

\*)9BEG

% =====  
% = ROUTINE 21.1 =  
% =====

%  
% PROGRAM TO LAY OUT A SINTRAN III SYSTEM  
% ON A MAG-TAPE  
%  
% VALID FOR: S-III 79.07.15.A  
%  
% MODIFIED : 79.06.12 (BHQ)

MAGTDUMP-NPL % 54 MB

\*)9ENT MTAP1

BASE AA  
INTEGER FORM,CUNIT,CHCB,SAVX,SAVT,BUFFP:=BUFFER  
INTEGER DESTFILE,SOURCEFILE  
INTEGER POINTER OUTTEXT:=9OUTTEXT,READTEXT:=8READTEXT,INBT:=9INBT  
INTEGER POINTER OUTBT:=9OUTBT,MOVCHAR:=9MOVCHAR  
INTEGER POINTER STSPACE:=9STSPACE,STAPO:=9STAPO,STCHAR:=9STCHAR  
INTEGER PCINTER MON70:=9MON70,COPYTEXT:=TEXTCOPY,COPYFILE:=FILECOPY  
INTEGER POINTER LINK1,LINK2  
ESAB  
  
INTEGER APERIFILE:='\$PERIPHERAL FILE NAME: '  
INTEGER AUNLOAD:='\$UNLOAD THE TAPE? '  
INTEGER DEVFU:='DEVICE-FUNCTION'  
INTEGER REWIND:='REWIND'  
INTEGER ENDFILE:='WRITE-EOF'  
INTEGER UNLOAD:='UNLOAD'  
INTEGER MTLOAD:='(SYSTEM)MTLOADER:DATA'  
INTEGER MACM:='(SYSTEM)MACM-1718D:PROG'  
INTEGER PTERMINATE:='\$--- FINISHED ---\$'  
INTEGER FILCOPY:='COPY-FILE ('  
INTEGER ARRAY DIRNAM(40),PERIFILE(40),DEVNAM(40),BUFFER(140),YESNO(40)  
  
INTEGER SAVB,SAVL  
DISP 0  
INTEGER POINTER JOBNO                   % PARAMETER POINTER  
PSID

DISP 7  
INTEGER FINO  
PSID

% DISPLACEMENT IN FILE-NAMES

%% CONFIGURATION INDEPENDENT FILES.  
INTEGER BPUN1:=(SINTRAN)BPUN1:SYMB'  
INTEGER BPUN2:=(SINTRAN)BPUN2:SYMB'  
INTEGER BPUN4:=(SINTRAN)BPUN4:SYMB'  
INTEGER ULST1:=(SINTRAN)ULIST1:SYMB'  
INTEGER ULST2:=(SINTRAN)ULIST2:SYMB'  
INTEGER ULST4:=(SINTRAN)ULIST4:SYMB'

%% CONFIGURATION DEPENDANT FILES. (USER-NAME MUST BE 7 CHARACTERS.)  
INTEGER BPUN5:=(AUX-SIN)BPUN-00-5:SYMB'  
INTEGER BPUN6:=(AUX-SIN)BPUN-00-6:SYMB'  
INTEGER BPUN7:=(AUX-SIN)BPUN-00-7:SYMB'  
INTEGER BPUN8:=(AUX-SIN)BPUN-00-8:SYMB'  
INTEGER BPUNA:=(AUX-SIN)BPUN-00-A:BPUN'  
INTEGER BPUNB:=(AUX-SIN)BPUN-00-B:BPUN'  
INTEGER BPUNE:=(AUX-SIN)BPUN-00-E:BPUN'  
INTEGER BPUNX:=(AUX-SIN)BPUN-00-X:BPUN'  
INTEGER L5IST:=(AUX-SIN)LIST-00:SYMB'

%% JOB NUMBERS IN ASCII. (00-99)  
@ICR;  
INTEGER ARRAY JBNO:=(  
30060,30061,30062,30063,30064,30065,30066,30067,30070,30071,  
30460,30461,30462,30463,30464,30465,30466,30467,30470,30471,  
31060,31061,31062,31063,31064,31065,31066,31067,31070,31071,  
31460,31461,31462,31463,31464,31465,31466,31467,31470,31471,  
32060,32061,32062,32063,32064,32065,32066,32067,32070,32071,  
32460,32461,32462,32463,32464,32465,32466,32467,32470,32471,  
33060,33061,33062,33063,33064,33065,33066,33067,33070,33071,  
33460,33461,33462,33463,33464,33465,33466,33467,33470,33471,  
34060,34061,34062,34063,34064,34065,34066,34067,34070,34071,  
34460,34461,34462,34463,34464,34465,34466,34467,34470,34471);  
&CR;

%%

%% ENTRYPOINT CALLED FROM FORTRAN.

SUBR MTAP1  
MTAP1: A=:B=:SAVB  
A=:L=:SAVL

X=:JOBNO; A=:JBNO(X) % GET PARAMETER AND ASCII EQUIV.  
X:="BPUN5"; A=:X.FINO % MODIFY FILENAMES WITH  
X:="BPUN6"; A=:X.FINO % JOB-NUMBER.  
X:="BPUN7"; A=:X.FINO %  
X:="BPUN8"; A=:X.FINO %  
X:="BPUNA"; A=:X.FINO %  
X:="BPUNB"; A=:X.FINO %  
X:="BPUNE"; A=:X.FINO %  
X:="BPUNX"; A=:X.FINO %  
X:="L5IST"; A=:X.FINO %

CALL DUMP  
SAVB=:B  
SAVL=:L  
EXIT

RBUS  
%% RETURN TO FORTRAN.

SUBR DUMP  
DUMP: "AA"=:B  
A=:L:="LINK2"  
O=:FORM=:CUNIT

% READ THE PERIPHERAL FILE NAME OF THE MAG-TAPE  
LX: "APERIFILE"; CALL OUTTEXT; T:="PERIFILE"; CALL READTEXT

% REWIND THE MAG-TAPE  
O=:CHCB  
T:="DEVFU"; CALL MOVCHAR; CALL STSPACE  
T:="PERIFI"; CALL MOVCHAR; T:="REWIND"; CALL MOVCHAR  
CALL STAPO; CALL MON70

% OPEN THE MAG-TAPE FOR WRITE  
O=:CHCB  
T:="PERIFI"; CALL MOVCHAR  
CALL STAPO; X=:BUFFP; T=:0; \*MON 50; MON 65  
A=:DESTFILE

% COPY MT-LOAD PROGRAM TO THE MAG-TAPE  
X:="MTLOAD"; T=:1; \*MON 50; MON 65

```

A=:SOURCEFILE; CALL COPYFILE

% COPY MACM TO THE MAG-TAPE
T=:SOURCEFILE; *MON 43; MON 65
X:="MACM"; T:=1; *MON 50; MON 65
A=:SOURCEFILE; CALL COPYFILE

% CLOSE ALL FILES BEFORE USING DEVICE-FUNCTION
T:=-1; *MON 43; JMP *+1

% WRITE EOF AFTER THE MACM-FILE
O=:CHCB
T:="DEVFU"; CALL MOVCHAR; CALL STSPACE
T:="PERIFI"; CALL MOVCHAR; T:="ENDFILE"; CALL MOVCHAR
CALL STAPO; CALL MON70

% OPEN THE MAG-TAPE AGAIN FOR WRITE
O=:CHCB
T:="PERIFI"; CALL MOVCHAR
CALL STAPO; X:=BUFFP; T:=0; *MON 50; MON 65
A=:DESTFILE; GO LA; *)FILL

% COPY TEXT TO THE MAG-TAPE
LA: "TXT0"; CALL COPYTEXT
"TXT1"; CALL COPYTEXT; "TXT9"; CALL COPYTEXT

% OPEN THE FIRST BINARY FILE (FILE SYSTEM)
X:="BPUN5"; T:=1; *MON 50; MON 65
A=:SOURCEFILE; CALL COPYFILE

% COPY TEXT TO THE MAG-TAPE
"TXT8"; CALL COPYTEXT

% COPY NEXT BINARY FILE TO MAG-TAPE
T=:SOURCEFILE; *MON 43; MON 65
X:="BPUN7"; T:=1; *MON 50; MON 65
A=:SOURCEFILE; CALL COPYFILE

% COPY TEXT TO THE MAG-TAPE
"TXT8";CALL COPYTEXT

% COPY SWAP-DRIVER TO MAG-TAPE
T=:SOURCEFILE; *MON 43; MON 65
X:="BPUNB"; T:=1; *MON 50; MON 65
A=:SOURCEFILE; CALL COPYFILE

"TXT3"; CALL COPYTEXT

% OPEN THE NEXT BINARY FILE AND COPY TO MAG-TAPE
T=:SOURCEFILE; *MON 43; MON 65
X:="BPUN6"; T:=1; *MON 50; MON 65
A=:SOURCEFILE; CALL COPYFILE

% COPY TEXT TO THE MAG-TAPE
"TXT11"; CALL COPYTEXT

% OPEN THE NEXT BINARY FILE AND COPY TO MAG-TAPE
T=:SOURCEFILE; *MON 43; MON 65
X:="L5IST"; T:=1; *MON 50; MON 65
A=:SOURCEFILE; CALL COPYFILE

% COPY TEXT TO THE MAG-TAPE
"TXT12"; CALL COPYTEXT; "TXT8"; CALL COPYTEXT
GO L5; *)FILL

% COPY THE NEXT BINARY FILE TO THE MAG-TAPE
L5:T=:SOURCEFILE; *MON 43; MON 65
X:="BPUN?"; T:=1; *MON 50; MON 65
A=:SOURCEFILE; CALL COPYFILE

% COPY TEXT TO MAG-TAPE
"TXT11"; CALL COPYTEXT

% ULIST2 TO MAG-TAPE
T=:SOURCEFILE; *MON 43; MON 65
X:="ULST2"; T:=1; *MON 50; MON 65
A=:SOURCEFILE; CALL COPYFILE

% COPY TEXT TO THE MAG-TAPE
"TXT8"; CALL COPYTEXT

% COPY NEXT BINARY FILE TO MAG-TAPE
T=:SOURCEFILE;*MON 43;MON 65
X:="BPUNE";T:=1;*MON 50;MON 65
A=:SOURCEFILE;CALL COPYFILE

%TEXT TO MAG-TAPE
"TXT10"; CALL COPYTEXT

% NEXT BINARY FILE TO MAG-TAPE
T=:SOURCEFILE; * MON 43; MON 65
X:="BPUN1"; T:=1; *MON 50; MON 65
A=:SOURCEFILE; CALL COPYFILE

% TEXT TO MAG-TAPE

```

"TXT11"; CALL COPYTEXT

```
% ULIST1 TO MAG-TAPE
T:=SOURCEFILE; *MON 43; MON 65
X:="ULST1"; T:=1; *MON 50; MON 65
A:=SOURCEFILE; CALL COPYFILE; GO L6; *)FILL
```

```
% TEXT TO MAG-TAPE
L6: "TXT13"; CALL COPYTEXT
"TX13A"; CALL COPYTEXT
T:=SOURCEFILE; *MON 43; MON 65
X:="BPUNX"; T:=1; *MON 50; MON 65
A:=SOURCEFILE; CALL COPYFILE
"TX13B"; CALL COPYTEXT
```

```
"TXT14"; CALL COPYTEXT
T:=SOURCEFILE; *MON 43; MON 65
X:="BPUNA"; T:=1; *MON 50; MON 65
A:=SOURCEFILE; CALL COPYFILE
"TXT15"; CALL COPYTEXT
```

```
"TXT9"; CALL COPYTEXT
```

```
% NEXT BINARY FILE TO MAG-TAPE
T:=SOURCEFILE; *MON 43; MON 65
X:="BPUN4"; T:=1; *MON 50; MON 65
A:=SOURCEFILE; CALL COPYFILE
```

```
% TEXT TO MAG-TAPE
"TXT11"; CALL COPYTEXT
```

```
% ULIST4 TO MAG-TAPE
T:=SOURCEFILE; *MON 43; MON 65
X:="ULST4"; T:=1; *MON 50; MON 65
A:=SOURCEFILE; CALL COPYFILE
```

```
"TXT8"; CALL COPYTEXT
T:=SOURCEFILE; *MON 43; MON 65
X:="BPUN8"; T:=1; *MON 50; MON 65
A:=SOURCEFILE; CALL COPYFILE
```

```
% TEXT TO MAG-TAPE
"TXT5"; CALL COPYTEXT; "TXT6"; CALL COPYTEXT
"TXT7"; CALL COPYTEXT
```

```
% CLOSE ALL FILES
T:=-1; *MON 43; JMP *+1
```

```
% WRITE END OF FILE
O=:CHCB
T:="DEVFU"; CALL MOVCHAR; CALL STSPACE
T:="PERIFI"; CALL MOVCHAR; T:="ENDFILE"; CALL MOVCHAR
CALL STAPO; CALL MON70
GO ASK1; *)FILL
```

```
% ASK IF MAG-TAPE SHOULD BE UNLOADED
```

```
ASK1: O=:FORM
"AUNLOAD"; CALL OUTTEXT; T:="YESNO"; CALL READTEXT
X=:O; "YESNO"=:D; *LBYT; AAX 1
IF A=#Y THEN
    T=:D; *LBYT; AAX 1
    IF A><##E AND A><##' GO ASK1
    IF A=##' GO YES
    T=:D; *LBYT; AAX 1
    IF A><##S AND A><##' GO ASK1
    IF A=##' GO YES
    T=:D; *LBYT; AAX 1
    IF A><##' GO ASK1
    1=:FORM
```

YES:

```
ELSE
    IF A=#N THEN
        T=:D; *LBYT; AAX 1
        IF A><##O AND A><##' GO ASK1
        IF A=##' GO L2
        T=:D; *LBYT
        IF A><##' GO ASK1
```

```
ELSE
    GO ASK1
FI; FI
GO L2; *)FILL
```

```
% REWIND THE TAPE
```

```
L2: O=:CHCB
T:="DEVFU"; CALL MOVCHAR; CALL STSPACE
T:="PERIFI"; CALL MOVCHAR; T:="REWIND"; CALL MOVCHAR
CALL STAPO; CALL MON70
```

```
IF FORM><O THEN
```

```
% UNLOAD THE TAPE
O=:CHCB
T:="DEVFU"; CALL MOVCHAR; CALL STSPACE
T:="PERIFI"; CALL MOVCHAR; T:="UNLOAD"; CALL MOVCHAR
CALL STAPO; CALL MON70
```

FI

```
% PROGRAM IS TERM : TED
"PTERM"; CALL OUTTEXT
GO LINK2
```

RBUS

```
%
% READTEXT
% SUBROUTINE TO READ A TEXT STRING FROM THE TERMINAL
% THE INPUT SHOULD BE TERMINATED WITH CARRIAGE RETURN
%
% ENTRY: T=ADMS OF THE ARRAY WHERE THE TEXT STRING SHOULD
% BE STORED
%
% RETURN: TEXT STRING TERMINATED WITH THE CHARACTER "'" IN THE ARRAY
%
```

```
SUBR 8READTEXT
8READTEXT: A:=L:="LINK1"
X:=0; T:=0
DO CALL INBT WHILE A<15
IF A=21 THEN ##_; CALL OUTBT; X:=0; GO NEXT FI
IF A=1 THEN ##^; CALL OUTBT; X-1; GO NEXT FI
IF X>100 GO NEXT
T:=0; *SBYT; AAX 1
NEXT: OD; T:=0; ##'; *SBYT
GO LINK1
```

RBUS

```
%
% INBT
% SUBROUTINE TO READ ONE CHARACTER FROM THE TERMINAL
%
% RETURN: A=CHARACTER
%
```

```
SUBR 9INBT
9INBT: T:=SAVT:=1; *MON 1; MON 65
A/0177; T:=SAVT; EXIT
```

RBUS

```
%
% MOVCHAR
% SUBROUTINE TO MOVE CHARACTERS FROM A TEXT STRING TO AN ARRAY
%
% ENTRY: T=ADMS OF TEXT STRING
%
```

```
SUBR 9MOVCHAR
9MOVCHAR: A:=L:="LINK1"
T:=0; X:=0
DO T:=0; *LBYT
WHILE A><##'; CALL STCHAR; X+1
OD; GO LINK1
```

RBUS

```
%
% STCHAR
% SUBROUTINE TO STORE ONE CHARACTER IN THE ARRAY NAMED BUFFER
%
% ENTRY: A=CHARACTER
%
```

```
SUBR 9STCHAR
9STCHAR: X:=SAVX:=CHCB; T:=BUFFP; *SBYT
MIN CHCB; X:=SAVX; EXIT
```

RBUS

```
%
% STAPO - STSPACE
% SUBROUTINES TO STORE A SPACE OR A "'" IN THE ARRAY NAMED BUFFER
%
```

```
SUBR 9STAPO,9STSPACE
9STAPO: ##'; GO STCHAR
9STSPACE: 40; GO STCHAR
RBUS
```

```
%
% OUTTEXT
% SUBROUTINE TO PRINT A TEXT STRING ON THE TERMINAL
%
% ENTRY: A=ADMS OF THE TEXT STRING
%
```

```
SUBR 9OUTTEXT
9OUTTEXT: T:=L:="LINK1"
A:=0; X:=0
DO T:=0; *LBYT
WHILE A><##'
IF A=##$ THEN 15; CALL OUTBT; 12 FI
CALL OUTBT; X+1
OD; GO LINK1
```

RBUS



%% A = CRMAX , B = CLM , C = COADR , D = LONG , E = XDSKT  
%% F = MSTYP , G = DEVNO , H = DASA , L = MACAD , M = DRES  
%% N = FLTS , O = RTAD , P = 99MRE , Q =  
%%%

A=175777  
R=100  
C=30000  
D=30000  
E=0

)MCDEF REMOV  
F=2  
G=500  
H=10  
M=2000  
L=1000  
N=M  
O=4200  
P=L  
Q=3600  
R

)MCDEF FIXED  
F=2  
G=500  
H=100010  
M=102000  
L=101000  
N=M  
O=104200  
P=L  
Q=103600  
R

)MCDEF DRUM  
F=0  
G=540  
H=40  
M=4000  
L=2000  
N=M  
O=10400  
P=L  
Q=7400  
R

)MCDEF BD33  
F=3  
G=1540  
H=2  
L=200  
M=400  
N=M  
O=1040  
P=L  
Q=740  
R

)MCDEF BD38  
F=4  
G=1540  
H=2  
L=200  
M=400  
N=M  
O=1040  
P=L  
Q=740  
R

)MCDEF BD66  
F=3  
G=1540  
H=2  
L=200  
M=400  
N=M  
O=1040  
P=L  
Q=740  
R

)MCDEF BD75  
F=4  
G=1540  
H=2  
L=200  
M=400  
N=M  
O=1040  
P=L  
Q=740  
)KILL E  
R

```
)MCDEF BD?
F=5
G=1540
H=2
L=200
M=400
N=M
O=1040
P=L
Q=740
E=100000
R
)MCDEF BDREM
F=6
G=1540
H=2
L=200
M=400
N=M
O=1040
P=L
Q=740
R
)MCDEF BDFIX
F=6
G=1540
H=100002
L=100200
M=100400
N=M
O=101040
P=L
Q=100740
R
```

```
)9ASSM INDEV,1
%%
%% CHOOSE MASS STORAGE BY GIVING ONE OF THE FOLLOWING CODES
%% FOLLOWED BY CARRIAGE RETURN AND LINE FEED
%%
%% IF SYSTEM ON BIG-DISC 33 MEGABYTES, TYPE: BD33
%% IF SYSTEM ON BIG-DISC 38 MEGABYTES, TYPE: BD38
%% IF SYSTEM ON BIG-DISC 66 MEGABYTES, TYPE: BD66
%% IF SYSTEM ON BIG-DISC 75 MEGABYTES, TYPE: BD75
%% IF SYSTEM ON BIG-DISC 288 MEGABYTES, TYPE: BD288
%% IF SYSTEM ON REMOVABLE CARTRIDGE DISC, TYPE: REMOV
%% IF SYSTEM ON FIXED CARTRIDGE DISC, TYPE: FIXED
%% IF SYSTEM ON DRUM, TYPE: DRUM
%% IF SYSTEM ON REMOVABLE PHOENIX DISC, TYPE: BDREM
%% IF SYSTEM ON FIXED PHOENIX DISC, TYPE: BDFIX
%%
%% TYPE XX,0$ AND WHEN THE MACHINE ANSWERS WITH CARRIAGE RETURN
%% AND LINE FEED, TYPE XX,0$ TO CONTINUE
1,0$
'
```

```
TXT3, '
)9BYTT F G C D B N L A L H
)9ASSM INDEV,0,INDEV
)9READ
'
```

```
TXT5, ')9ASSM INDEV,1
%%
%% ALL CORRECTIONS (PATCHES) EXCEPT IN THE FILE SYSTEM AND
%% RT-LOADER MAY BE DONE NOW.
%%
%% AFTER YOUR CORRECTIONS, TYPE XX,0$
1,0$
'
```

```
TXT6, ')9ASSM INDEV,1
%%
%% )GJEM AND )9BYTT WILL NOW BE EXECUTED
%%
)9ASSM INDEV,0
30/M
2/E
)GJEM
)KILL A
A=77777
)9BYTT F G C D B N M A L H
'
```

```
TXT7, ')9ASSM INDEV,1
%%
%%
%%
%% IF YOU WANT A CTOM-TAPE, (SEE MACM-MANUAL)
%% PAPER-TAPE-PUNCH MUST BE USED.
%%
%% TYPE 1,0,3$ AND )9CTOM
%%
%% AFTERWARDS,
%% THE RSINTRAN III SYSTEM MAY BE STARTED BY TYPING 22!
```



```
%%  
1,0$  
,  
TXT8, ' )9ASSM INDEV,0,INDEV  
)9READ  
,
```

```
TXT9, '  
)9BYTT F G C D B N M A L H  
)9ASSM INDEV,0,INDEV  
)9READ  
,
```

```
TXT10, ' )9ASSM INDEV,1
```

```
%%  
%% ALL CORRECTIONS (PATCHES) IN THE FILE SYSTEM MAY BE DONE NOW  
%% AFTER YOUR CORRECTIONS, TYPE XX,0$  
%%
```

```
1,0$  
)9ASSM INDEV,1
```

```
%%  
%% TYPE XX,0$ AND WHEN THE MACHINE ANSWERS WITH CARRIAGE RETURN  
%% AND LINE FEED, TYPE XX,0$ TO CONTINUE  
%%
```

```
)9ASSM INDEV,0  
)9BYTT F G C D B N O A L H  
)9ASSM INDEV,0,INDEV  
)9READ  
,
```

```
TXT11, ' )9ASSM INDEV,0
```

```
TXT12, ' )9ASSM INDEV,0
```

```
TXT13, ' )9ASSM INDEV,1
```

```
%%  
%% ALL CORRECTIONS (PATCHES) IN THE RT-LOADER MAY BE DONE NOW  
%% AFTER YOUR CORRECTIONS, TYPE XX,0$, AND WHEN THE MACHINE  
%% ANSWERS WITH CARRIAGE RETURN AND LINE FEED,  
%% TYPE XX,0$ TO CONTINUE  
%%
```

```
1,0$  
,
```

```
TX13A, '  
)9BYTT F G C D B N Q A L H  
)9ASSM INDEV,0,INDEV  
)9READ  
,
```

```
TX13B, '  
)9ASSM INDEV,1  
%% DUMPING OF SPOOLING-PROGRAM DONE.  
)9ASSM INDEV,0  
,
```

```
TXT14, '  
)9BYTT F G C D B N P A L H  
)9ASSM INDEV,0,INDEV  
)9READ  
,
```

```
TXT15, ' )9ASSM INDEV,1
```

```
%%  
%% ALL CORRECTIONS IN THE "PAGING-OFF" AREA MAY BE DONE NOW  
%%
```

```
1,0$  
,
```

```
)9END  
)9EOF  
@  
@EOF
```