```
0001                * NAME BASIC-INIT-A          DOC. 70181909000    REV. A                  BS100010
0002                *                                                                        BS100020
0003                *                                                                        BS100030
0004                * DESCRIPTION INITIALIZATION ROUTINE FOR STAND-ALONE VERSION OF BASIC-16BS100040
0005                *                                                                        BS100050
0006                *                                                                        BS100060
0007                * REVISION HISTORY                                                       BS100070
0008                *    REV.       DATE          ECO NO.                                  BS100080
0009                *     A                       RELEASED                                 BS100090
0010                *                                                                        BS100100
0011                *                                                                        BS100110
0012                *                                                                        BS100120
0013                *                                                                        BS100130
0014                *                                                                        BS100140
0015                * DOCUMENTATION REFERENCES                                               BS100150
0016                *    TITLE                                DOC. NO.                       BS100160
0017                *    BASIC-16 USER'S MANUAL               70130072543                    BS100170
0018                *                                                                        BS100180
0019                *                                                                        BS100190
0020                *                                                                        BS100200
0021                *                                                                        BS100210
0022                *                                                                        BS100220
0023                * COPYRIGHT 1971 BY HONEYWELL INFORMATION SYSTEMS INC., COMPUTER         BS100230
0024                * SYSTEMS DIVISION, FRAMINGHAM, MASSACHUSETTS.  CONTENTS OF THIS        BS100240
0025                * PUBLICATION MAY NOT BE REPRODUCED IN ANY FORM, IN WHOLE OR IN         BS100250
0026                * PART, WITHOUT PERMISSION OF THE COPYRIGHT OWNER.  ALL RIGHTS          BS100260
0027                * RESERVED.                                                             BS100270
0028                ************************************************************************BS100280
0029                     EJCT                                                              BS100290
```

```
0030            EXT   SIT     ADDRESS OF HIGHEST AVAILABLE MEMORY          BS100300
0031      *                     LOCATION                                     BS100310
0032            EXT   PTB     ADDRESS OF LOWEST WORD IN USER SPACE         BS100320
0033            EXT   TYPE    OUTPUTS MESSAGE ON ASR                       BS100330
0034            EXT   IPUT    INPUTS A LINE FROM THE ASR                   BS100340
0035            EXT   GNSC    GETS NEXT NON-BLANK CHARACTER FROM THE       BS100350
0036      *                     INPUT BUFFER                                 BS100360
0037            EXT   LODF    FLAGS PAPER TAPE INPUT FROM ASR              BS100370
0038            EXT   LFCR    OUTPUTS CARRIAGE RETURN, LINE FEED ON ASR    BS100380
0039            EXT   JOB     JOB COMMAND PROCESSOR ENTRY                  BS100390
0040            EXT   FINT    INTEGER TO FLOATING POINT CONVERSION         BS100400
0041            EXT   DEFT    DAC TO ROUTINE TO FLAG ERROR OF              BS100410
0042            EXT   ATND    DAC TO ARCTANGENT FUNCTION                   BS100420
0043            EXT   TAND    DAC TO TANGENT FUNCTION                      BS100430
0044            EXT   SIND    DAC TO SINE FUNCTION                         BS100440
0045            EXT   COSD    DAC TO COSINE FUNCTION                       BS100450
0046            EXT   SQRD    DAC TO SQUARE ROOT FUNCTION                  BS100460
0047            EXT   SCVF    STORES A NUMBER INTO THE FLOATING POINT      BS100470
0048      *                     ACCUMULATOR                                  BS100480
0049            EXT   PCVF    PRINTS THE NUMBER IN THE FLOATING POINT      BS100490
0050      *                     ACCUMULATOR                                  BS100500
0051            EXT   C1                                                   BS100510
0052            EXT   C10                                                  BS100520
0053            EXT   C215                                                 BS100530
0054            EXT   C240                                                 BS100540
0055            EXT   C241                                                 BS100550
0056            EXT   C260                                                 BS100560
0057      *                                                                BS100570
0058      *                                                                BS100580
0059      *                                                                BS100590
0060            ENT   INIT                                                 BS100600
0061            ENT   CSRH    HIGH CROSS SECTOR REFERENCE GENERATOR        BS100610
0062      *                                                                BS100620
0063      *                                                                BS100630
0064      *                                                                BS100640
0065            REL                                                        BS100650
0066      *                                                                BS100660
```

```
0067                *                                                       BSI00670
0068                *                                                       BSI00680
0069                     EJCT                                               BSI00690
```

```
0070  *                                                                    BSI00700
0071  *                                                                    BSI00710
0072  *          INITIALIZATION ROUTINE                                    BSI00720
0073  *                                                                    BSI00730
0074  *     CALLING SEQUENCE:                                              BSI00740
0075  *                                                                    BSI00750
0076  *          JSI    INIT                                               BSI00760
0077  *          RETURN IS MADE THROUGH JOB COMMAND PROCESSOR               BSI00770
0078  *                                                                    BSI00780
0079  *                                                                    BSI00790
0080  *          THE PROGRAM TEXT TABLE BASE POINTER(PTB) IS INITIALIZED   BSI00800
0081  *     TO THE LOWEST ADDRESS OF THE INITIALIZATION ROUTINE, AND AN     BSI00810
0082  *     IDENTIFICATION MESSAGE IS PRINTED.  IF THE IOS WAS LOADED AFTER BSI00820
0083  *     MTHPAK, OR IF MTHPAK WAS LOADED AFTER THE INITIALIZATION ROUTINE, BSI00830
0084  *     AN ILLEGAL CONFIGURATION MESSAGE IS PRINTED, AND THE PROGRAM HALTS.BSI00840
0085  *     IF ANYTHING WAS LOADED BETWEEN MTHPAK AND THE INITIALIZATION    BSI00850
0086  *     ROUTINE, THERE IS A JUMP TO SET THE STATEMENT INDEX HIGH POINTER BSI00860
0087  *     (SII).  OTHERWISE THE USER IS ASKED IF HE WOULD LIKE TO DELETE ATN.BSI00870
0088  *     A NO ANSWER CAUSES A JUMP TO SET SII.  A YES ANSWER CAUSES PTB TO BSI00880
0089  *     BE SET TO THE FIRST WORD OF THE ROUTINE, AND THE POINTER TO THE  BSI00890
0090  *     ROUTINE IS REPLACED BY A POINTER TO A ROUTINE WHICH FLAGS A DE-  BSI00900
0091  *     LETED LIBRARY FUNCTION(DF) ERROR.  THIS SEQUENCE IS THEN REPEATED BSI00910
0092  *     FOR SIN, COS, AND TAN, AND SQR.                                BSI00920
0093  *          SII IS INITIALIZED TO THE HIGH CORE MEMORY ADDRESS, AND THE BSI00930
0094  *     USER IS ASKED IF IT IS ALL RIGHT TO USE ALL OF CORE.  A YES ANSWER BSI00940
0095  *     CAUSES A JUMP TO CALCULATE THE AMOUNT OF USER SPACE.  OTHERWISE THEBSI00950
0096  *     USER INPUT HIGH OCTAL ADDRESS IS FORMED.  NON-OCTAL NUMBERS, OVER- BSI00960
0097  *     FLOW, OR AN ADDRESS GREATER THAN THE HIGH CORE ADDRESS CAUSES A  BSI00970
0098  *     MESSAGE TO BE PRINTED REQUESTING A YES OR A HIGH OCTAL ADDRESS.  BSI00980
0099  *     OTHERWISE SII IS SET TO THE USER INPUT HIGH OCTAL ADDRESS.  THE  BSI00990
0100  *     AMOUNT OF USER SPACE IS CALCULATED BY SUBTRACTING PTB FROM SIT AND BSI01000
0101  *     ADDING ONE.  IF THE RESULT IS LESS THAN ELEVEN, A MESSAGE IS    BSI01010
0102  *     PRINTED TO FLAG INSUFFICIENT USER SPACE, AND THE ROUTINE LOOPS TO BSI01020
0103  *     RESET SII.  OTHERWISE THE AMOUNT OF USER SPACE IS PRINTED, THE CALL BSI01030
0104  *     TO THE INITIALIZATION ROUTINE IS REPLACED WITH A CRA INSTRUCTION, BSI01040
0105  *     AND THE ROUTINE EXITS THROUGH THE JOB COMMAND PROCESSOR.        BSI01050
0106  *                                                                    BSI01060
```

```
0107                        *                                                    BSI01070
0108 00000   0 000000   INIA DAC   *          ADDRESS OF FIRST WORD OF INITIALIZATION  BSI01080
0109                        *                  ROUTINE                           BSI01090
0110 00001   0 000000   INIT DAC   **         ENTRY                             BSI01100
0111 00002   0 02 00000      LDA   INIA       INITIALIZE LOW POINTER TO FIRST WORD  BSI01110
0112 00003   0 04 00000      STA   PTB        OF INITIALIZATION ROUTINE         BSI01120
0113 00004   0 04 00000      STA   LODF       SET FLAG FOR NO PAPER TAPE INPUT  BSI01130
0114 00005   0 10 00000      JST   CRLF       C/R, LINE FEED                    BSI01140
0115 00006   0 10 00000      JST   TYPE       OUTPUT ID MESSAGE                 BSI01150
0116 00007   0 000000        XAC   IDMS       X                                 BSI01160
0117 00010   0 10 00000      JST   CRLF       A COUPLE OF LINE FEEDS            BSI01170
0118 00011   0 10 00000      JST   CRLF       X                                 BSI01180
0119                        *                                                    BSI01190
0120                        *    CHECK FOR ILLEGAL CONFIGURATION                 BSI01200
0121                        *                                                    BSI01210
0122 00012   0 02 00205      LDA   HMPK       LOAD HIGH ADDRESS OF MTHPAK       BSI01220
0123 00013   0 07 00206      SUB   IOSA       SUBTRACT ADDRESS IN IOS           BSI01230
0124 00014   100400          SPL              SKIP IF POSITIVE                  BSI01240
0125 00015   0 01 00156      JMP   IN06       JUMP TO FLAG ILLEGAL CONFIGURATION IF  BSI01250
0126                        *                  NEGATIVE                          BSI01260
0127 00016   0 02 00205      LDA   HMPK       LOAD THE LAST ADDRESS OF THE MTHPAK  BSI01270
0128 00017   101100          SLN              SKIP IF THE ADDRESS IS ODD        BSI01280
0129 00020   141206          AOA              IF IT'S EVEN ADD ONE SINCE LOADING STARTS  BSI01290
0130                        *                  ON AN EVEN LOCATION               BSI01300
0131 00021   141206          AOA              ADD ONE AND                       BSI01310
0132 00022   0 11 00000      CAS   INIA       COMPARE WITH THE ADDRESS OF THE FIRST WORD  BSI01320
0133                        *                  OF THE INITIALIZATION ROUTINE     BSI01330
0134 00023   0 01 00156      JMP   IN06       IF THE MTHPAK WAS LOADED AFTER THE INITIAL-BSI01340
0135                        *                  IZATION ROUTINE THEN JUMP TO FLAG AN  BSI01350
0136                        *                  ILLEGAL CONFIGURATION             BSI01360
0137 00024   100000          SKP              IF MTHPAK WAS LOADED JUST BEFORE THE  BSI01370
0138                        *                  INITIALIZATION ROUTINE THEN JUMP TO  BSI01380
0139                        *                  ASK USER IF HE WISHES TO DELETE ANY OF  BSI01390
0140                        *                  LIBRARY FUNCTION ROUTINES         BSI01400
0141 00025   0 01 00047      JMP   IN01       IF ANYTHING WAS LOADED BETWEEN THE MTHPAK  BSI01410
0142                        *                  AND THE INITIALIZATION ROUTINE THEN  BSI01420
0143                        *                  JUMP TO SET THE HIGH POINTER (SIT)  BSI01430
```

```
0144                        *                                                                   BSI01440
0145                        *     SET LOW POINTER, PTB                                          BSI01450
0146                        *                                                                   BSI01460
0147 00026   0 02 00214 IN02 LDA   AIND   LOAD ADDRESS OF ATN MESSAGE                           BSI01470
0148 00027   0 10 00161       JSI   DFUA   DELETE ARCTANGENT FUCTION                           BSI01480
0149 00030   0 02 00000       LDA   DELI   YES-REPLACE POINTER TO ATN WITH A                   BSI01490
0150 00031   0 13 00000       IMA   AIND   POINTER TO ROUTINE TO FLAG ERROR OF                 BSI01500
0151 00032   0 04 00000       STA   PTB    AND SET LOW POINTER TO FIRST WORD OF ATN            BSI01510
0152 00033   0 02 00216       LDA   SCOD   LOAD ADDRESS OF SIN, COS, TAN MESSAGE              BSI01520
0153 00034   0 10 00161       JSI   DFUA   DELETE SIN, COS, TAN FUNCTIONS                     BSI01530
0154 00035   0 02 00000       LDA   DELI   YES- REPLACE POINTER TO THOSE FUNCTIONS            BSI01540
0155 00036   0 04 00000       STA   TAND   WITH POINTER TO FLAG ERROR OF                      BSI01550
0156 00037   0 04 00000       STA   SIND   X                                                  BSI01560
0157 00040   0 13 00000       IMA   COSD   X                                                  BSI01570
0158 00041   0 04 00000       STA   PTB    SET LOW POINTER TO FIRST WORD OF COS               BSI01580
0159 00042   0 02 00215       LDA   SQOD   LOAD ADDRESS OF SQUARE ROOT MESSAGE               BSI01590
0160 00043   0 10 00161       JSI   DFUA   DELETE SQUARE ROOT FUNCTION                        BSI01600
0161 00044   0 02 00000       LDA   DELI   YES-REPLACE POINTER TO SQUARE ROOT FUNCTIONBSI01610
0162 00045   0 13 00000       IMA   SQRD   WITH POINTER TO ROUTINE TO FLAG ERROR OF          BSI01620
0163 00046   0 04 00000       STA   PTB    AND SET LOW POINTER TO FIRST WORD OF SQR           BSI01630
0164                        *                                                                   BSI01640
0165                        *     NOW SET HIGH POINTER, SIT                                     BSI01650
0166                        *                                                                   BSI01660
0167 00047   0 02 00211 IN01 LDA   C16K   LOAD HIGHEST POSSIBLE ADDRESS (16K)                  BSI01670
0168 00050   0 04 00000 IN10 STA   SIT    STORE ADDRESS IN HIGH POINTER                      BSI01680
0169 00051  -0 13 00000       IMA*  SIT    CHECK IF ADDRESS EXISTS BY STORING,                BSI01690
0170 00052  -0 13 00000       IMA*  SIT    THEN LOADING (BUT DON'T DESTROY CONTENTS OFBSI01700
0171 00053   0 05 00000       ERA   SIT    ADDRESS), AND COMPARING                            BSI01710
0172 00054   101040           SNZ          TEST IF THE SAME                                   BSI01720
0173 00055   0 01 00060       JMP   *+3    YES-HIGH MEMORY ADDRESS IS IN SIT                  BSI01730
0174 00056   0 07 00210       SUB   C4K    NO-SUBTRACT 4K                                     BSI01740
0175 00057   0 01 00050       JMP   IN10   AND TRY AGAIN                                      BSI01750
0176 00060   0 10 00000       JSI   TYPE   ASK USER IF HE WOULD LIKE TO USE ALL OF            BSI01760
0177 00061   0 000000         XAC   HMAM   CORE                                               BSI01770
0178 00062   0 10 00000 IN03 JSI   TYPE   REQUEST A YES OR HIGH OCTAL ADDRESS                 BSI01780
0179 00063   0 000000         XAC   AYOH   X                                                  BSI01790
0180 00064   0 10 00000       JSI   CRLF   C/R, LINE FEED                                     BSI01800
```

```
0181 00065   0 02 00000       LDA   C241          OUTPUT EXCLAMATION POINT                    BSI01810
0182 00066   0 10 00000       JST   INPT          THEN INPUT LINE FROM ASR                    BSI01820
0183 00067   0 000000        XAC   SBUF          INPUT BUFFER                                BSI01830
0184 00070   0 10 00000       JST   GNBC          GET THE FIRST NON-BLANK CHARACTER           BSI01840
0185 00071   0 11 00213       CAS   C331          IS IT A 'Y'                                 BSI01850
0186 00072   0 01 00062       JMP   IN03          NOT Y OR NUMBER-REQUEST YES OR HIGH         BSI01860
0187                    *                         OCTAL ADDRESS                               BSI01870
0188 00073   0 01 00127       JMP   IN04          YES-ASSUME YES ANSWER-JUMP TO CALCULATE     BSI01880
0189                    *                         USER SPACE                                  BSI01890
0190 00074   000201           IAB                 SAVE THE CHARACTER                          BSI01900
0191 00075   140040      CRA  CRA                                                             BSI01910
0192 00076   0 04 00207       STA   HOA           INITIALIZE HIGH OCTAL ADDRESS TO ZERO       BSI01920
0193 00077   000201           IAB                 AND CLEAR B REGISTER                        BSI01930
0194                    *                                                                    BSI01940
0195                    *     HERE TO INPUT USER HIGH OCTAL ADDRESS                          BSI01950
0196                    *                                                                    BSI01960
0197 00100   0 07 00000  IN05 SUB   C260          SUBTRACT '260 TO GET PURE NUMBER            BSI01970
0198 00101   100400           SPL                 MAKE SURE IT'S NOT LESS THAN ZERO           BSI01980
0199 00102   0 01 00062       JMP   IN03          IF LESS, REQUEST YES OR HIGH ADDRESS        BSI01990
0200 00103   0 11 00000       CAS   C10           OR GREATER THAN 7                           BSI02000
0201 00104   101000           NOP                 X                                           BSI02010
0202 00105   0 01 00062       JMP   IN03          IF GREATER, THEN REQUEST YES OR HIGH OCTAL  BSI02020
0203                    *                         ADDRESS                                     BSI02030
0204 00106   0 13 00207       IMA   HOA           MAKE ROOM FOR THE DIGIT                     BSI02040
0205 00107   0412 75          LLR   3             BY SHIFTING ADDRESS THREE PLACES LEFT       BSI02050
0206 00110   0 05 00207       ERA   HOA           ADD THE DIGIT                               BSI02060
0207 00111   000201           IAB                 MAKE SURE THERE IS NO OVERFLOW BY TESTING   BSI02070
0208 00112   100040           SZE                 THE B REGISTER                              BSI02080
0209 00113   0 01 00062       JMP   IN03          OVERFLOW-REQUEST YES OR HIGH OCTAL ADDRESS  BSI02090
0210 00114   000201           IAB                 REPOSITION                                  BSI02100
0211 00115   0 04 00207       STA   HOA           SAVE ADDRESS                                BSI02110
0212 00116   0 10 00000       JST   GNBC          GET NEXT NON-BLANK CHARACTER                BSI02120
0213 00117   0 11 00000       CAS   C215          CHECK FOR C/R                               BSI02130
0214 00120   0 01 00100       JMP   IN05          NO-LOOP TO ADD DIGIT TO HIGH ADDRESS        BSI02140
0215 00121   0 02 00207       LDA   HOA           YES-LOAD USER HIGH INPUT ADDRESS            BSI02150
0216 00122   101400           SMI                 SKIP IF USER INPUT HIGH ADDRESS IS NEGATIVEBSI02160
0217 00123   0 11 00000       CAS   SII           COMPARE WITH HIGH CORE ADDRESS              BSI02170
```

```
0218 00124   0 01 00062       JMP   IN03          USER INPUT HIGH ADDRESS IS TOO LARGE OR    BSI02180
0219                    *                        NEGATIVE-JUMP TO INPUT AGAIN               BSI02190
0220 00125   101000          NOP                  X                                          BSI02200
0221 00126   0 04 00000       STA   SIT           OK-SAVE AS HIGH POINTER                    BSI02210
0222                    *                                                                   BSI02220
0223                    *    CALCULATE AND PRINT TOTAL USER SPACE                           BSI02230
0224                    *                                                                   BSI02240
0225 00127   0 02 00000 IN04 LDA   SIT           LOAD HIGH POINTER                          BSI02250
0226 00130   0 07 00000       SUB   PTR           SUBTRACT LOW POINTER                       BSI02260
0227 00131   0 06 00000       ADD   C1            ADD ONE                                    BSI02270
0228 00132   0 11 00000       CAS   C10           MAKE SURE IT ISN'T TOO SMALL               BSI02280
0229 00133   0 01 00141       JMP   **6           OK-JUMP TO PRINT USER SPACE MESSAGE        BSI02290
0230 00134   101000          NOP                  X                                          BSI02300
0231 00135   0 10 00000       JST   TYPE          TOO SMALL-FLAG INSUFFICIENT USER SPACE     BSI02310
0232 00136   0 000000         XAC   ISSM          X                                          BSI02320
0233 00137   0 10 00000       JST   LFCR          C/R, LINE FEED                             BSI02330
0234 00140   0 01 00047       JMP   IN01          LOOP TO SET HIGH POINTER                   BSI02340
0235 00141   0 10 00000       JST   FINT          FLOAT THE NUMBER                           BSI02350
0236 00142   0 10 00000       JST   SCVL          STORE IT INTO FLOATING POINT ACCUMULATOR   BSI02360
0237 00143   0 02 00000       LDA   C240          SUPPRESS BLANKS                            BSI02370
0238 00144   0 10 00000       JST   PCVL          PRINT THE NUMBER                           BSI02380
0239 00145   0 10 00000       JST   TYPE          PRINT USER SPACE MESSAGE                   BSI02390
0240 00146   0 000000         XAC   USPM          X                                          BSI02400
0241 00147   0 10 00000       JST   LFCR          C/R, LINE FEED                             BSI02410
0242                    *                                                                   BSI02420
0243                    *    CHANGE CALL TO THE INITIALIZATION ROUTINE TO A CRA             BSI02430
0244                    *                                                                   BSI02440
0245 00150   0 02 00001       LDA   INIT          LOAD ADDRESS PLUS ONE OF CALL              BSI02450
0246 00151   0 07 00000       SUB   C1            SUBTRACT ONE                               BSI02460
0247 00152   0 04 00001       STA   INIT          AND SAVE                                   BSI02470
0248 00153   0 02 00075       LDA   CRA           LOAD CRA                                   BSI02480
0249 00154  -0 04 00001       STA*  INIT          STORE IN LOCATION OF CALL                  BSI02490
0250 00155   0 01 00000       JMP   JOB           EXIT THROUGH JOB COMMAND PROCESSOR         BSI02500
0251                    *                                                                   BSI02510
0252                    *    HERE TO FLAG AN ILLEGAL CONFIGURATION                          BSI02520
0253                    *                                                                   BSI02530
0254 00156   0 10 00000 IN06 JST   TYPE          PRINT MESSAGE TO RELOAD                    BSI02540
```

```
0255 00157   0 000217        DAC    ILCM                X                          BS102550
0256 00160   000000          HLT                        AND HALT                   BS102560
0257                       *                                                      BS102570
0258                          EJCT                                                BS102580
```

```
0259                        *                                                                   BSI02590
0260                        *                                                                   BSI02600
0261                        *          DELETE LIBRARY FUNCTION ROUTINE                          BSI02610
0262                        *                                                                   BSI02620
0263                        *    CALLING SEQUENCE:                                              BSI02630
0264                        *                                                                   BSI02640
0265                        *          JSI    DFUA          ADDRESS OF MESSAGE CONTAINING FUNCTION  BSI02650
0266                        *                               NAME(S) IN THE A REGISTER           BSI02660
0267                        *          ......RETURN         IF YES ANSWER                       BSI02670
0268                        *                                                                   BSI02680
0269                        *                                                                   BSI02690
0270                        *          A MESSAGE IS PRINTED ASKING THE USER IF HE WOULD LIKE TO BSI02700
0271                        *    DELETE THE LIBRARY FUNCTION(S) WHOSE NAME(S) IS CONTAINED IN THE BSI02710
0272                        *    MESSAGE.  A LINE IS INPUTED FROM THE ASR, AND THE FIRST NON-BLANK BSI02720
0273                        *    CHARACTER IS TESTED.  IF IT IS A Y, RETURN IS MADE.  IF IT IS A N, BSI02730
0274                        *    THE ROUTINE JUMPS INTO INIT TO SET SII.  IF THE FIRST NON-BLANK BSI02740
0275                        *    CHARACTER IS OTHER THAN A Y OR A N, A MESSAGE IS PRINTED REQUESTINGBSI02750
0276                        *    A YES OR A NO ANSWER.                                          BSI02760
0277                        *                                                                   BSI02770
0278                        *                                                                   BSI02780
0279 00161   0 000000   DFUA DAC   **                 ENTRY                                    BSI02790
0280 00162   0 04 00166      STA   DFID               STORE FUNCTION NAME MESSAGE              BSI02800
0281 00163   0 10 00000      JSI   TYPE               OUTPUT QUESTION PREFIX                   BSI02810
0282 00164   0 000000        XAC   DFQ                X                                        BSI02820
0283 00165   0 10 00000      JSI   TYPE               PRINT FUNCTION NAME                      BSI02830
0284 00166   0 000000   DFID DAC   **                 X                                        BSI02840
0285 00167   0 10 00000      JSI   LFCR               OUTPUT C/R, LINE FEED                    BSI02850
0286 00170   0 02 00000 DF02 LDA   C241               OUTPUT AN EXCAMATION POINT               BSI02860
0287 00171   0 10 00000      JSI   IPUT               INPUT FROM THE ASR UNTIL A C/R           BSI02870
0288 00172   0 000000        XAC   SBUF               INPUT BUFFER                             BSI02880
0289 00173   0 10 00000      JSI   GNBC               GET THE FIRST NON-BLANK CHARACTER        BSI02890
0290 00174   0 11 00213      CAS   C331               IS IT A 'Y'                              BSI02900
0291 00175   0 01 00202      JMP   DF01               NO-REQUEST A YES OR A NO                 BSI02910
0292 00176  -0 01 00161      JMP*  DFUA               Y-ASSUME YES ANSWER-EXIT TO RESET LOW    BSI02920
0293                        *                        POINTER                                  BSI02930
0294 00177   0 11 00212      CAS   C316               IS IT A 'N'                              BSI02940
0295 00200   100000           SKP                    NO-REQUEST A YES OR NO ANSWER            BSI02950
```

```
0296 00201    0 01 00047       JMP    INO1      N-ASSUME NO ANSWER-JUMP TO SET HIGH POINTERBSI02960
0297 00202    0 10 00000 DF01  JSI    TYPE      NO-REQUEST A YES OR NO ANSWER              BSI02970
0298 00203    0 000000         XAC    AYON      X                                          BSI02980
0299 00204    0 01 00170       JMP    DF02      LOOP TO INPUT ANSWER                       BSI02990
0300                           EJCT                                                        BSI03000
```

```
0301                        *                                                                    BSI03010
0302                        *                                                                    BSI03020
0303                        *                                                                    BSI03030
0304 00205   0 000000   HMPK XAC   PP12          ADDRESS OF LAST WORD OF MTKPAK                  BSI03040
0305 00206   0 000000   IOSA XAC   IPOI          ADDRESS IN IOS PACKAGE                          BSI03050
0306 00207   000000     HOA  BSZ   1                                                            BSI03060
0307 00210   010000     C4K  OCI   10000                                                        BSI03070
0308 00211   037777     C16K OCI   37777                                                        BSI03080
0309 00212   000316     C316 OCI   316           ASCII N                                         BSI03090
0310 00213   000331     C331 OCI   331           ASCII Y                                         BSI03100
0311 00214   0 000000   AIQD XAC   AING          ADDRESS OF AIN NAME MESSAGE                     BSI03110
0312 00215   0 000000   SQQD XAC   SQRQ          ADDRESS OF SIN, COS, AND TAN NAME MESSAGE       BSI03120
0313 00216   0 000000   SCQD XAC   SCIQ          ADDRESS OF SQR NAME MESSAGE                     BSI03130
0314 00217   144714     ILCM BCI   14,ILLEGAL CONFIGURATION-RELOAD                               BSI03140
     00220   146305
     00221   143701
     00222   146240
     00223   141711
     00224   147306
     00225   144701
     00226   152722
     00227   140724
     00230   144711
     00231   147255
     00232   151305
     00233   146311
     00234   140704
0315 00235   000000          OCI   0                                                            BSI03150
0316         000236     CSRH EQU   *             USING LDR-APM, THIS WILL GENERATE THE LAST      BSI03160
0317                    *                        CROSS SECTOR REFERENCE IN SECTOR ZERO          BSI03170
0318                    *                                                                       BSI03180
0319                    *                                                                       BSI03190
0320                    *                                                                       BSI03200
0321                         END                                                                BSI03210

AING  000000E   AIQD  000214   C1    000000E   C10   000000E
C16K  000211    C215  000000E  C240  000000E   C241  000000E
```

C260	000000E	C316	000212	C331	000213	C4K	000210
COSD	000000E	CRA	000075	CSR	000236	DEL1	000000E
DF01	000202	DF02	000170	DF1D	000166	DFDA	000161
FINI	000000E	GNBC	000000E	HMPK	000205	HOA	000207
ILCM	000217	IN01	000047	IN02	000026	IN03	000062
IN04	000127	IN05	000100	IN06	000156	IN10	000050
IN11	000001	INIA	000000	IOSA	000206	IP01	000000E
JOB	000000E	LFCR	000000E	LODF	000000E	PCVL	000000E
PIB	000000E	SCQD	000216	SCVL	000000E	SIND	000000E
SII	000000E	SQQD	000215	SQRD	000000E	TAND	000000E
TYPE	000000E						

0000 WARNING OR ERROR FLAGS

DAP-16 MOD 2 REV. D 06-28-71