

## OVERVIEW

### 1. LIST DATA IN HEX

```
<-INPUT DD SYSUT1 OR 'INDD=..'
->OUTPUT DD SYSPRINT
```

TWO TYPES OF HEX FORMATS-EITHER LIKE A DUMP OR A CHARACTER ON TOP OF HEX.

### 2. COPY A DATASET

```
<-INPUT DD SYSUT1 OR 'INDD=..'
->OUTPUT DD SYSUT2 OR 'OUTDD=..'
```

THIS COPIES LIKE IEBGENER. THE OUTPUT IS A COPY OF THE INPUT. THE RECORDS MAY BE REARRANGED BY YOU TO A NEW FORMAT.

\*\* (1) AND (2) MAY BE COMBINED. THE OUTPUT PRODUCED IS CONTROLLED BY VARIOUS SELECTION COMMANDS.

DEFAULT USES THE DSORG,RECFM,BLKSIZE DEFINED IN THE OUTPUT DATASET IF CODED OR THE INFORMATION IS COPIED FROM THE INPUT DATASET IF OMITTED. THE LRECL AND BLKSIZE OF 'INDD' AND 'OUTDD' MAY BE DIFFERENT.

## CLIST AND BATCH SYNTAX

THE PROGRAM RUNS IN BOTH FOREGROUND AND BACKGROUND. REFER END OF THIS DOCUMENTATION FOR EXECUTION IN BACKGROUND. REFER TO 'EXAMPLES' FOR EXAMPLES.

```
TSO USE : %SPGEN DSNAME |VOL(..) PRINT(..) SYSUT2(..)|
```

```
BG USE : //SUPERGEN EXEC SPGEN,OUT=T,GENPARM='OPTIONAL CMDS'
//SYSUT1 ---- INPUT DD AS REQUIRED.
//SYSUT2 ---- OUTPUT DD AS REQUIRED.
//SYSIN DD *- OPTIONAL INPUT COMMANDS.
```

## TSO CLIST CMD SYNTAX

DSNAME - NAME OF DATASET.

VOL(..) - VOLUME SERIAL NUMBER. OVERRIDES SYSTEM CATALOG OR SPECIFIES THE DSNAME LOCATION IF NOT CATALOGED. THE DATASET MUST EXIST ON SPECIFIED VOLUME.

PRINT(..) - AFFECTS THE 'SYSPRINT' ALLOCATION STATEMENT. DEFAULT IS TO THE TUBE. SPECIFY 'PRINT(SYSOUT)' TO DIRECT THE SYSPRINT TO HARDCPY INSTEAD OF THE TUBE. YOU MUST

RE-LOGON OR LOGOFF TO OBTAIN PRINTOUT. 'SYSIN' IS STILL TO THE TUBE.

\*\* PLEASE NOTE THAT NO MESSAGES ARE PRINTED TO THE TUBE. IT IS NOW 'SYSOUT' SO BEWARE.

SYSUT2(.) - AFFECTS THE 'SYSUT2' ALLOCATION STATEMENT. DEFAULT IS TO THE TUBE. SPECIFY 'SYSUT2(SYSOUT)' TO DIRECT SYSUT2 TO HARDCPY INSTEAD OF THE TUBE. YOU MUST RE-LOGON OR LOGOFF TO OBTAIN PRINTOUT.

#### SPECIAL FEATURE

THE ATTENTION INTERRUPT (PA2) FEATURE IS SUPPORTED IN THIS CLIST AND IS NORMALLY USED WHILE DUMPING A DATASET. TO USE:

1. PRESS THE PA2 KEY.
2. IF THE SCREEN IS FULL (END OF PAGE) PRESS THE 'ENTER' KEY TO RECEIVE THE MESSAGE IN RESPONSE TO THE PRESSING OF THE PA2 KEY.
3. KEY IN THE INDICATED RESPONSE AND PRESS 'ENTER'.
4. IF 'Y' WAS ENTERED THE EXECUTING PROGRAM IS RE-EXECUTED, RE-USING YOUR CURRENT OPTIONS AND ALLOCATIONS.
5. 'N' IS TREATED THE SAME AS '/' AND ALL ALLOCATIONS ARE FREED.

#### SYNTAX

ALL COMMANDS APPLY TO BOTH 'FG' AND 'BG' EXECUTION.

MOST COMMANDS MAY BE ABBREVIATED TO THE MINIMUM REQUIRED TO DISTINGUISH ONE FROM THE OTHER. THE ABBREVIATED FORM IS LISTED FIRST FOLLOWED BY A COMPLETE SPELLING OF THE COMMAND. THE '=', '(', AND ')' MAY BE REPLACED WITH BLANKS WHERE THEY APPEAR.

'SPECIAL' COMMANDS MUST BE ENTERED IN THEIR ENTIRETY. (E.G., 'EXCP=Y'. REFER TO 'SPECIAL' AT END OF OPERANDS)

OPERANDS (SUMMARY) SUBCOMMANDS FOLLOW:

SUBCOMMANDS FOLLOW:

MULTIPLE SUBCOMMANDS MAY BE ENTERED ON ONE LINE, SEPARATED BY A DELIMITER (A BLANK OR COMMA). THE SUBCOMMAND 'GO' STARTS THE

EXECUTION OF THE PROGRAM ACCORDING TO THE COMMAND OPTIONS CHOSEN.  
IF NO SUBCOMMANDS ARE ENTERED AND 'GO' IS ENTERED, YOU GET A  
HEXDUMP OF THE DATA ON TSO AND A COPY OF THE DATA IN 'BG'. AT THE  
COMPLETION OF THE 'TASK', CONTROL RETURNS TO 'SYSIN FOR MORE INPUT.  
A '/' ENTERED WITHOUT ANY PRIOR 'GO' IS SIMILAR, EXCEPT THAT THE  
PROGRAM ENDS WHEN THE TASK IS COMPLETED.

\*\*\*\*\*  
\* HERE IS A SUMMARY OF ALL SUBCOMMANDS \*  
\*\*\*\*\*

DUMP=N/Y,HC,ALL	GET A HEXDUMP OF INPUT
INDD=SYSUT1	THE INPUT DDNAME
OUTDD=SYSUT2	THE OUTPUT DDNAME
FIND(S,E) CC SKIP/STOP X/C' '	TO FIND DATA
AND	COMBINE ALL 'FIND'S
OR	ANY FIND WILL DO.
BLKSIZE=N	OVERRIDE THE BLKSIZE (RECFM=U)
COUNT=N (DEFAULT=ALL)	# OF OUTPUT RECORDS TO PROCESS
SKIP=N (DEFAULT=NONE)	# OF INPUT RECORDS TO SKIP.
PUNCH(S,E) X/C' '	CREATE AN OUTPUT RECORD.
MOVE(S,E,I) OR (S,E) X/C' '	SPECIFY WHICH COLUMNS TO COPY
RESET	RESET EVERYTHING BACK TO NOTHING
KEEP=N/Y	SAVE CMDS FOR REUSE.
GO	PERFORM FUNCTIONS
END	END EXECUTION
/* (PLEASE READ BELOW)	END EXECUTION
*** THE FOLLOWING ARE SPECIAL	REFER TO SPECIAL PLEASE ***
DSN=	DSNAME TO PROCESS
EXCP=N/Y	USE EXCP
VTOC=N/Y	PROCESS THE VTOC
CCHHR=CCCCHHHR	
CYL=C	
HEAD=H	
REC=R	
TTR=TTR	
EOF=N/Y	
DEB=N/Y	

OPERANDS (INDIVIDUALLY EXPLAINED)

\* COMMENT, ALL INFO AFTER '\*' IS TREATED AS COMMENT.

D DUMP=N/Y LIST OUTPUT IN 'DUMP' FORMAT.(DDNAME=SYSPRINT)  
DUMP=Y IS THE DEFAULT IF EXECUTED IN 'FG' TSO.  
DUMP=HC LIST OUTPUT IN HEX FORMAT WITH TRANSLATION ON TOP  
OF THE HEX PRINTOUT OF EACH POSITION.  
DUMP=ALL SPECIAL FORMAT FOR 'EXCP=Y'.READS ALL EXTENTS.

I INDD=SYSUT1 INPUT DDNAME. EACH TIME 'INDD' IS CODED, THE OLD  
'INDD' IS CLOSED AND THE NEW 'INDD' IS OPENED.  
NORMALLY THE INPUT IS LEFT POSITIONED AT THE LAST  
RECORD READ + 1, UNLESS AN END-OF-FILE IS READ,  
THEN THE FILE IS CLOSED AND PROCESSING STARTS AT  
THE BEGINNING AGAIN.

O OUTDD=SYSUT2 OUTPUT DDNAME. DCB INFORMATION IS COPIED FROM THE  
'INDD' DATASET IF NOT PRESENT IN THE JCL. EACH TIME  
'OUTDD' IS SPECIFIED' THE OLD 'OUTDD' IS CLOSED AND  
THE NEW 'OUTDD' IS OPENED. IF THE DDNAME IS MISSING  
OR THE FILE CANNOT BE SUCCESSFULLY OPENED, A MESSAGE  
IS ISSUED AND PROCESSING CONTINUES. (E.G. DUMP THE  
DATA IN HEX)

'OUTDD' OR SYSUT2 IS NOT REQUIRED WHEN LISTING WITH  
DUMP=Y.

\*\* THIS DATASET REPRESENTS A 'COPY' OF THE 'INDD'  
DATA AS CONTROLLED BY THE OTHER COMMANDS.

=N A SPECIAL FORM IMPLIES NO OUTPUT TO GO TO 'OUTDD'.  
'N' IS THE DEFAULT WHEN EXECUTED IN 'FG' TSO.

F FIND X'----' X=HEX , 2 CHARACTERS PER BYTE.  
C'----' C=CHARACTER.  
'/' IS ALSO A VALID DELIMITER.  
DEFINES THE SEARCH STRING.  
SEARCHES THE WHOLE RECORD FOR THE 'STRING'.

F FIND(N)= SAME AS ABOVE-- 'C' OR 'X'.  
N=COLUMN NUMBER TO SEARCH.  
COUNTING STARTS AT 1. ADD +4 TO N FOR  
RECFM=V,VB. VBS FOR THE RDW.

F FIND(S,E)= SAME AS ABOVE-- 'C' OR 'X'.  
S=STARTING COLUMN  
E=ENDING COLUMN (DEFAULT IS 'S'.)

```
F FIND(S,E) CC SKIP C'----'  
                      X'----'  
F FIND(S,E) CC STOP C'----'  
                      X'----'
```

S=OPTIONAL-STARTING COLUMN (DEFAULT IS COL 1.)  
E=OPTIONAL-ENDING COLUMN (DEFAULT IS 'S',  
IF 'S,E' IS NOT CODED, THE WHOLE RECORD IS  
SEARCHED.  
CC=OPTIONAL-TYPE OF LOGICAL COMPARISON.

COMPLETE SPELLING FOR 'CC', 'STOP' AND 'SKIP'.

EQ= STRING EQUAL (THIS IS DEFAULT.)  
NE= NOT EQUAL.  
LT= LESS THAN.  
GT= GREATER THAN.  
LE= LESS THAN OR EQUAL TO.  
GE= GREATER THAN OR EQUAL TO.  
SKIP=OPTIONAL THIS 'FIND' WILL START PROCESSING  
THE INPUT WHEN A MATCH OCCURS. THIS IS THE  
FIRST RECORD TO PROCESS.  
STOP=OPTIONAL THIS 'FIND' WILL STOP PROCESSING  
THE INPUT WHEN A MATCH OCCURS. THIS IS THE  
LAST RECORD TO PROCESS.

AS MANY 'FINDS' AS DESIRED MAY BE ENTERED. EACH 'HIT' WILL BE  
DUMPED TO SYSPRINT OR COPIED TO 'OUTDD' WHEN 'GO' OR '/' IS  
ENTERED. REFER TO 'AND' AND 'OR' FOR COMBINING MULTIPLE FINDS.

USE 'SKIP' TO START PROCESSING THE INPUT DATASET AT THE PLACE WHERE  
THE DATA IS EQUAL TO ANY/ALL OF THE 'FIND..SKIP' SEARCH STRINGS.  
'SKIP=N' MAY BE COMBINED WITH 'SKIP=FIND' (I.E., YOU MAY SKIP 'N'  
RECORDS BEFORE YOU START LOOKING FOR STRING MATCHES).

AND/OR APPLY TO ALL 'FIND'S

AND IF CODED, ALL OF THE 'FIND' STRINGS MUST COMPARE  
BEFORE A RECORD IS SELECTED.

OR DEFAULT VALUE. IF ANY ONE OF THE 'FIND' STRINGS  
COMPARE THE RECORD IS SELECTED.

B BLKSIZE=N N=MAX BLKSIZE TO READ. IMPLIES RECFM=U. READS  
'1' PHYSICAL BLOCK ' L.E.' N BYTES. ALSO USED  
WITH 'EXCP=Y'.

C COUNT=N N=NUMBER OF OUTPUT RECORDS TO PROCESS. THE COUNT  
IS INCREMENTED BY 1 EACH TIME A RECORD IS

SELECTED. DEFAULT IS ALL RECORDS.

S SKIP=N N=NUMBER OF RECORDS TO SKIP. DEFAULT=NONE.

P PUNCH(S,E) X'---' X=HEX, 2 CHARACTERS PER BYTE.  
C'---' C=CHARACTER.  
S=OPTIONAL-STARTING COLUMN (DEFAULT IS COLUMN 1.)  
E=OPTIONAL-ENDING COLUMN (DEFAULT IS 'S'),  
IF 'S,E' IS NOT CODED, THE WHOLE RECORD IS FILLED  
WITH THE FIRST BYTE OF THE STRING CODED.  
(PUNCH C'\*' WILL FILL AN OUTPUT LINE WITH '\*'  
DO NOT CODE 'E' UNLESS YOU WANT THE FIRST CHARACTER  
PROPAGATED. FOR EXAMPLE: CODING PUNCH(1,16) C'\*DATA  
WILL GENERATE '\*\*\*\*\*DATA'.

NOTE: THIS COMMAND REQUIRES THE DEFAULT 'OUTDD=' OR YOUR OWN.

ONE (1) OUTPUT RECORD IS CREATED FROM ALL THE 'PUNCH' STATEMENTS  
UP TO THE 'GO' COMMAND. IT IS YOUR RESPONSIBILITY TO SPECIFY EACH  
OUTPUT POSITION. (USE PUNCH WITHOUT 'S,E' AS THE FIRST STATEMENT  
TO FILL THE LINE WITH A DEFAULT CHARACTER.) IF NO INPUT RECORDS  
ARE TO BE READ AND YOU JUST WANT TO PUNCH A RECORD, CODE:

YOUR 'PUNCH' COMMANDS, COUNT=0,GO

POSITIONS ARE FILLED IN THE ORDER OF THE 'PUNCH' STATEMENTS. YOU  
MAY USE 'KEEP=Y' TO KEEP THE CMDS ENTERED, THEN EACH 'GO' WILL  
CREATE ONE NEW OUTPUT RECORD BEFORE IT SELECTS FROM THE INPUT AND  
STARTS COPYING IT TO THE 'OUTDD'. REFER TO 'EX10' FOR AN  
EXAMPLE.

M MOVE(S,E,I)

S=REQD-OUTPUT STARTING COLUMN.  
E=REQD-OUTPUT ENDING COLUMN.  
I=REQD-INPUT STARTING COLUMN.  
THE (S,E) POSITIONS CODED ARE FILLED  
WITH THE DATA FROM INPUT LOCATION 'I'.  
NO MOVE IS DONE IF THE DATA DOESN'T FIT.  
THE OUTPUT SHOULD BE FILLED WITH A DEFAULT FILL  
CHARACTER (SEE THE NEXT 'M' FORMAT) THEN FOLLOWED BY  
AS MANY MOVE'S AS REQUIRED TO CREATE A NEW OUTPUT  
RECORD.  
M(1,80) C' ',MOVE(1,72,9) WILL COPY  
COLUMNS 8,80 OF THE INPUT TO COLUMNS 1-72 IN THE  
OUTPUT WITH BLANKS IN COLUMNS 73,80.

MOVE(S,E) X'---' X=HEX, 2 CHARACTERS PER BYTE.  
C'---' C=CHARACTER.  
S=REQD-OUTPUT STARTING COLUMN.  
E=OPTIONAL-OUTPUT ENDING COLUMN (DEFAULT=S).  
THE (S,E) POSITIONS CODED ARE FILLED  
WITH THE STRING CODED. IF THE STRING IS SHORTER

THAN THE OUTPUT LENGTH, THE FIRST BYTE IS PROPAGATED  
TO FILL OUT THE REMAINDER.

E.G. M(1,80) C'\*' WILL FILL AN OUTPUT LINE WITH '\*'  
FOR EXAMPLE: CODING MOVE(1,16) C'\*DATA'  
WILL GENERATE '\*\*\*\*\*DATA'.

NOTE: THIS COMMAND REQUIRES THE DEFAULT 'OUTDD=' OR YOUR OWN.

IT IS YOUR RESPONSIBILITY TO SPECIFY EACH OUTPUT POSITION.  
POSITIONS ARE FILLED IN THE ORDER OF THE 'MOVE' STATEMENTS. YOU  
MAY USE 'KEEP=Y' TO KEEP THE CMDS ENTERED, THEN EACH 'GO' WILL  
CREATE OUTPUT RECORDS FROM THE INPUT ACCORDING TO 'MOVE' S.  
REFER TO 'EX11' FOR AN EXAMPLE.

THIS COMMAND MAY BE COMBINED WITH ANY OTHER COMMAND TO LIMIT THE  
NUMBER AND TYPE OF DATA.

R RESET                RESET EVERYTHING BACK TO WHAT IT WAS WHEN PROGRAM  
                      'INDD' AND 'OUTDD' NAMES REMAIN AS MODIFIED. THE  
                      INPUT FILE IS ALSO CLOSED AND REPOSITIONED AT THE  
                      BEGINNING OF THE FILE. THE OUTPUT IS NOT CLOSED.

NOTE: 'DUMP,INDD,OUTDD,DSN' REMAIN IF CODED.

K KEEP=N/Y            DO NOT RESET SELECTION CMDS ENTERED AT END OF TASK.  
                      IF YOU CODE 'KEEP=Y' YOU CAN REPEAT THE LAST SET OF  
                      COMMANDS WITHOUT HAVING TO KEY THEM IN AGAIN. FOR  
                      EXAMPLE IF YOU CODE : FIND X'02',COUNT=1 THEN EACH  
                      TIME YOU KEY IN 'GO' IT WOULD TRY TO FIND 1 RECORD  
                      AT A TIME THAT HAD A X'02' ANYWHERE IN THE RECORD.

NOTE: THIS DOES NOT APPLY TO 'SKIP=N'. IT RESETS TO '0'.

G GO                   GO AND PERFORM THE FUNCTIONS DEFINED BY PRIOR  
                      CONTROL PARAMETERS.

E END                  END EXECUTION.

/\*                     SIMILAR TO END BUT WILL PROCESS THE INPUT DATASET  
                      ACCORDING TO THE DEFAULT VALUES 'FG'=DUMP,'BG'=COPY.  
                      IF NO PREVIOUS 'GO' WAS ENTERED AND END THE PROGRAM  
                      IF A 'GO' WAS PREVIOUSLY ENTERED.

SPECIAL                THE FOLLOWING CMDS MUST BE ENTERED AS SHOWN.  
                      PREVIOUS COMMANDS SUCH AS 'DUMP,OUTDD,COUNT,FIND'  
                      STILL APPLY, AND MAY BE USED TO LIMIT THE 'READS'.

DSN=DATA.SET.NAME

NAME OF DATASET YOU WANT TO PROCESS. DATASET IS ON  
VOL POINTED TO BY 'INDD'. DEFAULT IS THE JCL DSN.

EXCP=N/Y            USE EXCP TO READ THE DATA, MAY BE FOLLOWED BY A  
                     BLKSIZE=N, WHERE 'N' IS THE MAXIMUM BLKSIZE TO READ.  
                     THE BLKSIZE MAY BE LESS THAN THE ACTUAL SIZE, AND  
                     COULD SHORTEN THE OUTPUT IF ITS JUST LOOKING  
                     'AROUND'.

                     WILL READ ANY NONVSAM DATASET.  
                     CCHHR,KEYLEN,BLKSIZE,TTR AND EOF ARE LISTED.  
                     UNLESS A CCHHR,CYL,HEAD,REC, OR TTR IS SPECIFIED,  
                     READING STARTS AT THE BEGINNING OF THE DATASET AND  
                     STOPS AT THE FIRST 'EOF'.

                     IF 'DUMP=ALL' IS ALSO ENTERED, THE WHOLE EXTENT IS  
                     READ AND ALL 'EOFS' OR I/O ERRORS ARE LISTED. IF  
                     EOF=Y IS ALSO ENTERED, ONLY EOF(S) ARE LISTED.

VTOC=N/Y            PROCESS THE VTOC OF THE PACK POINTED TO BY SYSUT1  
                     (INDD) USING EXCP. THE DATA IS PROCESSED 'AS IS'.

CCHHR=CCCCHHHHRR  
                     C,H,R ARE HEXADECIMAL DIGITS REPRESENTING THE DISK  
                     ADDRESS TO START AT. MAY BE USED WITH VTOC AND EXCP.

#### OPTIONAL CMDS

THE FOLLOWING CMDS FOLLOW EXCP OR VTOC AND ARE OPTIONAL

CYL=N                N=HEX NUMBER, LEADING ZEROES NOT REQD.  
                     BEGIN DISK CYL.

HEAD=N               N=HEX NUMBER,LEADING ZEROES NOT REQD.  
                     BEGIN HEAD.

REC=N                N=HEX NUMBER,LEADING ZEROES NOT REQD.  
                     BEGIN RECORD.

TTR=NRR             N=HEX NUMBER,LEADING ZEROES NOT REQD.  
                     BEGIN RELATIVE TRACK. RR=2 DIGIT HEX NUMBER,  
                     GIVING THE BEGINNING RECORD.

EOF=N/Y              USED WITH EXCP=Y TO LIST ONLY 'EOF' RECORDS.  
                     NO DATA RECORDS ARE LISTED.

DEB=N/Y              BYPASS DEBCHK.

EXAMPLES            SOME EXAMPLES OF SUPERGEN IN 'FG' AND 'BG'.



EXAMPLES (FOREGROUND)

\*\*\*\*\*  
\*\*\*\*\* EXAMPLES OF RUNNING THIS ON 'TSO' \*\*\*\*\*  
\*\*\*\*\*

ALL EXAMPLES FOR 'TSO' ASSUME YOU HAVE KEYED IN 'SPGEN DSNAME'

EX1: FIND(3)=C'CHARS TO LOOK FOR' \* COLUMN 3 FOR LENGTH STRING.  
FIND(1,80)=X'090612' \* SEARCH COLUMNS 1,80 FOR HEX STUFF.  
COUNT=1 \* DO IT FOR ONLY 1 RECORD.  
KEEP=Y,GO \* KEEP THIS INFORMATION AND 'GO' AND DO IT  
GO \* AND GO DO IT AGAIN FOR THE SAME.

THE ABOVE WILL GIVE A 'HEXDUMP' OF ANY 'HIT' ON THE SCREEN  
EACH TIME 'GO' IS ENTERED (DEFAULT IS DUMP=Y FOR TSO).

EX2: DUMP=N,OUTDD=SYSUT2,SKIP=6,COUNT=2,GO  
D=N,O=SYSUT2,S=6,C=2,G \* SAME AS ABOVE EXCEPT ABBREVIATED.  
D N,O SYSUT2,S 6,C 2,G \* SAME AS ABOVE EXCEPT MORE SO.

THIS WOULD RESULT IN A 'CHARACTER' DISPLAY OF RECORDS 7 AND  
8 ON SYSUT2 WHICH IS DIRECTED TO THE SCREEN.

EX3: BLKSIZE(4560),COUNT(5),GO

NOTICE THE USE OF '(' AND ')' WHICH IS OK.  
THIS DUMPS 5 BLOCKS OF DATA EACH A MAXIMUM OF 4560 BYTES.  
THE DATA IS NOT DEBLOCKED.

EX4: DUMP=N,OUTDD=SYSUT2,GO

THIS WOULD LIST ON THE SCREEN (SYSUT2) THE DATA IN  
'CHARACTER' AS IT APPEARS IN THE FILE.

EXAMPLES (BACKGROUND)

\*\*\*\*\*  
\*\*\*\*\* EXAMPLES OF RUNNING THIS IN BATCH \*\*\*\*\*  
\*\*\*\*\*

THE JCL DEFINING THIS JOB IS IN A PROC CALLED 'SPGEN'. IT HAS  
SYMBOLICS : OUT=A FOR SYSPRINT OUTPUT CLASS  
GENPARM= FOR PARM FIELD ENTRY OF ANY COMMANDS.

EX5: //SPGEN EXEC SPGEN,OUT=T  
//SYSUT1 DD DSN=YOUR.IN.DATA,DISP=SHR (INPUT DSN)

```
//SYSUT2 DD DSN=YOUR.OUT.DATA,DISP=SHR (OUTPUT DSN)
```

IF NO COMMANDS ARE ENTERED THE DEFAULT IS TO COPY THE DATASET TO SYSUT2 AS IN A IEBCGENER.

```
EX6: //SPGEN EXEC SPGEN,OUT=T
//SYSUT1 DD DSN=INPUT,DISP=SHR
//SYSUT2 DD DSN=OUTPUT,DISP=SHR
//SYSIN DD *
DUMP=Y,SKIP=23,COUNT=78,GO
```

THIS WILL GIVE A DUMP OF 78 RECORDS (24-101) AND COPY THEM TO SYSUT2.

```
EX7: //SUPERGEN EXEC SPGEN,OUT=T
//INDD1 DD DSN=SOME.INDATA,DISP=SHR
//INDD2 DD DSN=SOME.INDATA2,DISP=SHR
//OUTDD DD DSN=OUT.DATA.SET,DISP=NEW,...
//SYSIN DD *
INDD=INDD1,OUTDD=OUTDD,GO
INDD=INDD2,GO
/*
```

THE ABOVE MERGES INDD1+INDD2 AND PLACES IT IN OUTDD. YOU COULD HAVE CONTROLLED THE NUMBER AND TYPE OF RECORDS WITH SKIP,COUNT, AND FIND.

E.G. INDD=INDD1,SKIP=5,FIND(1)=C'THISONE',OUTDD=OUTDD,GO

```
EX8: //SUPERGEN EXEC SPGEN,OUT=T
//SYSUT1 DD DSN=INPUT.TO.SEARCH,DISP=SHR
//SYSUT2 DD DSN=SELECTED.OUTPUT,DISP=SHR
//SYSIN DD *
DUMP=Y * HEXDUMP OF DATA ON SYSPRINT
FIND(1,80) SKIP X'0571AC' * START AT THIS HEX STRING.
FIND(3)=X'65' * AND PROCESS ALL THESE TYPE OF RECORDS.
FIND(20) NE STOP X'95' * BUT STOP WHEN COLUMN 20 NE X'95'.
COUNT=500 * OR WHEN 500 RECORDS HAVE BEEN PROCESSED.
GO * GO AND DO IT
```

THE ABOVE LOOKS FOR X'0571AC', IT THEN START SELECTING THAT RECORD AND 499 MORE RECORDS (=500) OR UNTIL X'95' IS FOUND IN COLUMN 20. IT THEN LISTS THE FILE IN HEX AND COPIES IT.

```
EX9: //SPGEN EXEC SPGEN,OUT=T
//SYSUT1 DD DSN=YOUR.DATA.SET,DISP=SHR
DUMP=Y,OUTDD=N * DUMP ONLY
AND * ALL OF THE 'FIND'S MUST BE TRUE.
FIND(3) LT X'0010' * COLUMN 3 LESS THAN '0010'
FIND(3) GT X'0006' * AND GREATER THAN '0006'.
GO * GO AND DO
```

```
EX10: //SPGEN EXEC SPGEN,OUT=T
```

PEOPLES NATIONAL BANK UTILITIES

040.000.000 UTILITIES

.030 SUPERGEN - FILE SCAN/DUMP UTILITY

4 30 87 PAGE 28

```
//SYSUT1 DD DSN=INPUT.DATA.SET,DISP=SHR
//SYSUT2 DD DSN=OUTPUT.DATA.SET,DISP=SHR
PUNCH C' ' * DEFAULT FILL IS A BLANK.
PUNCH 1 C'//JOBNAME JOB (...ETC...) * A JOB CARD
COUNT=0,GO * CREATE A RECORD.
PUNCH C' ',PUNCH 1 C'//STEP1 EXEC PGM=SOMEPROG
COUNT=0,GO * CREATE A RECORD.
PUNCH X'00' * DEFAULT X'00' FILL.
PUNCH 3 X'231896' * THESE HEX STARTING IN POSITION 3.
PUNCH 15 C'SOME DATA' * SOME CHARACTER DATA.
PUNCH 35 X'574123' * AND SOME MORE HEX.
COUNT=0,GO * CREATE A RECORD WITH THE ABOVE DATA.
GO * NOW COPY THE REST OF THE INPUT TO OUTPUT.
```

THE ABOVE ASSUMES OUTDD=SYSUT2. IF SYSUT1/2 HAD AN  
 LRECL=80 THEN THE FOLLOWING IS CREATED IN SYSUT2:

```
//JOBNAME JOB (...ETC...) RECORD 1.
//STEP1 EXEC PGM=SOMEPROG RECORD 2.
...THIRD RECORD WITH HEX AND CHAR... RECORD 3.
....THE REST OF INPUT... RECORDS 4 ...ETC.
```

```
EX11: //SPGEN EXEC SPGEN,OUT=T
//SYSUT1 DD DSN=DATA.TO.BE.COPIED,DISP=SHR
//SYSUT2 DD DSN=OUTPUT.DATA,DISP=SHR
MOVE(1,80) C' ' * FILL WITH BLANKS
MOVE(1,4,10) * MOVE COLUMN 10-13 TO COLUMN 1-4.
MOVE(5,13,1) * MOVE COLUMN 1-9 TO COLUMN 5-13.
GO * CREATE THE NEW FORMAT OUTPUT.
```