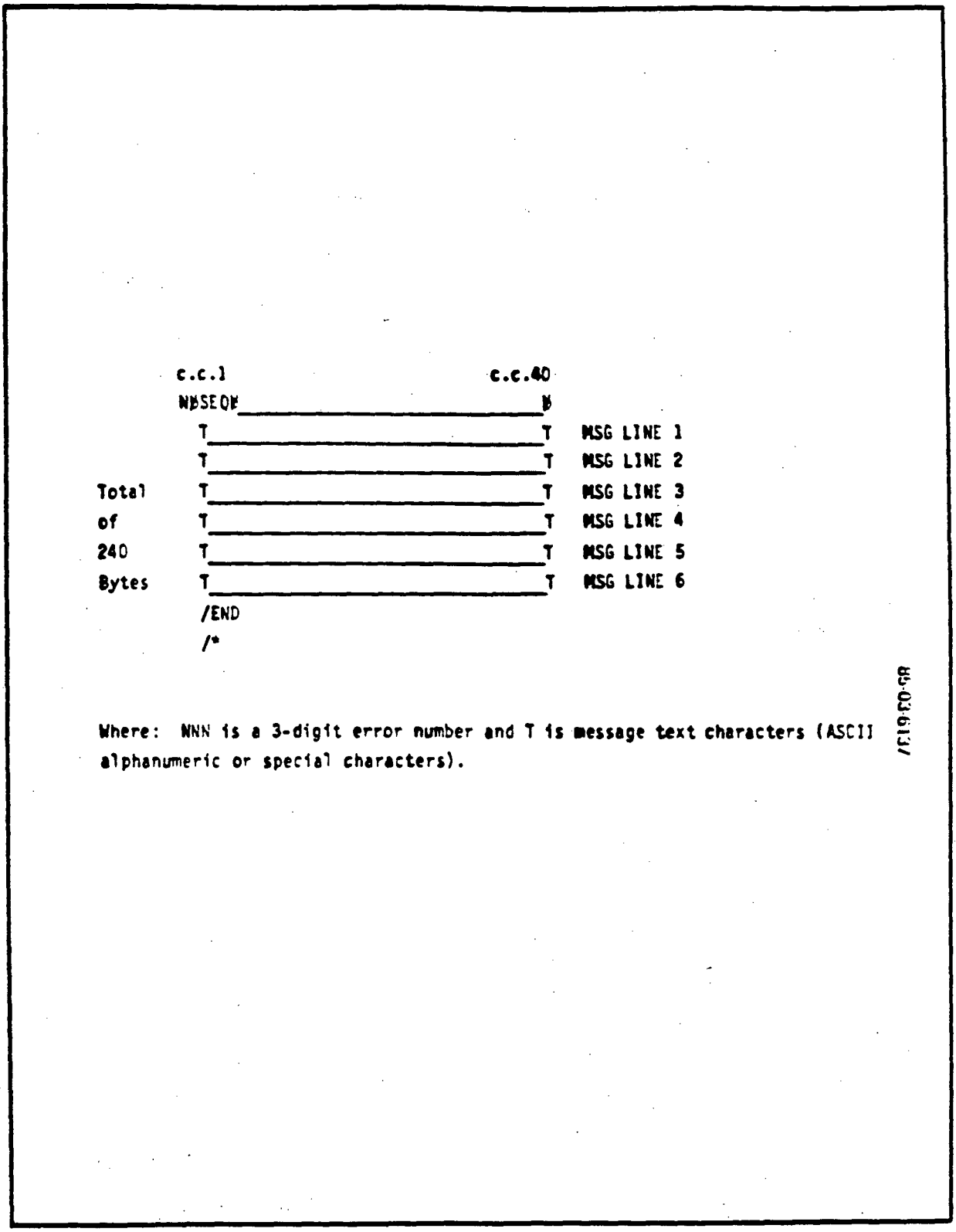


Figure E-6. Display Control Request Block

maintenance in non-realtime mode. This program and procedure require a card image input file that adheres to the format given in Table E-4.

At least one MSG line (statement) is required. Also, the set of input statements may consist of multiple subsets to allow updating of multiple records. The /* statement denotes final end of input. In addition, records may be updated in any order.

Table E-4. Program MFUPDATE and Procedure MFUP Card Image Input File Example



APPENDIX F

GSMF DATA BASE GENERATION

APPENDIX F

GSMF DATA BASE GENERATION

1. OVERVIEW

Prior to the actual generation of the GSMF data base certain procedures must be executed. These procedures will properly create data which will be used in the creation of the GSMF Data Base. Basically these are five areas which must be addressed by the user to satisfy these apriori data requirements. Thse areas are as follows:

1) SWID Relations Forms
2) SWID Pairs File
3) SWID Data File
4) Run Documentation File
5) SWID Initial Data File

The data output from the aforementioned areas shall be generated within the IBM 4381 VM/CMS environment. This allows the user to be prompted for inputs in an interactive conversational atmosphere. The inputs required are basic and minimal and are tabulated in Figure F-1. It is assumed that the user is familiar with the IBM 4381 VM/CMS interactive system.

The following is an explanation of each of the aforementioned areas and the steps required for their respective data generator.

NOTE: All steps mentioned in the following paragraphs must be executed by the user on numeric ascending sequence.

1.1 SWID Relations Forms/SWID Pairs File

Once the SWID Relations forms as explained in Appendix C have been generated, Step 1 of Figure F-1, then the SWID pairs file for a given Stimuli can be defined. The SWID pairs file as explained in Appendix D defines Stimuli/Measurement SWID relationships and their incorporation into the SWID Relations form. The process used to accomplish this is defined in steps 2-4 of Figure F-1 which will selectively display the SWID Relations form for a given ATE subtable on the IBM terminal for editing and then save the new version on IBM 4341 for disk eventual GSMF data base addressing.

Step	Procedure	Function	Inputs	Reference
1	SWDPART	GENERATE A SWID RELATIONS FORM FOR A GIVEN ATE SUBTABLE IN IBM 4341 DISK DATA SET GSMF.SWDPRS.PARTS (MS#TBL) WHERE # IS THE MISSION NUMBER TBL IS THE ATE SUBTABLE NAME	SWD PART # TBL	APPENDIX C
2	SPRSIN	OBTAINS SWID PAIRS MEMBER TO BE UPDATED FROM OS DATA SET GSMF.SWDPRS. PARTS (MS#TBL) WHERE # IS THE MISSION NUMBER TBL IS THE ATE SUBTABLE NAME	SPRSSIN # TBL	APPENDIX F
3	SPRSMEM	PUT UPDATED SWID PAIRS NUMBER IN OS DATA SET GSMF.SWDPRS.PARTS (MS#TBL) WHERE # IS THE MISSION NUMBER TBL IS THE ATE SUBTABLE NAME	SPRSMEM # TBL	APPENDIX F
4	SWDPRS	PUT ALL SWID PAIRS MEMBERS IN OS DATA SET GSMF.SWDPRS.PARTS (MS#) WHERE # IS MISSION NUMBER	SWDPRS #	APPENDIX F
5	SMIDIN	OBTAINS SMID DATA MEMBER TO BE UPDATED FROM OS DATA SET GSMF.SMID. DATA (M#); WHERE # IS MISSION NUMBER	SMIDIN #	APPENDIX F
6	SMIDMEM	PUT UPDATED SMID DATA FILE MEMBER IN IBM DISK DATA SET GSMF.SMID.DATA (M#); WHERE # IS THE MISSION NUMBER	SMIDMEM #	APPENDIX F
7	RDOC IN	OBTAINS RUN DOCUMENTATION MEMBER TO BE UPDATED FROM IBM DISK DATA SET GSMF.RUN.DOC (M#); WHERE # IS THE MISSION NUMBER	RDOC IN #	APPENDIX F
8	RDOCMEM	PUT UPDATED RUN DOCUMENTATION MEMBER IN IBM DISK DATA SET GSMF.RUN.DOC (M#); WHERE # IS THE MISSION NUMBER	RDOCMEM #	APPENDIX F
9	INITDAT	OBTAINS SWID INITIAL DATA FILE MEMBER TO BE UPDATED FROM IBM 4341 DISK DATA SET GSMF.SWDINIT.DATA (MS#); WHERE # IS THE MISSION NUMBER	INITDAT #	APPENDIX F
10	INITDATM	PUT UPDATED SWID INITIAL DATA FILE MEMBER IN IBM 4341 DISK DATA SET GSMF.SWDINIT.DATA (MS#) WHERE # IS THE MISSION NUMBER	INITDATM #	APPENDIX F
11	GSMFDB	GENERATE GSMF DATA BASE NOTE: # IS MISSION NUMBER	GSMFDB #	APPENDIX F

Figure F-1. Setup Mode VM/CMS Procedures

12	ESFETC	GENERATE ECOS AND SCOS SDF FETCH COMMANDS ON LEV III/II FORMAT CARDS AND ENTER THEM IN IBM 4341 DISK DATA SET GSMF.FETCH.CARDS (MS#); WHERE # IS THE MISSION NUMBER	ESFETC #	APPENDIX G
13	FETCOM	GENERATES ECOS AND SCOS SDF FOR; MATTED LEVEL III/II FETCH COMMANDS FOR GCOS LOAD INTO GCID AND ENTERS THEM IN IBM 4341 DISK DATA SET FMNSDK1D.MISXXX.VFESMF.ATE.FETCOM; WHERE XXX IS MISSION NUMBER RIGHT JUSTIFIED	FETCOM XXXX	APPENDIX G
14	SINBINTP	GENERATES ECOS AND SCOS SDF FORMATTED LEVEL II/II FETCH COMMANDS ON GCOS LOAD TAPE	SINBINTP	APPENDIX G
15	GSMFPRT	GENERATES PRINT IMAGE OF ALL FILES ON THE GSMF DATA BASE	GSMFPRT	APPENDIX G

Figure F-1. Setup Mode VM/CMS Procedures (Continued)

Figures F-2 and F-3 represent an example of the VM/CMS interactive conversation using steps 1, 2, 3 and 4 with all inputs underlined.

1.2 Generation of SWID Data File

The SWID data file contains definitions of all end-items not contained in the ATE Spacelab Data Base (SLDB). Also contained in the SWID data file are SCOS and ECOS Stimuli SWID (not contained in the ATE SLDB) which affect ATE measurements. Initially this data file must be created manually as a VM/CMS file. Figure F-4 illustrates this date file and its associated fields.

It must be noted that all SWIDs will be assigned a negative SWID number to differentiate it from all defined end-items on the ATE SLDB.

The procedures used to maintain this data file under VM/CMS are defined in steps 5 and 6 of Figure F-1 which will selectively display the SWID data file for a given Spacelab mission upon an IBM terminal for editing and then save the new version on IBM disk for eventual GSMF data base addressing. Figure F-5 represents an example of the VM/CMS interactive conversation using steps 5 and 6 with all inputs underlined.

1.3 Run Documentation File

The Run Documentation file will provide configuration management for the GSMF data base. The components (Free Format) of this file are illustrated in Figure 3.2-4. Initially this file must be created manually as a VM/CMS file. The procedures used to maintain this file under VM/CMS are defined in steps 7 and 8 of Figure F-1 which will selectively display the run documentation file for a given Spacelab mission upon a IBM terminal for editing and then save the new version on IBM disk for eventual GSMF data base processing. Figure F-6 represents an example of the VM/CMS interactive conversation using steps 7 and 8 with all inputs underlined.

1.4 The SWID Initial Data File

The SWID initial data file as defined in Section 3.2.2.6 and illustrated in Figure F-7 shall be initially created for each Spacelab mission as a VM/CMS file via manual inputs. This VM/CMS file shall contain initital values in integer and floating point formats.

The procedures to maintain this data file under VM/CMS are defined in steps 9 and 10 of Figure F-1 which will selectively display the SWID initial

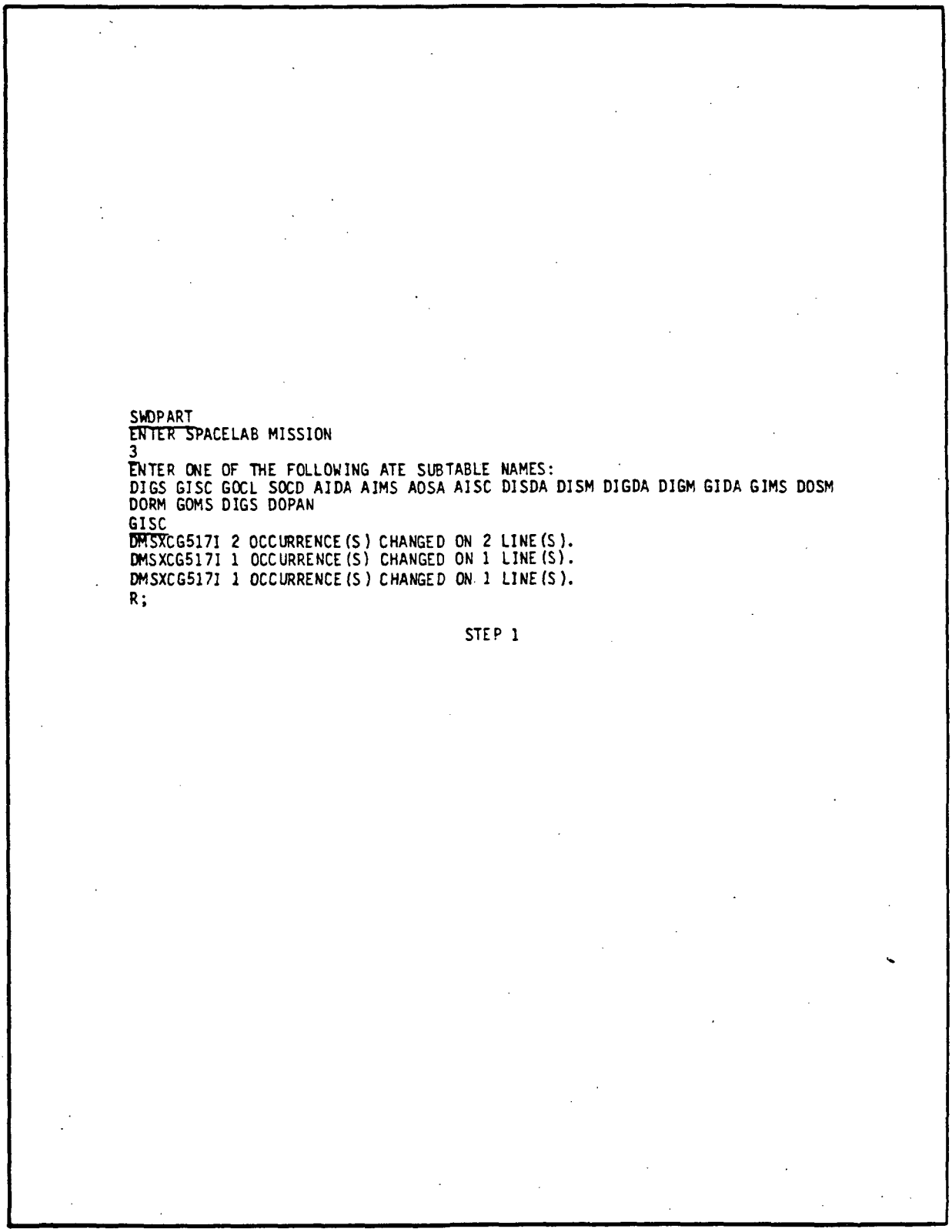


Figure F-2. SWID Relations VM/CMS Procedure

```
SPRSIN
ENTER SPACELAB MISSION
3
ENTER SUBTABLE NAME
GISC
DMSACC723I M (579) R/O - OS
DASD 579 DETACHED
R;
```

STEP 2

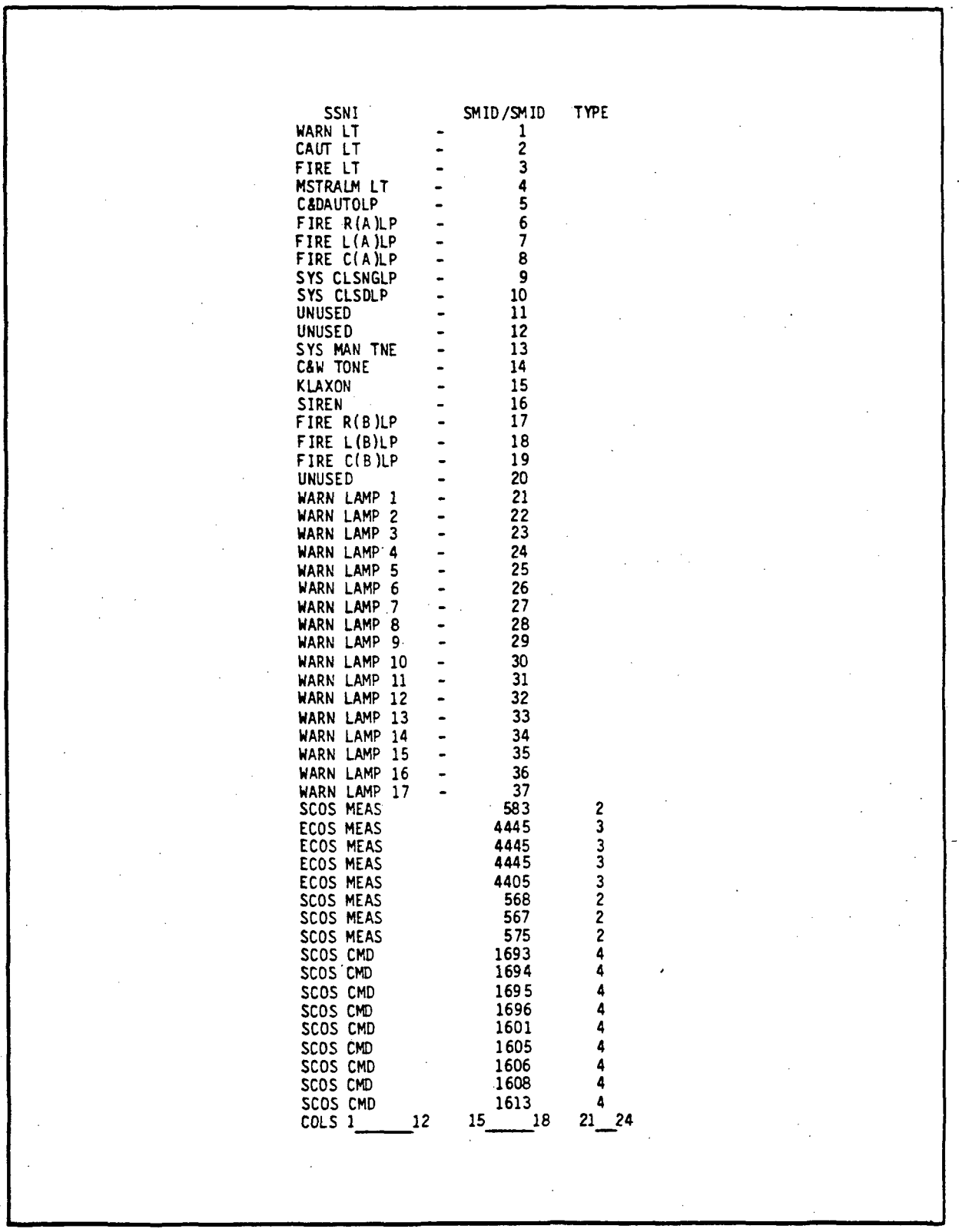
```
SPRSMEM
ENTER SPACELAB MISSION
3
ENTER SUBTABLE NAME
GISC
DMSXCG517I 1 OCCURRENCE(S) CHANGED ON 1 LINE(S).
DMSXCG517I 1 OCCURRENCE(S) CHANGED ON 1 LINE(S).
DMSXGT564W EOF REACHED
R;
SMIDMEM
ENTER SPACELAB MISSION
3
DMSXCG517I 2 OCCURRENCE(S) CHANGED ON 2 LINE(S).
DMSXGT564W EOF REACHED
R;
```

STEP 3

```
SWDPRS
ENTER SPACELAB MISSION
3
DMSXCG517I 1 OCCURRENCE(S) CHANGED ON 1 LINE(S).
DMSXCG517I 20 OCCURRENCE(S) CHANGED ON 20 LINE(S).
R;
```

STEP 4

Figure F-3. SWID Pairs VM/CMS Procedures



```
   SSNI                SMID/SMID   TYPE
WARN LT          -          1
CAUT LT          -          2
FIRE LT          -          3
MSTRALM LT       -          4
C&DAUTOLP        -          5
FIRE R(A)LP      -          6
FIRE L(A)LP      -          7
FIRE C(A)LP      -          8
SYS CLSNGLP      -          9
SYS CLSDLP       -         10
UNUSED           -         11
UNUSED           -         12
SYS MAN TNE      -         13
C&W TONE         -         14
KLAXON           -         15
SIREN            -         16
FIRE R(B)LP      -         17
FIRE L(B)LP      -         18
FIRE C(B)LP      -         19
UNUSED           -         20
WARN LAMP 1      -         21
WARN LAMP 2      -         22
WARN LAMP 3      -         23
WARN LAMP 4      -         24
WARN LAMP 5      -         25
WARN LAMP 6      -         26
WARN LAMP 7      -         27
WARN LAMP 8      -         28
WARN LAMP 9      -         29
WARN LAMP 10     -         30
WARN LAMP 11     -         31
WARN LAMP 12     -         32
WARN LAMP 13     -         33
WARN LAMP 14     -         34
WARN LAMP 15     -         35
WARN LAMP 16     -         36
WARN LAMP 17     -         37
SCOS MEAS                 583       2
ECOS MEAS                4445       3
ECOS MEAS                4445       3
ECOS MEAS                4445       3
ECOS MEAS                4405       3
SCOS MEAS                 568       2
SCOS MEAS                 567       2
SCOS MEAS                 575       2
SCOS CMD                 1693       4
SCOS CMD                 1694       4
SCOS CMD                 1695       4
SCOS CMD                 1696       4
SCOS CMD                 1601       4
SCOS CMD                 1605       4
SCOS CMD                 1606       4
SCOS CMD                 1608       4
SCOS CMD                 1613       4
COLS 1_____12    15____18    21__24
```

Figure F-4. SMID Data File Listing

ORIGINAL PAGE IS
OF POOR QUALITY

```
SMIDIN
ENTER SPACELAB MISSION
3
DMSACC723I M (579) R/O - OS
DASD 579 DETACHED
R;
```

STEP 5

```
SMIDMEM
ENTER SPACELAB MISSION
3
DMSXCG517I 2 OCCURRENCE(S) CHANGED ON 2 LINE(S).
DMSXGT564W EOF REACHED
R;
```

STEP 6

Figure F-5. SMID Data File VM/CMS Procedures

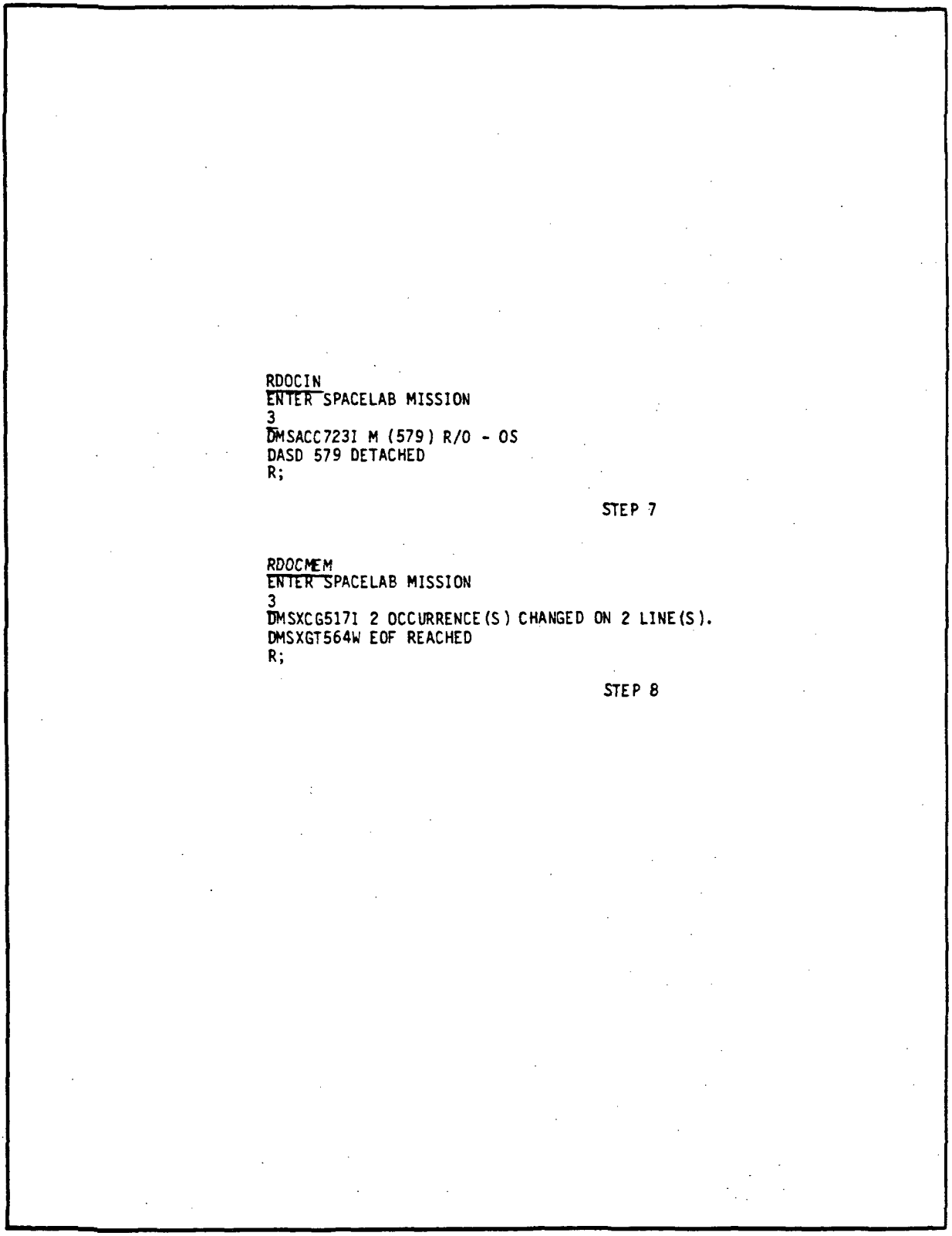


Figure F-6. Run Documentation VM/CMS Procedures

	SWID	DATA
	7900	1
	7901	7.0
	7902	0
	7903	1
	7904	20.0
	7905	7.0
	7906	0
	7907	1
	7908	20.0
	7909	30.0
	7910	20.0
	7911	0
	7912	110.0
	7913	110.0
	7914	110.0
	7915	1
	7916	0
	7917	0
	7918	0
	7919	0
	7920	0
	7921	0
	7922	0
	7923	0
	7931	0
	7932	0
	7933	0
	7934	0
	7935	0
	7936	0
	7937	20.0
	7938	20.0
	7939	1.0
	7940	1.0
	7941	0
	7942	0
	7943	0
	7944	0
	7946	0
	7947	0
	7949	1
	7951	1.0
	7961	0
	7962	0
	8729	1
	8730	1
	8731	1
	8793	1
	8794	1
	8805	3.35
COLS.	4__8	11_15

Figure F-7. SWID Initial Data Listing

data file for a given Spacelab mission upon an IBM terminal for editing and then save the new version on an IBM 4341 disk for eventual inclusion in the GSMF data base. Figure F-8 represents an example of the VM/CMS interactive conversation using steps 9 and 10 with all inputs underlined.

When steps 1-10 have been exercised, all the preliminary data has been created and GSMF data base generation can begin. Step 11 in Figure F-1 defines the procedure and VM/CMS user inputs necessary for GSMF data base generation. Figure F-9 reppresents an example of the VM/CMS interactive conversation using step 11 with all user inputs underlined. The user should note, as illustrated in Figure F-8, that in addition to the procedure "GSMFDB", additional inputs are required to satisfy magnetic tape allocation.
Only when the response "TAPE 181 ATTACHED" is received, should the user enter "GSMFDB". This insures that a magnetic tape will be generated containing the GSMF data base.

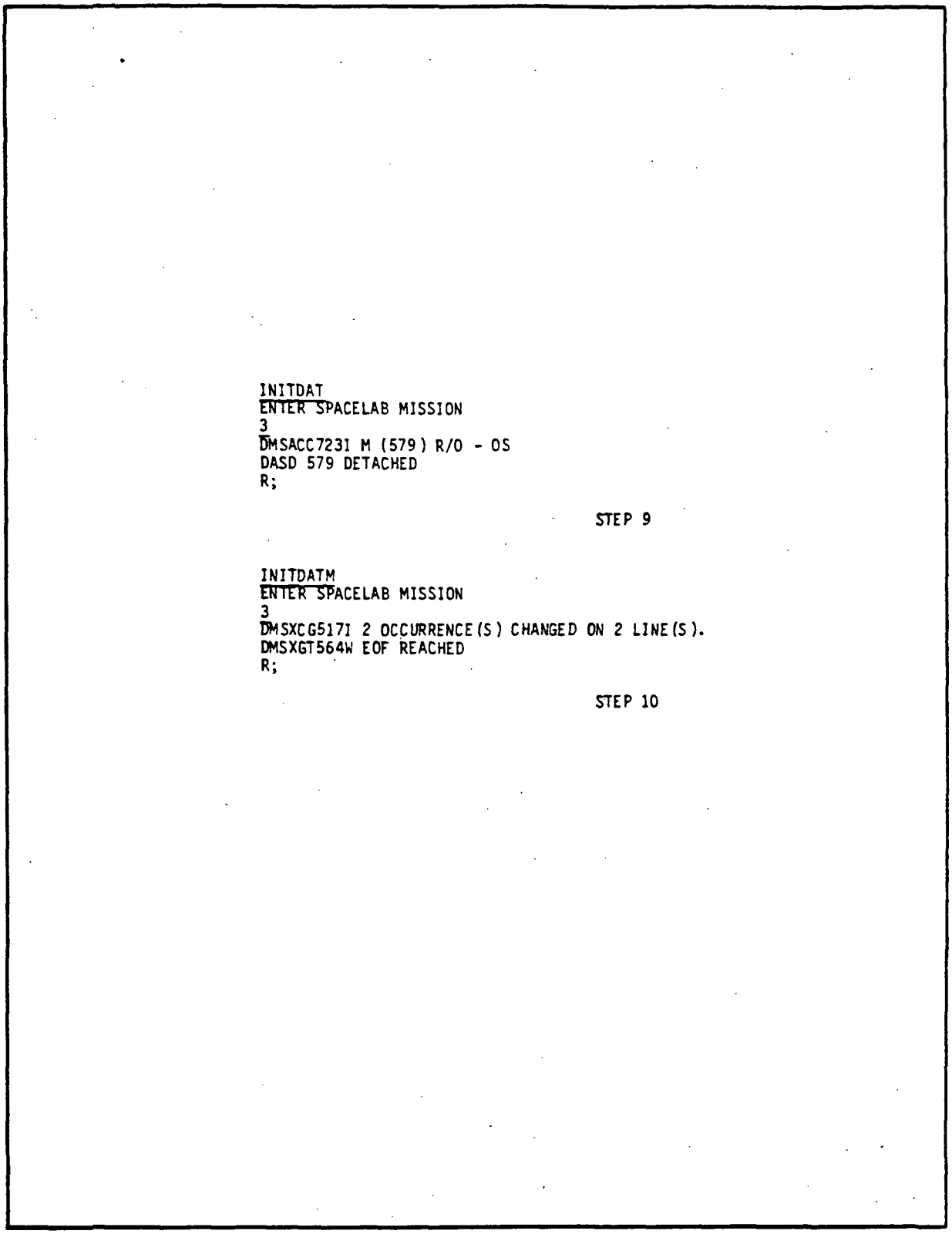


Figure F-8. SWID Initial DATA VM/CMS Procedures

```
MOUNT SCRTCH 181
MSG OP SCRTCH HAS VOL SER OF GSMFM*L** AND TITLE OF
     GSMF DATA BASE MIS M LEVL  (*M IS MISSION NUMBER, **L IS LEVEL)
TAPE 181 ATTACHED
GSMFDB
     THIS CMS EXEC WILL GENERATE THE GSMF DATA BASE TAPE.  THE FORMAT OF
     THIS TAPE IS AS FOLLOWS:
     FILE                    CONTENTS
      1                      ECOS OFFSET FILE
      2                      SCOS OFFSET FILE
      3                      SWID/SMID MEAS OFFSET FILE
      4                      STIMULI OFFSET FILE
      5                      HARDWARE OFFSET FILE
      6                      SWID TYPE FILE
      7                      SWID INITIAL DATA FILE
      8                      RUN DOCUMENTATION FILE
ENTER SPACELAB MISSION
3
DASD 507 DETACHED
DASD 507 DETACHED
DASD 508 DETACHED
DASD 507 DETACHED
DASD 507 DETACHED
DASD 508 DETACHED
DASD 507 DETACHED
DASD 508 DETACHED
DASD 507 DETACHED
STARTING GENERATION OF ECOS OFFSET FILE
GENERATION OF ECOS OFFSET FILE SUCCESSFUL

STARTING GENERATION OF SCOS OFFSET FILE
GENERATION OF SCOS OFFSET FILE SUCCESSFUL

STARTING GENERATION OF SWID MEAS OFFSET FILE
GENERATION OF SWID MEAS OFFSET FILE SUCCESSFUL

STATING GENERATION OF STIMULI OFFSET FILE
GENERATION OF STIMULI OFFSET FILE SUCCESSFUL

STARTING GENERATION OF HARDWARE OFFSET FILE
GENERATION OF HARDWARE OFFSET FILE SUCCESSFUL

STARTING GENERATION OF SWID TYPE FILE
GENERATION OF SWID TYPE FILE SUCCESSFUL

STARTING GENERATION OF SWID INITIAL DATA FILE
GENERATION OF SWID INITIAL DATA FILE SUCCESSFUL

STARTING GENERATION OF RUN DOCUMENTATION FILE
GENERATION OF RUN DOCUMENTATION FILE SUCCESSFUL

GSMF DATA BASE TAPE GENERATION IS COMPLETE
R;
                                        STEP 11
```

Figure F-9. GSMF Data Base Generation VM/CMS Procedure

APPENDIX G

MITRA SETUP MODE DATA

APPENDIX G

MITRA SETUP MODE DATA

1. SCOS/ECOS FETCH COMMANDS

In order to support GCOS monitoring of ECOS and SCOS telemetry data, a set of SCOS/ECOS fetch commands must be loaded with the MITRA for eventual download into the GCID. These fetch commands will be consistant with those utilized by the SCOS/ECOS simulation process in the Software Development Facilities (SDF1 and SDF2) for a given Spacelab mission.

The procedures used to create these fetch commands under the IBM 4381 VM/CMS system are defined in steps 12-14 of Figure F-1. The output from these steps will be a magnetic tape which must be loaded onto MITRA.

Figure G-1 represents an example of the IBM VM/CMS interactive conversation using steps 12-14 with all user inputs underlined. The MITRA load procedure is illustrated in Figure G-2.

2. DGNC DATA

In order to support DGNC data uplinks to SCOS and ECOS in the integrated mode, a tape containing DGNC data must be loaded onto the MITRA. Presently this data is being supplied by KSC, i.e., a tape is transmitted to Huntsville. This tape is then loaded onto the MITRA. The load procedure is illustrated in Figure G-3 where all user inputs are underlined.

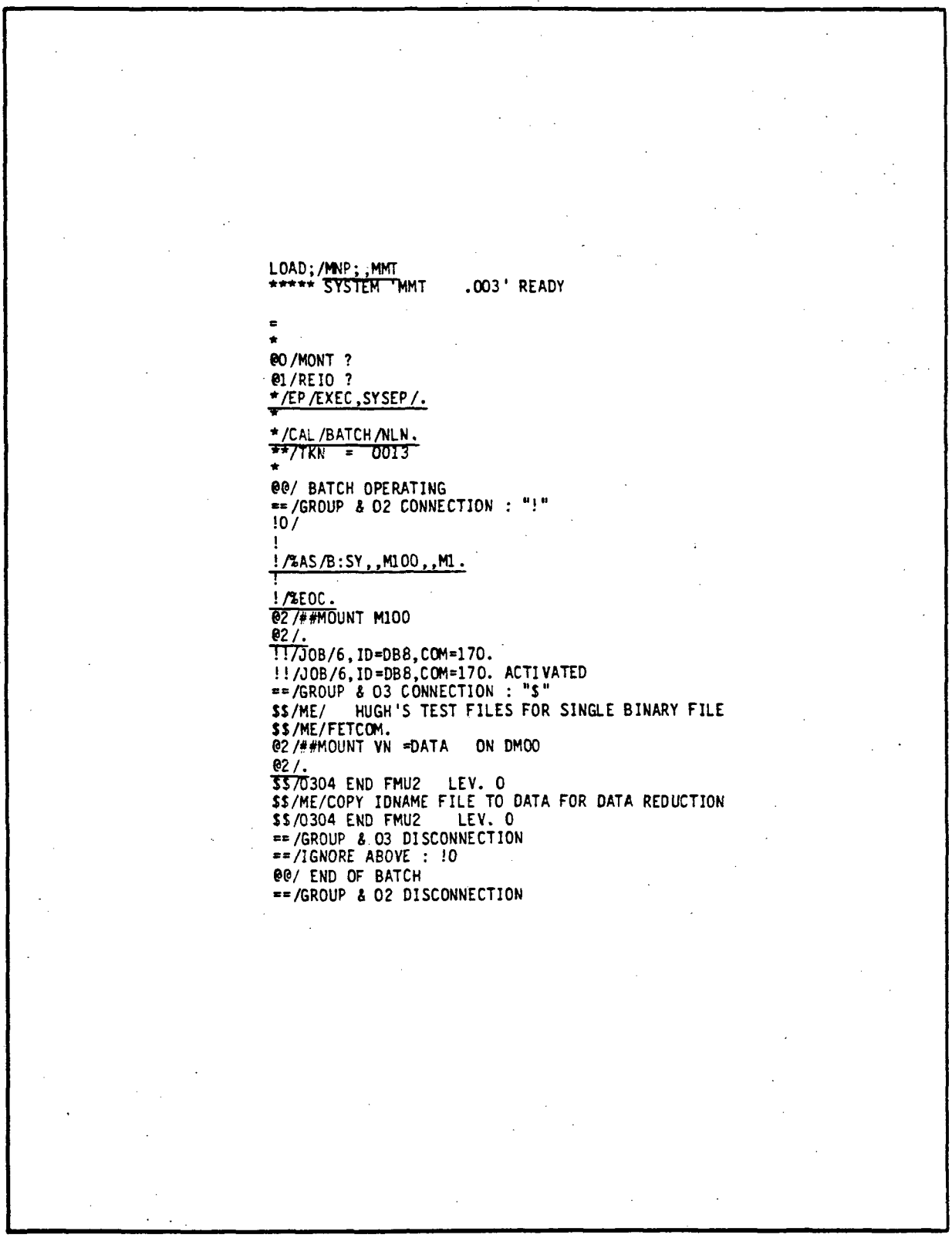
```
ESFETC
ENTER SPACELAB MISSION
3
DMSACC723I M (579) R/O  -  OS
DASD 579 DETACHED
DMSACC723I M (579) R/O  -  OS
DASD 579 DETACHED
DMSACC723I M (579) R/O  -  OS
DASD 579 DETACHED
DMSXCG517I 2 OCCURRENCE(S) CHANGED ON 2 LINE(S).
DMSXGT564W EOF REACHED
R;
                                        STEP 12

FETCOM
ENTER SPACELAB MISSION
3
DMSXCG517I 4 OCCURRENCE(S) CHANGED on 4 LINE(S).
R;
                                        STEP 13

SINBINTP
ENTER LIBRARY NAME OF MITRA JCL,MISSION,& CDT
     EXAMPLE:  JR01 MIS003 V1302
JR01 MIS003 V1302
EDIT:
EOF REACHED
//
TOF:
//            VB8=VB8XXX ,
//            VB8=MIS003 ,
//            VB8=MIS003,
//            VB9=VB9XXX
//            VB9=V1302
//REBLK.INPUT DD *
EOF REACHED
TOF IN.
     ENTER 'X' TO XEDIT
     JUST ENTER TO SKIP EDIT.
X
     PRESS ENTER TO SUBMIT
              "NO" TO CANCEL.

     XSYSGBIN EXEC COMPLETED.
R;
                                        STEP 14
```

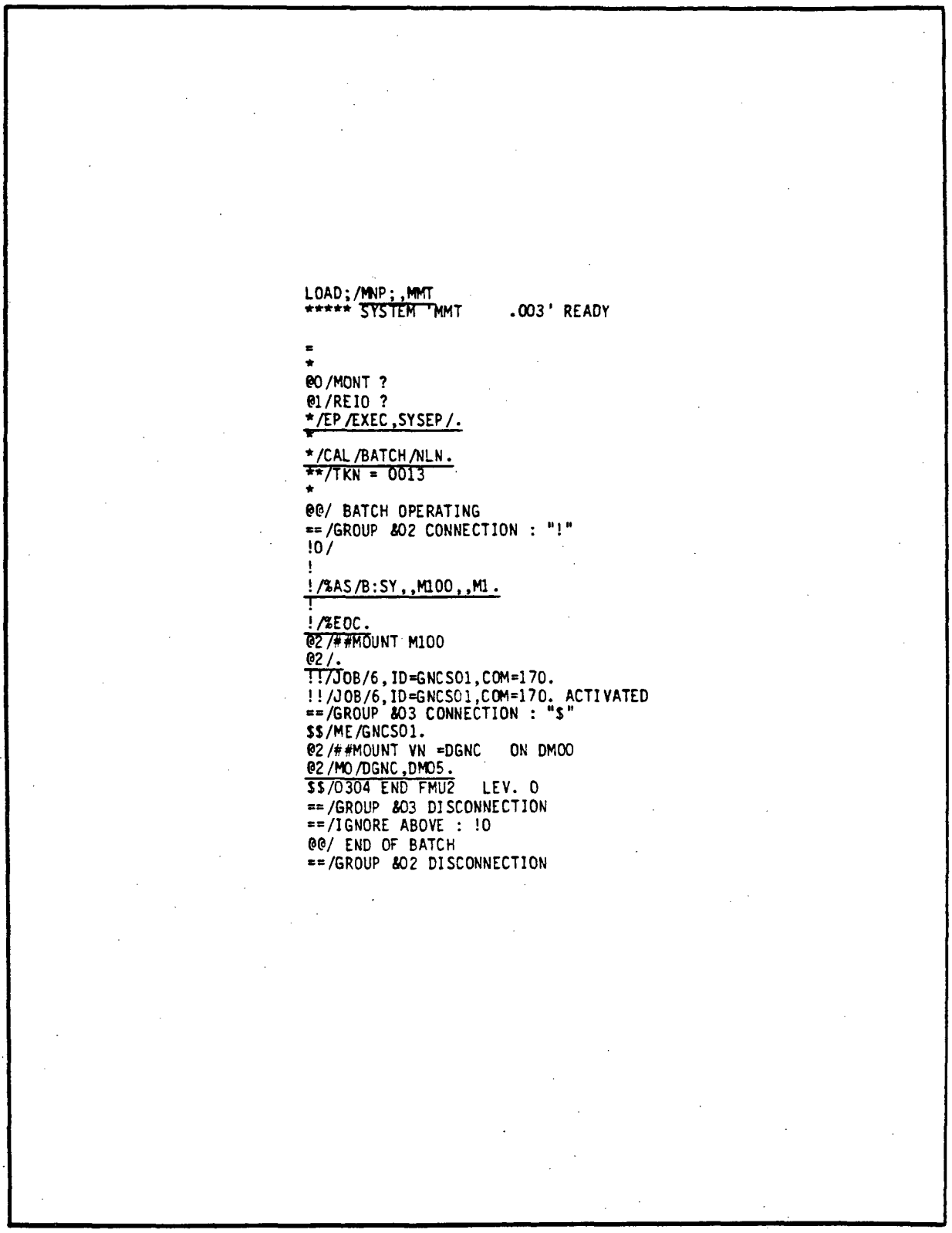
Figure G-1. ATE Fetch Command Generation VM/CMS Procedure



```
LOAD;/MNP;,MMT
***** SYSTEM 'MMT    .003' READY

=
*
@O/MONT ?
@1/REIO ?
*/EP/EXEC,SYSEP/.
*
*/CAL/BATCH/NLN.
**/TKN  =  0013
*
@@/ BATCH OPERATING
==/GROUP & 02 CONNECTION : "!"
!0/
!
!/%AS/B:SY,,M100,,M1.
!
!/%EOC.
@2/##MOUNT M100
@2/.
!!/JOB/6,ID=DB8,COM=170.
!!/JOB/6,ID=DB8,COM=170. ACTIVATED
==/GROUP & 03 CONNECTION : "$"
$$/ME/   HUGH'S TEST FILES FOR SINGLE BINARY FILE
$$/ME/FETCOM.
@2/##MOUNT VN =DATA   ON DM00
@2/.
$$/0304 END FMU2   LEV. 0
$$/ME/COPY IDNAME FILE TO DATA FOR DATA REDUCTION
$$/0304 END FMU2    LEV. 0
==/GROUP & 03 DISCONNECTION
==/IGNORE ABOVE : !0
@@/ END OF BATCH
==/GROUP & 02 DISCONNECTION
```

Figure G-2. MITRA Fetch Command Tape Load Procedure



```
LOAD;/MNP;,MMT
***** SYSTEM 'MMT      .003' READY

=
*
@0/MONT ?
@1/REIO ?
*/EP/EXEC,SYSEP/.
*
*/CAL/BATCH/NLN.
**/TKN = 0013
*
@@/ BATCH OPERATING
==/GROUP &02 CONNECTION : "!"
!0/
!
!/%AS/B:SY,,M100,,M1.
!
!/%EOC.
@2/##MOUNT M100
@2/.
!!/JOB/6,ID=GNCS01,COM=170.
!!/JOB/6,ID=GNCS01,COM=170. ACTIVATED
==/GROUP &03 CONNECTION : "$"
$$/ME/GNCS01.
@2/##MOUNT VN =DGNC    ON DM00
@2/M0/DGNC,DM05.
$$/0304 END FMU2   LEV. 0
==/GROUP &03 DISCONNECTION
==/IGNORE ABOVE : !0
@@/ END OF BATCH
==/GROUP &02 DISCONNECTION
```

Figure G-3. DGNC MITRA Tape Load Procedure

APPENDIX H

MITRA OPERATING PROCEDURES

APPENDIX H

MITRA OPERATING PROCEDURES

Appendix H details the operating and execution instructions for loading and starting the MITRA, starting GCOS, executing the GOAL LINK 3101 program, stopping LINK, MITRA control of the MET, and executing the ITTS test software.

1. LOAD AND START THE MITRA

   1) Verify the following MITRA panel switch settings:

      a) Key to Mode 2 Manual

      b) Mode to RD

      c) H/AD to Off

      d) MPC to SEL

      e) Switches 13, 18, 19 UP; others down

      f) Display to IND

   2) Load the GCOS load deck and press RESET on the card reader.

   3) On the MITRA panel, press INI and RUN.

   4) On the MITRA teletype (A24).

      Enter/MNP;MMTV1.
      */CR.

      On the MITRA panel change display switch to DATA

   5) Upon completion of the card deck, terminal A14 will display "1S A WARM START REQUIRED?".

   6) To disable the TLC 80 millisecond time out message, enter the following patch on the teletype:

      %/MO/&3252.&0:C711
      %/MO/&325F.&A:&C718

2. START GCOS

   1) Start GCOS by entering (from terminal A14):

      BGIN COLD
      Y

2) During cold start, the following undefined hardware addresses will be displayed on terminal A17:

1840,1C40,2840,2C40,3840,3C40,4840,4C40

These stimuli are either undefined in the SL3 data base or their behavior is unknown.

3) Upon completion, terminal A14 will display 'STARTING CHECKOUT'.

3. RUN LINK CONFIGURATION 3101

1) Turn on the ECOS and SCOS links by entering the following from terminal A14:

```
P HA
CHAT TLCS ON
CHAT TLCE ON
CHAT TLMS ON
CHAT TLME ON
```

2) From terminal A14, load a monitor table:

```
P SG
LMOT SMT
```

Use terminal A15 to view the performance of LINK by

```
P SA
```

3) From terminal A14, begin LINK by entering:

```
PERF LINK
P LINK
RESU
REPY 2 (Activate Spacelab Only)
REPY X3101
```

Observe the programs that will be executed and enter:

```
REPY 1
```

4) Use terminal A13 to display the LINK program names by entering:

```
P LINK
```

5) If any programs require operator action the program name and "Q" will appear at the top of the page. To view the message, enter "P" followed by the program name (example, P ABAN).

6) If any errors are encountered, set the measurement manually from the Perkin-Elmer Value Read/Write function. Then enter REPY 1 (RESTART) from terminal A14.

7) If any Manual intervention is required, the program name and "S" will appear at the top of the page. To view the message, enter "P" followed by the program name (example P DACF). After performing the manual operation, enter RESU.

8) NOTE: During execution of LINK 3101 the following undefined hardware address message will appear: 19C5. The behavior for this stimulus is unknown.

9) The last LINK program EXCM, loads the experiment computer via the MMU. While it is executing, select DMON ELAV from terminal A14 and observe the IPL block count (7899) updated by the model:

10) Upon completion, observe the message "SL ACTIVATION PROGRAM COMPLETED" on terminal A13.

4. EXECUTE THE DOWN PROGRAM

1) From terminal A14, enter:

```
P SG
PERF DOWN
P DOWN
RESU
```

2) From terminal A15, select P SA to observe execution of DOWN.

3) Upon completion, observe "SL DEACTIVATION COMPLETE" on terminal A15.

4) Stop the MITRA by pressing the HALT button.

5. MITRA CONTROL OF THE MET

5.1 GMT and MET Transfer to the MIT

1) At the A15 enter the command GMT ON.

2) GMT will be displayed on A13. It should match the time displayed on the GCID time code generator and GSMF terminals.

3) Enter the command MET ON at A14.

4) MET will be displayed on A13. The value should be constant because MET is initially stopped.

5.2 MITRA Control of MET

1) Enter the following commands at A14:

```
CD 7101
CD 7223
CD 7310
CD 7458
CD 7545
```

2) After entry of the above commands the MET displayed on A13 should be:

123 10 58 45

but should remain stopped.

3) Enter the command CD 77F0 at the A15.

4) The MET displayed on A13 should begin incrementing.

5) Examine GMT and MET on the GSMF Performance Analysis terminal.

6. EXECUTING THE ITTS TEST PROGRAMS

6.1 SCCD Commands

OVERVIEW: This exercise sends SCCD commands to light or extinguish lamps on the Operator Console.

6.1.1 Verify Emergency Panel

1) Select to view emergency panel, function key 15.

2) From ITTS, issue Fire Right and Fire A (CD F040).

3) Observe that Fire Right and Fire A lamps are lit.

4) From ITTS, reset Fire Right A (CD F000).

5) From ITTS, issue Fire Cabin B (CD F102).

6) Observe that Fire Cabin and Fire B lamps are lit.

7) From ITTS, reset Fire Cabin B (CD F100).

6.1.2 Verify Caution and Warning Panel

1) Select to view Caution and Warning, function key 14.

2) From ITTS, set warning lamps 1-8 (CD FBFF).

3) Observe that warning lamps 1-8 are lit.

4) From ITTS, reset warning lamps 1-8 (CD FB00).

5) Observe that warning lamps 1-8 are extinguished.

6) From ITTS, set caution lamps 1-8 (CD F8FF).

7) Observe that caution lamps 1-8 are lit.

8) From ITTS, reset caution lamps 1-8 (CD F800).

6.1.3 Verify Exception/Enable Panel

1) Select to view Exception/Enable Panel, function key 13.

2) From ITTS, set enable 1-8 (CD F2FF).

3) Observe that enable lamps 1-8 are lit.

4) From ITTS, reset enable 1-8 (CD F200).

5) Observe that enable lamps 1-8 are extinguished.

6) From ITTS, set exception 1-4 (CD F30F).

7) Observe that exception lamps 1-4 are lit.

8) From ITTS, reset exception 1-4 (CD F300).

9) Observe that exception lamps 1-4 are extinguished.

6.2 MSE Commands

OVERVIEW: This exercise sends MSE commands to affect Operator Console lamps.

6.2.1 Control and Display Panel 1

1) Select C&D1 via shifted function 13.

2) From ITTS, issue APS-A ON (DO 1.1.03 ON).

3) Observe that lamp 1 is lit.

4) From ITTS, issue APS OFF (DO 1.1.07 OFF).

5) Observe that lamp 1 is extinguished.

6.2.2 Control and Display Panel 2

1) Select C&D2 via shifted function 14.

2) From ITTS, issue AVFAN1 Low (DO 1.3.36 ON).

3) Observe that lamp 1 is lit.

4) From ITTS, issue AVFAN12 OFF (DO 1.3.38 OFF).

5) Observe that lamp 1 is extinguished.

6.2.3 Control and Display Panel 3

1) Select C&D3 via shifted function 15.

2) From ITTS, issue RAU A ON (DO. 1.5.30 ON).

3) Observe that lamp 1 is lit.

4) From ITTS, issue SS RAU's OFF (DO 1.5.29 OFF).

5) Observe that lamp 1 is extinguished.

6.2.4 EGSE Control and Display Panel

1) Select EGSE Control and Display via shifted F16.

2) From ITTS, issue APS-A ON (DO 1.1.03 ON).

3) Observe that lamp 1 is lit.

4) From ITTS, issue APS-A OFF (DO 1.1.07 OFF).

5) Observe that lamp 1 is extinguished.

6.3 TLC Commands

This procedure transmits a TLC message and verifies its effect on measurements.

The following stimuli affect SCOS measurements through the "DOPAN" behavior function:

Stimuli	Measurement	Effect
1613	568	1
1639	625,626	1
1707	840	1
1614	568	0
1640	625,626	0
1708	840	0

1) At the Data Display Terminal, select file DOPANTST

2) At the ITTS terminal, build the TLC commands

-TLCS A01 FF00 3AFF 0443 064D 0667 06AB

-TLCS B01 FF00 3AFF 0443 064E 0668 06AC

3) At the ITTS terminal, enter

TLCS SEND A

4) At the Data Display Terminal, verify measurements 568,625,626, and 840 are set to 1

5) At the ITTS terminal, enter

TLCS SEND B

6) At the Data Display terminal, verify measurements 568,625,626, and 840 are set to 0.

6.4 Fetch Memory

This procedure provides a demonstration of fetch memory load and verify.

6.4.1 Fetch Memory Load

1) To prepare the MSE for the data transfer, enter the command MSE MODE AS at A14.

2) VDU1 should not display any errors.

3) Enter the command LP ER to disable the line printer.

4) Enter the command FM LOAD to start the load process.

5) Each MSE command executed will be displayed on A13.

6) To speed up the transfer, enter A13 ER to disable the command display.

7) Upon completion, the message FETCH MEMORY LOADED will be displayed on A13.

6.4.2 Fetch Memory Verify

1) To start the verify process enter FM CHECK at A14.

2) The MITRA will then read the fetch memory via the MSE and compare the data with a load file. Errors will be displayed by the line printer.

3) The process takes about 2 minutes. At completion, A13 will indicate the fetch memory has been checked.