APPENDIX A
AMCC-SUPPLIED SHELL SCRIPTS

APPENDIX A

THE SHELLS:

RUN_AGIF	NETLIST GENERATION
RUN_AMCC	SUPER SHELL
RUN_DD	DANCE AND DRINK
RUN_ERC	AMCC ERC
RUN_ANN	AMCC ANNOTATION
RUN_SIFT	SIFT (TIMING)
RUN_SMAKER	CREATE SOM_MCF.SING AND FMT
RUN_SMT	AMCCSIMFMT
RUN_SOM	INTEGRATION
RUN_VRC	AMCCVRC

THESE ARE SUPPLIED AS PART OF THE AMCC MACROMATRIX SOFTWARE

ONCE A SCHEMATIC IS CAPTURED, TYPE:

```
      RUN_AMCC
```

THE SUPER SHELL CALLS THE OTHER SHELLS WHICH IN TURN CALL OUT THE VARIOUS COMMANDS TO EXECUTE THE DAISY AND THE AMCC SOFTWARE IN THE PROPER ORDER OF EXECUTION

THE MENU WILL ASK FOR A CHOICE OF 1, 2 OR 3

```
    1 WILL CAUSE RUN_DD   DANCE-DRINK TO A DAISY NETLIST
                 RUN_AGIF REFORMAT TO AMCC AGIF NETLIST
                 RUN_ERC  ERC SOFTWARE
```

AFTER DEBUG OF ANY ERRORS,

```
    2 WILL CAUSE RUN_ANN  ANNOTATION


    3 WILL CAUSE RUN_SIFT   TIMING LIBRARY SELECTION
                 RUN_SMAKER SING-TO-DAISY

      THIS WILL END WITH: TEC SOM_MCF.SING
```

AFTER EDITING THE SIMULATION CONTROL FILE, ON EXITING TEC, THE MENU RETURNS

```
    0 WILL CAUSE AN EXIT
```

IF OTHER EDITING SUCH AS DATA FILE CREATION IS REQUIRED, SELECT 0, ELSE,

```
    4 WILL CAUSE RUN_SOM    SOM AND TCAL STEPS
```

FROM HERE, EXIT USING 0 AND RUN DLS OR DTV AS DESIRED

ALSO, USE TEC TO CREATE THE SIGNAL ANALYSIS FILE

RECALL RUN_AMCC

```
    5 WILL CAUSE RUN_SMT    AMCCSIMFMT

    6 WILL CAUSE RUN_VRC    AMCCVRC
```

YOU MAY RUN THE SUB-SHELLS BY CALLING THEM INDIVIDUALLY

THE PROPER ORDER IS:

```
     RUN_DD
     RUN_AGIF
     RUN_ERC
```

DEBUG AS NEEDED

```
     RUN_ANN
     RUN_SIFT
     RUN_SMAKER
```

EDIT SIMULATION CONTROL FILE
CREATE A DATA FILE IF DESIRED

```
     RUN_SOM
```

FROM HERE, RUN DLS OR DTV AS DESIRED

AND WHEN SIMULATION IS COMPLETE, TYPE

```
     RUN_SMT
```

AFTER CREATING THE SIGNAL ANALYSIS FILE, TYPE

```
     RUN_VRC
```

YOU MAY ALSO RUN THE REQUIRED STEPS BY TYPING IN THE INDIVIDUAL SPECIFIC COMMANDS

THE COMMANDS:

```
DANCE -T -N -ERR -M3
DRINK -T -M3

RUN_AGIF                      SING-TO-AGIF; USE THIS SHELL

AMCCERC

AMCCANN

SIFT -M3 -LIB $FAMILY/PATCH$1.SLIB $FAMILY/Q50IO$1.SLIB\
        $FAMILY/Q50GE$1.SLIB $FAMILY/Q50RA$1.SLIB

SING -M3 -MCF /AMCC/SOM_MAKER/SOM_MCF
FMT_CSD.SING
TEC SOM_MCF.SING
    (EDIT THE SIMULATION CONTROL FILE)

SOM SOM_MCF.SING -M3 -L SOM.ERR
TCAL -MCF FNTxxx.DSY -M3
```

FROM HERE, RUN DLS OR DTV AS DESIRED

```
       DLS
       GET FMT
       MODE
         (EDIT)
       {ENTER}
       VIEW 9999 10000
       RUN nnnnnnnn
       START 0
       LIST OR WAVE AS DESIRED
       {ENTER}
       QUIT
       N
```

THE $OUTPUTS SECTION FILE DEFINED IN THE SOM_MCF.SING FILE IS THE INPUT TO AMCCSIMFMT

```
AMCCSIMFMT
```

THE OUTPUT OF AMCCSIMFMT IS INPUT TO AMCCVRC

```
AMCCVRC
```

```
Date:  14 SEP 88 14:42   File:  /AMCC/SUBMIT/RUN_AGIF

echo 'FALSE' > ERROR.ERR
echo '
        ********* RUNNING AGIF NETLISTER ***********
     '
MKDIR ERC 2> /DEV/NULL
AMCCAGIF > ERC/CIRCUIT.SDI 2> /DEV/NULL
if (test -s ERC/CIRCUIT.SDI)
then
   echo
else
  echo '        *** AGIF SYSTEM ERROR (Contact AMCC) !!!!! ***
        ;       **** PROBIBLE CAUSE: EXCCEDED REAL MEMORY ****' > AGIF_SYS.ERR
  echo '
        ********** AGIF SYSTEM ERROR !!! ***********
        **                                        **
       '***** PLEASE ENTER "TYPE AGIF_SYS.ERR" *****
       '
  echo 'TRUE' > ERROR.ERR
  exit
fi
if egrep -s 'ERROR' RUN_AGIF.ERR
then
  echo '
        ********* AGIF NETLISTER ERROR !!! *********
        **                                        **
       '***** PLEASE ENTER "TYPE RUN_AGIF.ERR" *****
       '
  echo 'TRUE' > ERROR.ERR
  exit
else
  echo '
       '********* AGIF NETLIST COMPLETE !!! ********
       '
fi
```

```
Date:  14 SEP 88 14:42   File:  /AMCC/SUBMIT/RUN_AMCC

menu=9
echo 'FALSE' > ERROR.ERR
while (test $menu -gt Ø)
do
   MAX_MENU.LIB
   echo -n '         Please Enter [Ø - 6] : '
   read menu
   echo
   if (test $menu -gt 6)
   then
      echo '         Only Values Between Ø through 6 are Valid...'
      exit
   fi
   if (test $menu -eq Ø)
   then
      exit
   fi
   if (test $menu -eq 1)
   then
      RUN_DD
      if egrep -s 'TRUE' ERROR.ERR
      then
         exit
      fi
      RUN_AGIF
      if egrep -s 'TRUE' ERROR.ERR
      then
         exit
      fi
      RUN_ERC
      if egrep -s 'TRUE' ERROR.ERR
      then
         exit
      fi
   fi
   if (test $menu -eq 2)
   then
      RUN_ANN
      if egrep -s 'TRUE' ERROR.ERR
      then
         exit
      fi
   fi
```

```
if (test $menu -eq 3)
then
   if egrep -s 'Q14ØØØ' ERC/CIRCUIT.SDI
   then
      echo -n '       Please Enter Product Grade (MIN, NOM, COM4, COM5 or MIL) : ';
   else
      echo -n '       Please Enter Product Grade (MIN, NOM, COM or MIL) : ';
   fi
   read grade
   EXPORT grade
   RUN_SIFT $grade
   if egrep -s 'TRUE' ERROR.ERR
   then
      exit
   fi
   RUN_SMAKER
   if egrep -s 'TRUE' ERROR.ERR
   then
      exit
   fi
   fi
   if (test $menu -eq 4)
   then
      echo -n '       Please Enter SOM Master Control File : ';
      read mcf
      EXPORT mcf
      echo
      echo -n '       Please Enter TCAL Delay File : ';
      read tcal
      EXPORT tcal
      RUN_SOM $mcf $tcal
      if egrep -s 'TRUE' ERROR.ERR
      then
         exit
      fi
   fi
   if (test $menu -eq 5)
   then
      RUN_SMT
      if egrep -s 'TRUE' ERROR.ERR
      then
         exit
      fi
   fi
   if (test $menu -eq 6)
   then
      RUN_VRC
      if egrep -s 'TRUE' ERROR.ERR
      then
         exit
      fi
   fi
done
```

```
Date:  14 SEP 88 14:42   File:  /AMCC/SUBMIT/RUN_ANN

echo 'FALSE' > ERROR.ERR
echo '
        ************** RUNNING AMCCANN *************
     '
cd ERC
echo -n '        Need to Edit Package Pin Data? (Y or N) ; '
read temp
echo ''
if (test $temp = 'Y' -o $temp = 'y')
then
   if (test -f OUTPUT.DLY)
   then
      AMCCANN <<!

!
   else
      AMCCANN <<!


!
   fi
else
   AMCCANN
fi
if egrep -s 'ERROR' AMCCANN.LST
then
  echo '
        ************ AMCCANN ERROR !!!!! ***********
        **                                        **
        ****** PLEASE ENTER "TYPE AMCCANN.LST"*******
       '
  cd ..
  echo 'TRUE' > ERROR.ERR
  exit
else
  COPY *.DSY TO ../*.DSY -B  2> /DEV/NULL
  rm *.DSY
  cd ..
  echo '
        ************* AMCCANN OK !!!!! *************
       '
fi
```

RUN_DD

```
Date:  14 SEP 88 14:42    File:  /AMCC/SUBMIT/RUN_DD

echo 'FALSE' > ERROR.ERR
echo '
        ****** RUNNING DANCE -T -N -ERR -M3 ********
     '
DANCE -T -N -ERR -M3  2> DANCE.ERR
if egrep -s 'PAGES FLAGGED. HIGHEST SEVERITY WAS Ø' DANCE.ERR
then
  echo '
        ************* DANCE OK !!!!! ***************
       '
else
  echo '
        ************ DANCE ERROR !!!!! *************
        **                                        **
        ****** PLEASE ENTER "TYPE DANCE.ERR" *******
       '
  echo 'TRUE' > ERROR.ERR
fi
echo '
        *********** RUNNING DRINK -T -M3 ***********
     '
DRINK -T -M3  2> DRINK.ERR
if egrep -s 'Ø BLOCKS FLAGGED. HIGHEST SEVERITY WAS Ø' DRINK.ERR
then
  echo '
        ************* DRINK OK !!!!! ***************
       '
else
  echo '
        ************ DRINK ERROR !!!!! *************
        **                                        **
        ****** PLEASE ENTER "TYPE DRINK.ERR" *******
       '
  echo 'TRUE' > ERROR.ERR
fi
```

RUN_ERC

```
Date:  14 SEP 88 14:42   File:  /AMCC/SUBMIT/RUN_ERC

echo 'FALSE' > ERROR.ERR
echo '
        ************* RUNNING AMCCERC **************
     '
cd ERC
AMCCERC
if egrep -s '^ERROR' AMCCERC.LST
then
  echo '
        ************ AMCCERC ERROR !!!!! ***********
        **                                        **
        **** PLEASE ENTER "TYPE ERC/AMCCERC.LST" ***
       '
  cd ..
  echo 'TRUE' > ERROR.ERR
  exit
else
  echo '
        ************* AMCCERC OK !!!!! *************
       '
fi
```

NOTE: /ERC/AMCCERC.LST IS THE ERC REPORT

```
Date:  14 SEP 88 14:42   File:  /AMCC/SUBMIT/RUN_SIFT

echo 'FALSE' > ERROR.ERR
if test -z "$1"
then
  echo '
        ******* NO GRADE VALUE ENTERED !!!!  *******
        **                                        **
        ** ENTER  "RUN_SIFT MIN, NOM, COM or MIL" **
        '
  echo 'TRUE' > ERROR.ERR
  exit
fi
echo "
        *************  RUNNING SIFT  ***************

        FAMILY = /AMCC/Q5ØØØ_LIBS       GRADE = $1
     "
SIFT -M3 -LIB /AMCC/Q5ØØØ_LIBS/PATCH$1.SLIB /AMCC/Q5ØØØ_LIBS/Q5ØI1$1.SLIB \
              /AMCC/Q5ØØØ_LIBS/Q5ØG1$1.SLIB /AMCC/Q5ØØØ_LIBS/Q5ØG2$1.SLIB \
              /AMCC/Q5ØØØ_LIBS/Q5ØG3$1.SLIB /AMCC/Q5ØØØ_LIBS/Q5ØM1$1.SLIB \
              /AMCC/Q5ØØØ_LIBS/Q5ØG1$1H.SLIB /AMCC/Q5ØØØ_LIBS/Q5ØG2$1H.SLIB \
              /AMCC/Q5ØØØ_LIBS/Q5ØG3$1H.SLIB /AMCC/Q5ØØØ_LIBS/Q5ØM1$1H.SLIB \
              /AMCC/Q5ØØØ_LIBS/Q5ØI1$1H.SLIB 2> SIFT.ERR
if egrep -s 'WRITING TREE SIF FILE' SIFT.ERR
then
  echo '
        ************* SIFT OK !!!!! ***************
        '
else
  echo '
        ************ SIFT ERROR !!!!! *************
        **                                        **
        ****** PLEASE ENTER "TYPE SIFT.ERR" ********
        '
  echo 'TRUE' > ERROR.ERR
  exit
fi
```

RUN_SMAKER

```
Date:  14 SEP 88 14:43   File:  /AMCC/SUBMIT/RUN_SMAKER
echo 'FALSE' > ERROR.ERR
echo '
       **********  RUNNING SING TO SOM   ************
    '
SING -M3 -MCF /AMCC/SOM_MAKER/SOM_MCF 2> SOM_MAKER.ERR
if egrep -s 'SING ERROR' SOM_MAKER.ERR
  then
  echo '
       ********  SING TO DAISY ERROR !!!!! *********
       **                                         **
      '***** PLEASE ENTER "TYPE SOM_MAKER.ERR" *****
  echo 'TRUE' > ERROR.ERR
  exit
else
  echo '
      '********** SING TO DAISY OK !!!!! ***********
fi
echo '
       *********** RUNNING FMT_CSD.SING ************
    '
FMT_CSD.SING 2>> SOM_MAKER.ERR
if egrep -s 'No user errors' SOM_MAKER.ERR
  then
  echo '
      '**********  FMT FORMAT OK !!!!!   ************
else
  echo '
       *********  FMT FORMAT ERROR !!!!! ***********
       **                                         **
      '***** PLEASE ENTER "TYPE SOM_MAKER.ERR" *****
  echo 'TRUE' > ERROR.ERR
  exit
fi
TEC SOM_MCF.SING
```

NOTE: THIS SHELL LEAVES YOU IN THE TEC EDITOR

RUN_SOM

```
Date:  14 SEP 88 14:42   File:  /AMCC/SUBMIT/RUN_SOM

echo 'FALSE' > ERROR.ERR
if test -z "$1"
then
  echo '
        **** NO SOM.MCF or TCAL.MCF ENTERED !!!! ***
        **                                        **
        ***** ENTER "RUN_SOM somfile tcalfile" *****
       '
  echo 'TRUE' > ERROR.ERR
  exit
fi
if test -z "$2"
then
  echo '
        ********* NO TCAL.MCF ENTERED !!!! *********
        **                                        **
        ***** ENTER "RUN_SOM somfile tcalfile" *****
       '
  echo 'TRUE' > ERROR.ERR
  exit
fi
echo '
        *********** RUNNING SOM and TCAL ************
      '
SOM $1 -M3 -L SOM.ERR
if egrep -s 'No user errors' SOM.ERR
  then
  echo '
        ***********  SOM FORMAT OK !!!!!  ***********
       '
else
  echo '
        *********  SOM FORMAT ERROR !!!!! ***********
        **                                         **
        ******* PLEASE ENTER "TYPE SOM.ERR" *********
       '
  echo 'TRUE' > ERROR.ERR
  exit
fi
TCAL -MCF $2 -M3 2> TCAL.ERR
if egrep -s '.' TCAL.ERR
then
  echo '
        *********  TCAL FORMAT ERROR !!!!! **********
        **                                        **
        ******* PLEASE ENTER "TYPE TCAL.ERR" ********
       '
  echo 'TRUE' > ERROR.ERR
  exit
else
  echo '
        ***********  TCAL FORMAT OK !!!!!  **********
       '
fi
```

Date: 14 SEP 88 14:43   File: /AMCC/SUBMIT/RUN_SMT

```
#
# This shell script will run AMCCSIMFMT
#
AMCCSIMFMT
```

Date: 14 SEP 88 14:43   File: /AMCC/SUBMIT/RUN_VRC

```
#
# This shell script will run AMCCVRC and delete any junk files left over
#
# History:        * Feb 19, 1988 - original implementation
#
#
#
AMCCVRC
rm ASF*.DAT
```