```
 OPTIONS NODECK,LIST,XREF,NOREL,OBJ(P)
THE LIST OF OPTIONS USED DURING THIS ASSEMBLY IS--   NODECK,LIST,XREF,NOREL,OBJ
```

```
                                 EXTERNAL SYMBOL LIST
SYMBOL   TYPE                                                  VER 15, MOD 00  05/08/20  PAGE   1
#LOADR   MODULE
```

```
    0000                          1 #LOADR START 0
                                  2        PRINT ON,NODATA
                                  3 *      @SYS  EXP-N
                                214+       PRINT ON
                                215 *      @SPF  EXP-N
                                678+       PRINT ON
                                679 *      @FXD  EXP-N
                               1084+       PRINT ON
                               1085 *      @B@E  EXP-N
                               1985+       PRINT ON
                               1986 *      @ERM  EXP-N
                               2608+       PRINT ON
                               2609 *      @VMD  EXP-N
                               2730+       PRINT ON
                               2731 *      $V$E  EXP-N
                               3153+       PRINT ON
                               3154 *      @WKA  EXP-N
                               3224+       PRINT ON
                          00A0 3225 $$$NLN EQU   X'A0'                     TEMP HJS 2020
```

```
                               3227 ***********************************************************************
                               3228 * 5703-XM1   COPYRIGHT IBM CORP 1970                                  *
                               3229 *            REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083      *
                               3230 *                                                                     *
                               3231 ***********************************************************************
                               3232 *STATUS -                                                             *
                               3233 *   VERSION 1 MODIFICATION 0                                          *
                               3234 *                                                                     *
                               3235 *FUNCTION -                                                           *
                               3236 *   * LALLOC ALLOCATES VIRTUAL MEMORY SPACE FOR ARITHMETIC AND        *
                               3237 *     CHARACTER ARRAYS                                                *
                               3238 *   * THE ARRAY DOPE VECTOR IMAGES IN THE FUNCTION AND ARRAY TABLE    *
                               3239 *     ARE COMPLETED AS THE ARRAYS ARE ALLOCATED                       *
                               3240 *   * VIRTUAL MEMORY IS INITIALIZED WITH THE INTERPRETER VIRTUAL      *
                               3241 *     MEMORY FUNCTIONS                                                *
                               3242 *                                                                     *
                               3243 *ENTRY POINTS -                                                       *
                               3244 *   * ENTRY POINT - LALLOC, FOR ARRAY ALLOCATION                      *
                               3245 *     THE CALLING SEQUENCE IS:                                        *
                               3246 *        B     $BLOAD                                                 *
                               3247 *        DC    AL2'DPL'                                               *
                               3248 *     WHERE DPL IS THE PARAMETER LIST FOR GET THE LOADER.             *
                               3249 *   * ENTRY POINT - LAL000, FOR VIRT MEMORY FUNCTION INITIALIZATION.  *
                               3250 *     THE CALLING SEQUENCE IS:                                        *
                               3251 *        B     LAL000                                                 *
                               3252 *                                                                     *
                               3253 *INPUT -                                                              *
                               3254 *   * LALVA1 - 2 BYTES, FOR THE FIRST FREE VIRTUAL ADDRESS IN VIRTUAL *
                               3255 *     MEMORY REGION 1 (END OF PMC)                                    *
                               3256 *   * LALVA2 - 2 BYTES, FOR THE FIRST NON-FREE VIRTUAL ADDRESS TW     *
                               3257 *     VIRTUAL MEMORY REGION 1 (START OF CONSTANTS)                    *
                               3258 *   * LALVA3 - 2 BYTES, FOR THE FIRST FREE VIRTUAL ADDRESS IN VIRTUAL *
                               3259 *     MEMORY  REGION 2 (END OF VARIABLES)                             *
                               3260 *   * LALVA4 - 2 BYTES, FOR THE FIRST NON-FREE VIRTUAL ADDRESS IN     *
                               3261 *     VIRTUAL MEMORY REGION 2 (START OF FUNCTION AND ARRAY TABLE)     *
                               3262 *   * ARITHMETIC ARRAY SYMBOL TABLE - 58 BYTES, 29 2-BYTE ENTRIES     *
                               3263 *       * CONTAINS A VIRTUAL ADDRESS IF SYMBOL WAS REFERENCED         *
                               3264 *       * CONTAINS ZEROS IF SYMBOL WAS NOT REFERENCED                 *
                               3265 *   * CHARACTER ARRAY SYMBOL TABLE - 58 BYTES, 29 2-BYTE ENTRIES      *
                               3266 *       * CONTAINS A VIRTUAL ADDRESS IF SYMBOL WAS REFERENCED         *
                               3267 *       * CONTAINS ZEROS IF SYMBOL WAS NOT REFERENCED                 *
                               3268 *   * FUNCTION AND ARRAY TABLE - 406 BYTES, CONTAINS:                 *
                               3269 *       * ARRAY DOPE VECTOR IMAGES                                    *
                               3270 *          * 29 8-BYTE ARITHMETIC ARRAY DOPE VECTOR ENTRIES           *
                               3271 *          * 29 4-BYTE CHARACTER ARRAY DOPE VECTOR ENTRIES            *
                               3272 *   * VIRTUAL MEMORY FUNCTION ROUTINES, IN PRECISION REQUIRED         *
                               3273 *                                                                     *
                               3274 *OUTPUT -                                                             *
                               3275 *   * ARRAY DOPE VECTOR IMAGES, THE REFERENCED DOPE VECTORS HAVE      *
                               3276 *     BEEN COMPLETED                                                  *
                               3277 *       * DIMENSIONS (1 ONLY IF CHARACTER)                            *
                               3278 *       * MAXIMUM SIZE                                                *
                               3279 *       * BASE ADDRESS                                                *
                               3280 *   * VIRTUAL MEMORY REGION POINTERS 1 - 4, UPDATED TO REFLECT THE    *
                               3281 *     ALLOCATED ARRAYS                                                *
                               3282 *   * VIRTUAL MEMORY FUNCTION ROUTINES, IN PRECISION REQUIRED         *
                               3283 *                                                                     *
                               3284 *EXTERNAL REFERENCES -                                                *
                               3285 *        $XIND1 - SYSTEM EXECUTION INDICATOR 1                        *
                               3286 *        $XIND3 - SYSTEM EXECUTION INDICATOR 3                        *
                               3287 *        DL2ICS - 2 - TRACK LIOCS                                     *
                               3288 *        DL2RAD - DL2ICS BASE PARAMETER                               *
                               3289 *        DL4ICS - 4 - TRACK LIOCS                                     *
                               3290 *        $DISKN - SYSTEM DISK IOCR                                    *
                               3291 *        $RLOAD - SYSTEM LOADER ENTRY                                 *
                               3292 *        $CAERK - SYSTEM ERROR MESSAGE RMINE                          *
                               3293 *        5CAERR - ERROR CODE INDICATOR PARAMETER                      *
                               3294 *        LDFILE - FILE BUFFER ALLOCATJON                              *
                               3295 *                                                                     *
                               3296 *EXITS, NORMAL -                                                      *
                               3297 *   LALLOC HAS TWO NORMAL EXITS                                       *
                               3298 *        LDFILE - AFTER ARRAY ALLOCATION                              *
                               3299 *        $RLOAD - AFTER VM FUNCTION INITIALIZATION                    *
                               3300 *                                                                     *
                               3301 *EXITS, ERROR -                                                       *
                               3302 *        $CAERK - WITH ERROR CODE @@E611, TOO MANY ARRAY ELEMENTS     *
                               3303 *                                                                     *
                               3304 *TABLESNORK AREAS -                                                   *
                               3305 *   * THE CONSTANTS AND WORK AREAS RESIDE AT THE END OF THE           *
                               3306 *     EXECUTABLE CODE AND ARE REFERENCED BY @BR                       *
                               3307 *   * EXECUTION LOADER PARAMETER AREA, LOCATED AT CORE ADDRESS        *
                               3308 *     1A00 TO 1E00 AND CONTAINS                                       *
                               3309 *      * FOUR VIRTUAL MEMORY REGION POINTERS (SEE INPUT)              *
                               3310 *      * FIVE VARIABLE SYMBOL TABLES                                  *
                               3311 *      * FUNCTION AND ARRAY TABLES                                    *
                               3312 *                                                                     *
                               3313 *ATTRIBUTES -                                                         *
                               3314 *   LALLOC IS REUSABLE                                                *
                               3315 *                                                                     *
                               3316 *CHARACTER CODE DEPENDENCY                                            *
                               3317 *   N/A                                                               *
                               3318 *                                                                     *
                               3319 *NOTES -                                                              *
                               3320 *   ERROR PROCEDURES                                                  *
                               3321 *      * ERROR CODE IS SET AT $CAERR                                  *
                               3322 *      * $ERRPG IS SET WITH $$$LNL TO OMIT THE LINE NUMBER            *
                               3323 *                                                                     *
                               3324 *   REGISTER USAGE                                                    *
                               3325 *      * BOTH REGISTERS ARE USED DURING EXECUTION                     *
                               3326 *      * THE REGISTERS ARE NOT SAVED OR RESTORED                      *
                               3327 *                                                                     *
                               3328 *   SAVED/RESTORED AREAS                                              *
                               3329 *      N/A                                                            *
                               3330 *                                                                     *
                               3331 *   MODIFICATION CONSIDERATIONS                                       *
                               3332 *      N/A                                                            *
                               3333 *                                                                     *
                               3334 *   REQUIRED MODULES                                                  *
                               3335 *        @SYSEQ - COMMON SYSTEM EQUATES                               *
                               3336 *        @FYDEQ - SYSTEM NUCLEUS ADDRESSES AND INDICATORS             *
                               3337 *        @CANEQ - SYSTEM LOCATION EQUATES                             *
                               3338 *        @WKAEQ - SYS WORK AREA DADDR EQUATES                         *
                               3339 *        @VMDEQ - VM DIRECTORY EQUATES                                *
                               3340 *        @SPFEQ - SYSTEM PROG FILE EQUATES                            *
                               3341 *        @ERMEQ - GENERAL ERROR MESSAGE EQUATES                       *
                               3342 *        $B$EQU - COMPILER FIXED EQUATES                              *
                               3343 *        $B@EOU - COMPILER SYSTEM EQUATES                             *
                               3344 *        LDFILE - LOADER BUFFER ALLOCATION                            *
                               3345 *        DL2ICS - 2 TRACK LIOCS                                       *
                               3346 *        DL4ICS - 4 TRACK LIOCS                                       *
                               3347 *                                                                     *
                               3348 *   OTHER                                                             *
                               3349 *      LALLOC AT ENTRY POINT LAL000 WILL OVERLAY CORE PAGES 0700      *
                               3350 *      TO 1600 (15 PAGES) DURING VM FUNCTION INITIALIZATION           *
                               3351 *      (INCLUDES PORTION OF LALLOC)                                   *
                               3352 ***********************************************************************

                               3354 *       HDR   #LOADR
                               3355 ***********************************************************************
                               3356 *  PROGRAM HEADER FOR DISK LOAD
                               3357 ***********************************************************************
                               3358 *#$LOAD EQU   X'0100'                   DISK ADDR OF #LOADR
                               3359 *#$$LOA EQU   X'0600'                   CORE LOAD ADDRESS OF #LOADR
                               3360 *#$@LOA EQU   019                       SECTOR CNT OF #LOADR
    0600                       3361        ORG   #$$LOA                    CORE LOAD ADDRESS
                          0600 3362 $$$$$$ EQU   *                         FIRST LOCATION IN PROGRAM
    0600 7BD3D6C1C4D9     0605 3363        DC    CL6'#LOADR'               PROGRAM NAME
    0606 05               0606 3364        DC    IL1'005'                  PROGRAM NUMBER OF #LOADR
                          0607 3365 #LOAD  EQU   *                         ENTRY POINT TO PROGRRAM
                               3366 *** END OF EXPANSION ***

                               3368 ***********************************************************************
                               3369 ***********************************************************************
                               3370 *                                                                     *
                               3371 *      INITIAL EXECUTION LOADER ENTRY                                 *
                               3372 *                                                                     *
                               3373 ***********************************************************************
                               3374 ***********************************************************************
                               3375 *
                          0607 3376 LALLOC EQU   *                         LALLOC ENTRY POINT
    0607 C0 87 069F            3377        B     LAL100                    ALLOCATE ARRAY SPACE
```

```
                               3379 ***********************************************************************
                               3380 ***********************************************************************
                               3381 *                                                                     *
                               3382 *      ALLOC PHASE 2 - VIRTUAL MEMORY FUNCTION INITIALIZATION         *
                               3383 *                                                                     *
                               3384 ***********************************************************************
                               3385 ***********************************************************************
                               3386 *
                               3387 ***********************************************************************
                               3388 * LOAD APPROPRIATE VM FUNCTIONS DEPENDING ON THE PROGRAM PRECISION
                               3389 ***********************************************************************
                               3390 *
                          060B 3391 LAL000 EQU   *                         LALLOC ENTRY POINT TWO
                          060B 3392        USING LAL000,@BR                SET BASE ADDR
    060B C2 01 060B            3393        LA    LAL000,@BR                LOAD LALLOC BASE
                               3394 *
                               3395 * DETERMINE PRECISION
                               3396 *
    060F 38 80 03D8            3397        TBN   $XIND3,$VMDEF             VM FUNCTIONS IN VM ?
    0613 F2 90 1B              3398        JF    LAL020                    NO, TEST PROGRAM PREC
    0616 38 40 03D8            3399        TBN   $XIND3,LALPMK             IS PRESENT VM RUNCTIONS LONG ?
    061A F2 90 0A              3400        JF    LAL010                    SHORT, IS PRESENT PREC SHORT
    061D 38 40 03D0            3401        TBN   $XIND1,LALPMK             IS PRESENT PROG PREC SHORT ?
    0621 F2 90 14              3402        JF    LAL030                    SHORT, READ SHORT PREC FUNC
    0624 F2 87 5A              3403        J     LAL060                    TO TNTERPETER RTN
    0627 38 40 03D0            3404 LAL010 TBN   $XIND1,LALPMK             IS PRESENT PROG PREC SHORT ?
    062B F2 10 12              3405        JT    LAL040                    LONG, READ LONG PREC VM FUNC
    062E F2 87 50              3406        J     LAL060                    SHORT, EXIT TO INTERPRETER RTN
    0631 38 40 03D0            3407 LAL020 TBN   $XIND1,LALPMK             IS PRESENT PROG, PREC SHORT ?
    0635 F2 10 08              3408        JT    LAL040                    LONG, READ LONG PREC VM FUNC
                               3409 *
                               3410 * SET PROGRAM TO PLACE THE SHORT PRECISION FUNCTION SECTORS INTO VM
                               3411 *
    0638 1C 01 17E6 7E         3412 LAL030 MVC   DL2RAD,LALSFA(LAL2BY,@BR) SET SHORT PREC SECTOR DISP
    063D F2 87 05              3413        J     LAL050                    PREFORM I/O
                               3414 *
                               3415 * SET PROGRAM TO PLACE THE LONG PRECISION FUNCTION SECTORS INTO VM
                               3416 *
    0640 1C 01 17E6 80         3417 LAL040 MVC   DL2RAD,LALLFA(LAL2BY,@BR) SET LONG PREC SECTOR DISP
                               3418 *
                               3419 * READ FUNCTIONS FROM DISK AND WRITE THEM TO VIRTUAL MEMORY
                               3420 *
    0645 0E 01 17E6 0587       3421 LAL050 ALC   DL2RAD,$BSADR(LAL2BY)     SET SYS RELOCATION FACTOR
    064B C0 87 174E            3422 LAL055 B     DL2ICS                    DISK IOCR RTN
    064F 068D             0650 3423        DC    AL(@CADDR)(LALRFL)        ADDR DISK PARAM LIST
    0651 C0 87 17E7            3424        B     DL4ICS                    DISK IOCR RTN
    0655 0693             0656 3425        DC    AL(@CADDR)(LALWFL)        ADDR DISK PARAM LIST
    0657 5E 00 84 85           3426        ALC   LALFDA(1,@BR),LALSCT(,@BR)  INCR GET DPL DISP
    065B 5E 00 8A 85           3427        ALC   LALSDS(1,@BR),LALSCT(,@BR)  INCR PUT DPL DISP
    065F 5F 00 81 7C           3428        SLC   LALCTR(,@BR),LALX01(LALB01,@BR)  WAIT CODE
    0663 C0 84 064B            3429        BH    LAL055
    0667 3C 05 0690            3430        MVI   LALRFL+@DCNT,LALSC5       SET UP TO WRITE LAST 5 SECT 1-4
    066B 3C 05 0696            3431        MVI   LALWFL+@DCNT,LALSC5       SET UP TO WRITE LAST 5 SECT 1-4
    066F C0 87 174E            3432        B     DL2ICS                    READ LAST 5 SECTORS
    0673 068D             0674 3433        DC    AL(@CADDR)(LALRFL)
    0675 C0 87 17E7            3434        B     DL4ICS                    WRITE LAST 5 SECTORS
    0679 0693             067A 3435        DC    AL(@CADDR)(LALWFL)
    067B C0 87 0025            3436        B     $DISKN                    WAIT FOR I/O COMPLETION
    067F 057F             0680 3437        DC    AL(@CADDR)($WAITF)        WAIT PARAM

                               3439 ***********************************************************************
                               3440 * LOAD INTERPRETER FOR PSEUDO CODE EXECUTION
                               3441 ***********************************************************************
    0681 C0 87 051E            3442 LAL060 B     $RLOAD                    LOAD INTERPRETER AND EXIT
    0685 0699             0686 3443        DC    AL2(LALPLI)               INPUT DISK PARAM LIST ADDR

                               3445 ***********************************************************************
                               3446 * LALLOC PHASE TWO - CONSTANTS, WORK AREAS AND EQUATES
                               3447 ***********************************************************************
                               3448 *
                               3449 * LALLOC EQUATES REFERENCING CONSTANTS
                               3450 *
                          0005 3451 LALSC5 EQU   5                         SECTOR CNT LAST 5 SECTORS
                               3452 *
                          0000 3453 LALB00 EQU   0                         CHECK FOR NULL INDICATOR
                          0001 3454 LALB01 EQU   1                         BYTES IN COUNTER
                          0003 3455 LALX03 EQU   3                         TO SET WRITE COUNTER
                          0002 3456 LAL2BY EQU   2                         BYTES IN RELOCATION FACTOR
                          0002 3457 LALSDP EQU   2                         FUNCTION DISP FROM VM START
                          0040 3458 LALPMK EQU   $XPREC                    PRECISION MASK TEST
                          0000 3459 LALSRF EQU   *-*                       SYSTEM RELOCATION FACTOR
                          0700 3460 LALOVR EQU   X'0700'                   LOADER OVERLAY CADDR START
                               3461 *
                               3462 * LALLOC CONSTANTS
                               3463 *
    0687 01               0687 3464 LALX01 DC    XL1'01'                   TO DECR COUNTER
    0688 0D00             0689 3465 LALSFA DC    AL2(#$FMST)               DISK ADDR SHORT PREC FUNC
    068A 1E00             068B 3466 LALLFA DC    AL2(#$FMLN)               DISK ADDR LONG PREC FUNC
                               3467 *
                               3468 * LALLOC WORK AREAS
                               3469 *
    068C                  068C 3470 LALCTR DS    CL1                       WRITE COUNTER
    068C                       3471        ORG   LALCTR                    * INITIALLY SET TO CONTAIN
    068C 06               068C 3472        DC    XL1'06'                   * FIVE
                               3473 *
                               3474 * DISK PARAMETER LIST
                               3475 *
                          068D 3476 LALRFL EQU   *                         ADDR DISK PARM LIST
    068D 01               068D 3477        DC    AL1(@DGET)                READ CODE
    068E                  068E 3478 LALCYL DS    CL1                       BASE CYL
    068F                  068F 3479 LALFDA DS    CL1                       DISP FROM BASE CYL
    068E                       3480        ORG   LALCYL                    * BASE AND DISP BOTH INITIALLY
    068E 0000             068F 3481        DC    XL2'00'                   * SET TO ZERO
    0690 0D               0690 3482 LALSCT DC    IL1'13'                   SECTORS TO READ
    0691 0700             0692 3483        DC    AL2(LALOVR)               ADDR CORE INPUT AREA
                               3484 *
                          0693 3485 LALWFL EQU   *                         ADDR DISK PARAM LIST
    0693 02               0693 3486        DC    AL1(@DPUT)                WRITE CODE
    0694 07               0694 3487        DC    AL1(@DVBCY)               BASE CYL FOR
    0695                  0695 3488 LALSDS DS    CL1                       DISP FROM BASE CYL
    0695                       3489        ORG   LALSDS                    INITIALLY SET TO THE
    0695 02               0695 3490        DC    AL1(LALSDP)               DISP OF FUNCTIONS FROM VM START
    0696 0D               0696 3491        DC    IL1'13'                   SECTORS TO WRITE.
    0697 0700             0698 3492        DC    AL2(LALOVR)               ADDR CORE OUTPUT AREA
                               3493 *
                          0699 3494 LALPLI EQU   *                         ADDR DISK PARM LIST
    0699 01               0699 3495        DC    AL1(@DGET)                READ FUNCT!ON CODE
    069A                  069B 3496 LALASC DS    CL2                       INTERPRETER DISK ADDRESS,
    069A                       3497        ORG   *-2                       * INITIALLY SET TO CONTAIN THE
    069A 0020             069B 3498        DC    AL2(#$INST)               * DISK ADDR SHORT PREC INTERP
    069C 10               069C 3499        DC    AL1(#$@INS)               * SECTOR COUNT
    069D 0600             069E 3500        DC    AL2(#$$INS)               INTERPRETER LOAD/ENTRY ADDR
```

```
                               3502 ***********************************************************************
                               3503 ***********************************************************************
                               3504 *                                                                     *
                               3505 *      LALLOC PHASE 1 - ARRAY ALLOCATION                              *
                               3506 *                                                                     *
                               3507 ***********************************************************************
                               3508 ***********************************************************************
                               3509 *
                               3510 ***********************************************************************
                               3511 * ALLOCATE ARITHMETIC ARRAY AREAS WITHIN VIRTUAL MEMORY
                               3512 ***********************************************************************
                               3513 *
                               3514 * SET LOADER ALLOCATION PHASE BASE
                               3515 *
                          070D 3516        USING LAL170,@BR                SET LALLOC SASE ADDR
    069F C2 01 070D            3517 LAL100 LA    LAL170,@BR                LOAD LALLOC BASE
                               3518 *
                               3519 * TEST FOR PRECISION
                               3520 *
    06A3 38 40 03D0            3521        TBN   $XIND1,$XPREC             IS PRECISION LONG ?
    06A7 F2 90 0B              3522        JF    LAL110                    YES, CHANGE LNG INSTRUCTIONS
                               3523 *
                               3524 * MODIFY PRECISION SENSITIVE INSTRUCTIONS TO PROCESS LONG PRECISION
                               3525 *
    06AA 7C 09 01              3526        MVI   LAL170+@Q(,@BR),B@LILP    SET LNG FOR LOOP CTR
    06AD 7C 09 E1              3527        MVI   LALAEL(,@BR),B@LILP       SET LNG TO DECR FOR BASE ADDR
    06B0 1C 01 069B EB         3528        MVC   LALASC(@DADDR),LALLPI(,@BR)  INTERPRETER CALL TO LNG PREC
                               3529 *
                               3530 * DETERMINE SPACE AVAILABLE FOR ARRAYS IN VIRTUAL MEMORY
                               3531 *
    06B5 4C 07 E9 1A07         3532 LAL110 MVC   LALVAP(LALX08,@BR),LALVA4 PLACE VADDR PARAMS IN WORK AREA
    06BA 5F 00 E4 F3           3533        SLC   LALAP2-1(1,@BR),LALH01(,@BR)  SET PSUEDO REG END
    06BE 5F 01 E5 E3           3534        SLC   LALAP2(LALX02,@BR),LALAP1(,@BR) REGION 1 SIZE
    06C2 5F 01 E9 E7           3535        SLC   LALAP4(LALX02,@BR),LALAP3(,@BR) REGION 2 SIZE
    06C6 5F 01 E3 E1           3536        SLC   LALAP1(LALX02,@BR),LALAEL(,@BR) DECR TO SET UP BASE ADDR
    06CA 5F 01 E7 E1           3537        SLC   LALAP3(LALX02,@BR),LALAEL(,@BR) DECR TO SET UP BASE ADDR
                               3538 *
                               3539 * SELECT ARRAY SYMBOL TABLE ELEMENT AND TEST FOR ARRAY DEFINITION
                               3540 *
    06CE C2 02 1CC4            3541 LAL120 LA    LALASM,@XR                ADDR ARITH SYM TBL LH BYTE
    06D2 B5 02 00              3542 LAL125 L     *-*(,@XR),@XR             LOAD DOPE VECTOR VADDR FROM
    06D4                       3543        ORG   LAL125+@D1                * ARITH ARRAY SYMBOL TBL,
    06D4 39               06D4 3544        DC    AL1(B@LL12-1)             * BEGINNING WITH FINAL TBL
    06D5                       3545        ORG   LAL125+3                  * ELEMENT
    06D5 76 02 F1              3546        A     LALH00(,@BR),@XR          TEST ENTRY FOR ZERO VADDR
    06D8 D0 81 3E              3547        BE    LAL220(,@BR)              ZERO, TRY NEXT TBL ENTRY
                               3548 *
                               3549 * TEST SECOND ARRAY DIMENSION FOR ZERO
                               3550 *
    06DB 76 02 ED              3551 LAL130 A     LALAAC(,@BR),@XR          CONVERT D/V VADDR TO CADDR
    06DE 9D 01 03 F1           3552        CLC   B@ACD2(,@XR),LALH00(B@LDMN,@BR)  IS 2ND DIM NULL
    06E2 F2 01 0E              3553        JNE   LAL150                    NO, CALCULATE NO. OF ELEMENTS
                               3554 *
                               3555 * TEST IF A MATRIX - SET DIMENSION DEFAULTS
                               3556 *
    06E5 9C 01 03 F7           3557        MVC   B@ACD2(,@XR),LALH10(B@LDMN,@BR)  SET DIM 2 DEFAULT
    06E9 B8 C0 00              3558        TBN   B@AFLG(,@XR),B@D2MK       TEST FOR A MATRIX ARRAY
    06EC F2 90 04              3559        JF    LAL150                    BRANCH IF NOT A MATRIX
    06EF 9C 01 01 F7           3560        MVC   B@ACD1(,@XR),LALH10(B@LDMN,@BR)  SET DIM 1 DEFAULT
                               3561 *
                               3562 * CALC NUMBER OF ELEMENTS IN ARRAY AND SET DOPE VECTOR MAXIMUM SIZE
                               3563 *
    06F3 BB C0 00              3564 LAL150 SBF   B@AFLG(,@XR),B@D2MK       SET ARRAY DEFINED BIT OFF
    06F6 5F 01 DB DB           3565        SLC   LALECT(LALEBC,@BR),LALECT(,@BR)  ZERO ELEMENT CTR
    06FA 6C 01 DD 01           3566        MVC   LALLCT(B@LDMN,@BR),B@ACD1(,@XR)  SET LOOP CTR
    06FE AE 01 05 03           3567 LAL160 ALC   B@AMAX(B@LDMN,@XR),B@ACD2(,@XR)  ADD 2ND DIM TO CTR
    0702 D0 02 CE              3568        BNL   LAL900(,@BR)              IF VM OVFLO GO TO ERROR RTN.
    0705 5F 01 DD F3           3569        SLC   LALLCT(LALEBC,@BR),LALH01(,@BR)  DECR LOOP CTR
    0709 C0 84 06FE            3570        BH    LAL160                    REPEAT LOOP UNTIL CTR LT 1
                               3571 *
                               3572 * CALCULATE VM SPACE THE ARRAY WILL OCCUPY
                               3573 *
    070D 7C 00 DB              3574 LAL170 MVI   LALECT(,@BR),*-*          SET LOOP COUNTER EQUAL TO THE
    070E                       3575        ORG   LAL170+@Q                 * LNG IN BYTES OF THE ELEMENT
    070E 05               070E 3576        DC    AL1(B@LISP)               * INITIALLY SET FOR THE SHORT
    0710                       3577        ORG   LAL170+3                  * PRECISION LENGTH
    0710 5F 01 DF DF           3578        SLC   LALSIZ(LALEBC,@BR),LALSIZ(,@BR)  ZERO ARRAY SIZE CTR
    0714 6E 01 DF 05           3579 LAL180 ALC   LALSIZ(LALEBC,@BR),B@AMAX(,@XR)  ADD ELEMENT CT TO SIZE CT
    0718 D0 02 CE              3580        BNL   LAL900(,@BR)              IF VM OVFLO GO TO ERROR RTN
    071B 5F 00 DB F3           3581        SLC   LALECT(LALSBC,@BR),LALH01(,@BR)  DECR MULTIPLY LOOP CT
    071F D0 84 07              3582        BH    LAL180(,@BR)              REPEAT LOOP UNTIL CT LT ONE
                               3583 *
                               3584 * DETERMINE IF ARRAY WILL FIT IN EITHER VIRTUAL MEMORY REGION
                               3585 *
    0722 5D 01 E9 DF           3586 LAL190 CLC   LALAP4(LALX02,@BR),LALSIZ(,@BR)  SIZE LT REGION 2 ?
    0726 F2 02 16              3587        JNL   LAL210                    YES, ALLOCATE SPACE
    0729 5D 01 E5 DF           3588        CLC   LALAP2(LALX02,@BR),LALSIZ(,@BR)  FIT IN REGION 1 ?
    072D F2 82 AB              3589        JL    LAL900                    NO, VM OVFLO GO TO ERROR RTN
                               3590 *
                               3591 * ALLOCATE ARRAY SPACE IN REGION 1, SET ARRAY BASE ADDRESS AND UPDATE
                               3592 * VM REGION 1 POINTERS
                               3593 *
    0730 9C 01 07 E3           3594 LAL200 MVC   B@ABAS(@VADDR,@XR),LALAP1(,@BR)  SET ARRAY BASE ADDR
    0734 5E 01 E3 DF           3595        ALC   LALAP1(LALX02,@BR),LALSIZ(,@BR)  INCR TO NEXT BASE ADDR
    0738 5F 01 E5 DF           3596        SLC   LALAP2(LALX02,@BR),LALSIZ(,@BR)  DECR TO NEW REGION SIZE
    073C F2 87 0C              3597        J     LAL220                    PROCESS NEXT TABLE ENTRY
                               3598 *
                               3599 * ALLOCATE ARRAY SPACE IN REGION 2,SET ARRAY BASE ADDRESS AND UPDATE
                               3600 * VM REGION 2 POINTERS
                               3601 *
    073F 9C 01 07 E7           3602 LAL210 MVC   B@ABAS(@VADDR,@XR),LALAP3(,@BR)  SET ARRAY BASE ADDR
    0743 5E 01 E7 DF           3603        ALC   LALAP3(LALX02,@BR),LALSIZ(,@BR)  INCR TO NEXT BASE ADDR
    0747 5F 01 E9 DF           3604        SLC   LALAP4(LALX02,@BR),LALSIZ(,@BR)  DECR TO NEW REGION SIZE
                               3605 *
                               3606 * DECREMENT ARRAY TABLE POINTER TO ACCESS NEXT TABLE ENTRY
                               3607 *
    074B 1F 00 06D4 F5         3608 LAL220 SLC   LAL125+@D1,LALH02(LALSTD,@BR)  DECR SYM TABLE POINTER
    0750 C0 84 06CE            3609        BH    LAL120                    PROCESS TABLE UNTIL LAST ENTRT

                               3611 ***********************************************************************
                               3612 * ALLOCATE CHARACTER ARRAY AREAS WITHIN VIRTUAL MEMORY
                               3613 ***********************************************************************
                               3614 *
                               3615 * PREPARE REGION 1 & 2 FIRST AVAIL ADDRESSES FOR CHAR ARRAY BASE ADDR
                               3616 *
    0754 5E 01 E3 E1           3617 LAL400 ALC   LALAP1(LALX02,@BR),LALAEL(,@BR)  RESTORE FROM ARITH BASE
    0758 5E 01 E7 E1           3618        ALC   LALAP3(LALX02,@BR),LALAEL(,@BR)  *
    075C 5F 01 E3 EF           3619        SLC   LALAP1(LALX02,@BR),LALCEL(,@BR)  DECR TO SET UP BASE ADDR
    0760 5F 01 E7 EF           3620        SLC   LALAP3(LALX02,@BR),LALCEL(,@BR)  *
                               3621 *
                               3622 * SELECT ARRAY SYMBOL TABLE ENTRY AND TEST FOR ARRAY DEFINITION
                               3623 *
    0764 C2 02 1CFE            3624 LAL410 LA    LALCSM,@XR                TABLE ADDR CHAR SYM TBL LH BYTE
    0768 B5 02 00              3625 LAL420 L     *-*(,@XR),@XR             LOAD DOPE VECTOR VADDR FROM
    076A                       3626        ORG   LAL420+@D1                * CHAR ARRAY SYMBOL TABLE,
    076A 39              076A 3627        DC    AL1(B@LL13-1)             * BEGINNING WITH FINAL TABLE
    076B                       3628        ORG   LAL420+3                  * ENTRY
    076B 76 02 F1              3629        A     LALH00(,@BR),@XR          TEST ENTRY FOR ZERO VADDR
    076E D0 81 B3              3630        BE    LAL490(,@BR)              YES, TEST NEXT ENTRY
                               3631 *
                               3632 * TEST IF ARRAY PREVIOUSLY DIMENSIONED - IF NO, SET DIMENSION DEFAULT
                               3633 *
    0771 76 02 ED              3634 LAL430 A     LALAAC(,@BR),@XR          CVRT D/V VADDR TO CADDR
    0774 BB 80 00              3635 LAL434 SBF   B@AFLG(,@XR),B@DAMK       SET DIM FLAG OFF
    0777 9D 01 01 F1           3636 LAL436 CLC   B@CDMN(,@XR),LALH00(B@LDMN,@BR)  ARRAY BEEN DIMENSIONED ?
    077B F2 01 04              3637        JNE   LAL440                           YES, SKIP DEFAULT SET
    077E 9C 01 01 F7           3638        MVC   B@CDMN(,@XR),LALH10(B@LDMN,@BR)  SET DIM DEFAULT
                               3639 *
                               3640 * CALULATE VIRTUAL MEMORY SPACE THE ARRAY WILL OCCUPY
                               3641 *
    0782 7C 13 DB              3642 LAL440 MVI   LALECT(,@BR),B@LCRV       LOOP CT = CHAR VAR LNG
    0785 5F 01 DF DF           3643        SLC   LALSIZ(LALEBC,@BR),LALSIZ(,@BR)  ZERO ARRAY SIZE CT
    0789 6E 01 DF 01           3644 LAL450 ALC   LALSIZ(LALEBC,@BR),B@CDMN(,@XR)  ADD DIM TO SIZE CT
    078D D0 02 CE              3645        BNL   LAL900(,@BR)              IF VM OVFLO GO TO ERROR RTN
    0790 5F 00 DB F3           3646        SLC   LALECT(LALSBC,@BR),LALH01(,@BR) DECR MULTIPLY LOOP CT
    0794 D0 84 7C              3647        BH    LAL450(,@BR)              REPEAT LOOP UNTIL CT LT ONE
                               3648 *
                               3649 * DETERMINE IF ARRAY WILL FIT IN EITHER VIRTUAL MEMORY REGION
                               3650 *
    0797 5D 01 E9 DF           3651 LAL460 CLC   LALAP4(LALX02,@BR),LALSIZ(,@BR)  LT REGION 2 ?
    079B F2 02 16              3652        JNL   LAL480                    YES, ALLOCATE SPACE
    079E 5D 01 E5 DF           3653        CLC   LALAP2(LALX02,@BR),LALSIZ(,@BR)  FIT IN REGION 1 ?
    07A2 F2 82 36              3654        JL    LAL900                    NO, VM OVFLO GO TO ERROR RTN
                               3655 *
                               3656 * ALLOCATE ARRAY SPACE IN REGION 1, SET ARRAY BASE ADDRESS AND UPDATE
                               3657 * VM REGION 1 POINTERS
                               3658 *
    07A5 9C 01 03 E3           3659 LAL470 MVC   B@CBAS(@VADDR,@XR),LALAP1(,@BR)  SET ARRAY BASE ADDR
    07A9 5E 01 E3 DF           3660        ALC   LALAP1(LALX02,@BR),LALSIZ(,@BR)  INCR TO NEXT BASE ADDR
    07AD 5F 01 E5 DF           3661        SLC   LALAP2(LALX02,@BR),LALSIZ(,@BR)  DECR TO NEW REGION SIZE
    07B1 F2 87 0C              3662        J     LAL490                           PROCESS NEXT TBL ENTRY
                               3663 *
                               3664 * ALLOCATE ARRAY SPACE IN REGION 2, SET ARRAY BASE ADDRESS AND UPDATE
                               3665 * VM REGION 2 POINTERS
                               3666 *
    07B4 9C 01 03 E7           3667 LAL480 MVC   B@CBAS(@VADDR,@XR),LALAP3(,@BR)  SET ARRAY BASE ADDR
    07B8 5E 01 E7 DF           3668        ALC   LALAP3(LALX02,@BR),LALSIZ(,@BR)  INCR TO NEXT BASE ADDR
    07BC 5F 01 E9 DF           3669        SLC   LALAP4(LALX02,@BR),LALSIZ(,@BR)  DECR TO NEW REGION SIZE
```

```
                               3670 *
                               3671 * DECREMENT ARRAY TABLE POINTER TO ACCESS HEFT TABLE
                               3672 *
    07C0 5F 00 5D F5           3673 LAL490 SLC   LAL420+@D1(,@BR),LALH02(LALSTD,@BR)  DECR SYM TBL PT
    07C4 D0 84 57              3674        BH    LAL410(,@BR)              PROCESS TABLE UNTIL LAST ENTRY
                               3675 *
                               3676 * PLACE THE COMPLETED CODE VECTORS INTO VIRTUAL MEMORY
                               3677 *
    07C7 5E 01 E3 EF           3678        ALC   LALAP1(LALX02,@BR),LALCEL(,@BR) RESTORE NEXT BYTE PTR  1-3
    07CB C0 87 17E7            3679 LAL495 B     DL4ICS                    DISK IOCR RTN
    07CF 0805             07D0 3680        DC    AL(@CADDR)(LALPUT)        ADDR DISK PARM LIST
    07D1 C0 87 0025            3681        B     $DISKN                    WAIT FOR COMPLETION
    07D5 057F             07D6 3682        DC    AL(@CADDR)($WAITF)        WAIT PARM
                               3683 *
                               3684 * EXIT LALLOC PHASE ONE TO LDFILE
                               3685 *
    07D7 C0 87 080B            3686 LAL500 B     LDFILE                    TO LDFILE
                               3687 *
                               3688 * LALLOC ERROR ROUTINE
                               3689 *
    07DB 3C B0 03CD            3690 LAL900 MVI   $CAERR,@@E611             SET ERROR COND CODE
    07DF 3C A0 03CE            3691        MVI   $ERRPG,$$$NLN             SET RTN TO NOT PRINT LINE NO.
    07E3 C0 87 0469            3692        B     $CAERK                    ABORT LOADER, PRINT ERROR MSG

                               3694 ***********************************************************************
                               3695 * LALLOC PHASE ONE - CONSTANTS, WORK AREAS AND EQUATES
                               3696 ***********************************************************************
                               3697 *
                               3698 * LALLOC EQUATES REFERENCING CONSTANTS
                               3699 *
                          1A00 3700 LALVA1 EQU   B$LDRP                    CADDR 1ST BYTE REGION 1
                          1A02 3701 LALVA2 EQU   B$LDRP+2                  CADDR LAST BYTE REGION 1
                          1A04 3702 LALVA3 EQU   B$LDRP+4                  CADDR 1ST BYTE REGION 2
                          1A07 3703 LALVA4 EQU   B$LDRP+B@DL04             CADDR LAST BYTE REGION 2
                          1CC4 3704 LALASM EQU   B$LDRP+B@DL11+1           CADDR ARITH SYM TBL LH BYTE
                          1CFE 3705 LALCSM EQU   B$LDRP+B@DL12+1           CADDR CHAR SYM TBL LH BITE
                          0001 3706 LALSBC EQU   1                         ARRAY SIZE CT BYTE COUNT
                          0001 3707 LALSTD EQU   1                         SYMBOL TBL PT DECR
                          0002 3708 LALX02 EQU   @VADDR                    BYTES IN VADDR
                          0002 3709 LALEBC EQU   2                         ELEMENT CT BYTE COUNT
                          0008 3710 LALX08 EQU   8                         NO. BYTES IN VADDR PARMS
                               3711 *
                               3712 * LALLOC WORK AREAS
                               3713 *
    07E7                  07E8 3714 LALECT DS    CL2                       ELEMENT COUNTER
    07E9                  07EA 3715 LALLCT DS    CL2                       MULTIPLY LOOP COUNTER
    07EB                  07EC 3716 LALSIZ DS    CL2                       ARRAY SIZE COUNTER
    07ED                  07EE 3717 LALAEL DS    CL2                       ARRAY ELEMENT LENGTH
    07ED                       3718        ORG   *-2                       * INITIALLY SET TO THE SHORT
    07ED 0005             07EE 3719        DC    AL2(B@LISP)               * PRECISION LENGTH
    07EF                  07F6 3720 LALVAP DS    CL8                       LALLOC PARM WORK AREA
                               3721 *
                               3722 * LALLOC CONSTANTS
                               3723 *
    07F7 1C84             07F8 3724 LALLPI DC    AL(@DADDR)(#$INLN)        LONG PREC INTERPRETER DADDR
    07F9 1F08             07FA 3725 LALAAC DC    AL2(B$LDRP+B@DL16+1)      ARITH ARRAY DOPE
```

```
                               3726 *                                      * VECTOR VIRTUAL TO CORE ADDR
    07FB 0013             07FC 3727 LALCEL DC    AL2(B@LCRV)               CHARACTER VARIABLE LENGTH
    07FD 0000             07FE 3728 LALH00 DC    IL2'0'                    BINARY INTEGER 0
    07FF 0001             0800 3729 LALH01 DC    IL2'1'                    BINARY INTEGER 1
    0801 0002             0802 3730 LALH02 DC    IL2'2'                    BINARY INTEGER 2
    0803 000A             0804 3731 LALH10 DC    IL2'10'                   FOR DIMENSION DEFAULTS
                               3732 *
                               3733 * DISK DARAMETER LIST
                               3734 *
                          0805 3735 LALPUT EQU   *                         ADDR DISK PARAM LIST
    0805 02               0805 3736        DC    AL1(@DPUT)                WRITE CODE
    0806 07               0806 3737        DC    AL1(@DVBCY)               BASE CYL FOR VM
    0807 FE               0807 3738        DC    AL1(@VMDDV)               SECTOR DISP FROM BASE
    0808 02               0808 3739        DC    XL1'02'                   SECTOR COUNT
    0809 1D08             080A 3740        DC    AL(@CADDR)(B$LDRP+B@DL16-512+1)  CORE OUTPUT AREA
                               3741 *
                               3742 * LALLOC EQUATES REFERENCING PROGRAM
                               3743 *
                          07F0 3744 LALAP1 EQU   LALVAP-6                  REGION 1 START AND ACCUMULATOR
                          07F2 3745 LALAP2 EQU   LALVAP-4                  REGION 1 END ADDRESS AND SIZE
                          07F4 3746 LALAP3 EQU   LALVAP-2                  REGION 2 START AND ACCUMULATOR
                          07F6 3747 LALAP4 EQU   LALVAP-0                  REGION 2 END ADDRESS AND SIZE
```

```
                               3749 ***********************************************************************
                               3750 * 5703-XM1   COPYRIGHT IBM CORP 1970                                  *
                               3751 *            REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083      *
                               3752 *                                                                     *
                               3753 ***********************************************************************
                               3754 *STATUS -                                                             *
                               3755 *   VERSION 1 MODIFICATION 0                                          *
                               3756 *                                                                     *
                               3757 *FUNCTION -                                                           *
                               3758 *   * LDFILE ALLOCATES VIRTUAL MEMORY SPACE FOR FILE BUFFERS          *
                               3759 *   * FILE DIRECTORY TWO IS GENERATED                                 *
                               3760 *   * FILE DIRECTORIES ONE AND TWO ARE PLACED IN VIRTUAL MEMORY       *
                               3761 *                                                                     *
                               3762 *ENTRY POINTS                                                         *
                               3763 *   * LDFILE HAS ONLY ONE ENTRY POINT                                 *
                               3764 *   * CALLING SEQUENCE IS:                                            *
                               3765 *        B     LDFILE                                                *
                               3766 *                                                                     *
                               3767 *INPUT -                                                              *
                               3768 *   * LALVAL - 2 BYTES, FOR FIRST FREE VIRTUAL ADDRESS IN VIRTUAL     *
                               3769 *     MEMORY REGION 1 (END OF ALLOCATED ARRAY SPACE)                  *
                               3770 *   * LALVA2 - 2 BYTES, FOR FIRST NON-FREE VIRTUAL ADDRESS IN         *
                               3771 *     VIRTUAL MEMORY REGION 1 (START OF CONSTANTS)                    *
                               3772 *   * LALVA3 - 2 BYTES, FOR FIRST FREE VIRTUAL ADDRESS IN VIRTUAL     *
                               3773 *     MEMORY REGION 2 (END OF ALLOCATED ARRAY SPACE)                  *
                               3774 *   * LALVA4 - 2 BYTES, FOR FIRST NON-FREE VIRTUAL ADDRESS IN         *
                               3775 *     VIRTUAL MEMORY REGION 2 (START OF FUNCTION AND ARRAY TABLE)     *
                               3776 *   * FILE DIRECTORY ONE - 256 BYTES, CONTAINS 8 32-BYTE RECORDS      *
                               3777 *   * TRACE REFERENCE LIST - IF IN TRACE MODE                         *
                               3778 *                                                                     *
                               3779 *OUTPUT -                                                             *
                               3780 *   * FILE DIRECTORY ONE - 256 BYTES, UNCHANGED FROM INPLT            *
                               3781 *   * FILE DIRECTORY TWO - 256 BYTES, 8 16-BYTE RECORDS               *
                               3782 *      * 3RD BYTE IN RECORD, CONTAINS THE FIRST VIRTUAL PAGE NUMBER   *
                               3783 *        ALLOCATED TO THAT FILE                                       *
                               3784 *      * 4TH BYTE IN RECORD, CONTAINS THE NUMBER OF VIRTUAL PAGES     *
                               3785 *        ALLOCATED FOR THE FILE                                       *
                               3786 *                                                                     *
                               3787 *EXTERNAL REFERENCES -                                                *
                               3788 *        $XIND1 - SYSTEM EXECUTION INDICATOR                          *
                               3789 *        DL4ICS - 4-TRACK LIOCS                                       *
                               3790 *        $DISKN - SYSTEM DISK IOCR                                    *
                               3791 *        $CAERK - SYSTEM ERROR MESSAGE ROUTINE                        *
                               3792 *        $CAERR - ERROR ROUTINE ERROR CODE PARAMETER                  *
                               3793 *        $ERRPG - ERROR ROUTINE LINE NUMBER PARAMETER                 *
                               3794 *        LVINIT - LOADER VM INITIALIZATION                            *
                               3795 *                                                                     *
                               3796 *EXIT, NORMAL -                                                       *
                               3797 *   LDFILE HAS ONLY ONE NORMAL EYIT                                   *
                               3798 *        LVINIT - AFTER FILE ALACATION                                *
                               3799 *                                                                     *
                               3800 *EXITS, ERROR -                                                       *
                               3801 *        $CAERK - WITH ERROR CODE                                     *
                               3802 *           @@E613 - STORAGE SPACE REQUIRED FOR FILES TOO LARGE       *
                               3803 *                                                                     *
                               3804 *TABLES/WORK ARFAS -                                                  *
```

```
3805 *   * THE CONSTANTS AND WORK AREAS RESIDE AT THE OF THE EXEC CODE     *
3806 *   * BUFFER 1 - 256 BYTES, FOR FILE DIRECTORY 1 (AT 06A0)            *
3807 *   * BUFFER 2 - 256 BYTES, FOR FILE DIRECTORY 2 (AT 1900)            *
3808 *   * BUFFER 3 - 256 BYTES, FOR TRACE REFERENCE LIST (AT 1800)        *
3809 *                                                                     *
3810 *ATTRIBUTES -                                                         *
3811 *   LDFILE IS REUSABLE                                                *
3812 *                                                                     *
3813 *CHARACTER CODE DEPENDENCY -                                          *
3814 *   N/A                                                               *
3815 *                                                                     *
3816 *NOTES -                                                              *
3817 *   ERROR PROCEDURES                                                  *
3818 *      * ERROR CODE IS SET AT $CAERR                                  *
3819 *      * $ERRPG IS SET WITH $$$LNL TO OMIT LINE NUMBER                *
3820 *                                                                     *
3821 *   REGISTER USAGE                                                    *
3822 *      * BOTH REGISTERS ARE USED DURING EXECUTION                     *
3823 *      * THE REGISTERS ARE NOT SAVED OR RESTORED                      *
3824 *                                                                     *
3825 *   SAVED RESTORED AREAS                                              *
3826 *      N/A                                                            *
3827 *                                                                     *
3828 *   MODIFICATION CONSIDERATIONS                                       *
3829 *      LDSFILE MUST LOAD CORE WITH THE TRACE REFERENCE LIST, IF       *
3830 *      IN TRACE MODE, BEFORE FILE DIRECTORY TWO IS PLACED IN          *
3831 *      VIRTUAL MEMORY OVER IT                                         *
3832 *                                                                     *
3833 *   REQUIRED MODULES                                                  *
3834 *        @SYSEQ - COMMON SYSTEM EQUATES                               *
3835 *        @FXDEQ - SYSTEM NUCLEUS ADDRESSES AND INDICATORS             *
3836 *        @VMDEQ - VM DIRECTORY EQUATES                                *
3837 *        @ERMEQ - GENERAL ERROR MESSAGE EQUATES                       *
3838 *        @B@EQU - COMPILER SYSTEM EQUATES                             *
3839 *        DL4ICS - 4-TRACK LIOCS                                       *
3840 *        LALLCC - LOADER ARRAY ALLOCATION                             *
3841 *        LVINIT - LOADER VM INITIALIZATION                            *
3842 *                                                                     *
3843 *   OTHER                                                             *
3844 *      N/A                                                            *
3845 ***********************************************************************
```

```
                               3847 ***********************************************************************
                               3848 *                                                                     *
                               3849 *      VIRTUAL MEMORY FILE BUFFER ALLOCATION                          *
                               3850 *                                                                     *
                               3851 ***********************************************************************
                               3852 *
                               3853 * LDFILE ENTRY AND SET LDFILE BASE
                               3854 *
                          080B 3855 LDFILE EQU  *                          LDFILE ENTRY POINT
                          0917 3856        USING LDF230,@BR                SET BASE ADDR
    080B C2 01 0917            3857        LA    LDF230,@BR                LOAD LDFILE BASE
                               3858 *
                               3859 * SAVE CORE LOCATIONS 1A00-1BFF ON DISK IN TEMPORARY WORK AREA
                               3860 *
    080F C0 87 0025            3861        B     $DISKN                    WRITE BUFFERS TO DISK
    0813 0A18             0814 3862        DC    AL(@CADDR)(LDFSBF)        DPL ADDR
    0815 C0 87 0025            3863        B     $DISKN                    WAIT FOR WRITE COMPLETE
    0819 057F             081A 3864        DC    AL(@CADDR)($WAITF)        WAIT DPL
                               3865 *
                               3866 * READ FILE DIRECTORY 1 INTO FILE BUFFER 1
                               3867 *
    081B C0 87 0025            3868 LDF100 B     $DISKN                    FILE DIRECTOR 1
    081F 09FA             0820 3869        DC    AL2(LDFFDR)               DPL ADDR
                               3870 *
                               3871 * CLEAR FILE DIRECTORY 2 TO ZERO
                               3872 *
    0821 3C 00 19FF            3873 LDF110 MVI   LDFEB2,@ZERO              PLACE ZERO IN RH BYTE
    0825 0C FE 19FE 19FF       3874        MVC   LDF2BP(LDFLTH),LDFEB2     PROPOGATE THROUGH FIELD
                               3875 *
                               3876 * MOVE THE WORK FILE NAME TO FILE DIRECTORY 2
                               3877 *
    082B C2 02 1900            3878 LDF120 LA    LDFBF2,@XR                 CADDR OF FILE 2
    082F 8C 07 0A 0443         3879        MVC   @$D2PN(@$L2PN,@XR),$WFNME MOVE FILE NAME
                               3880 *
                               3881 * CALCULATE REMAINING AVAILABLE PAGES IN VIRTUAL MEMORY
                               3882 *
    0834 1E 00 07EF D6         3883 LDF130 ALC   LDFAP1(LDFPGL),LDFH01(,@BR)  NO, INCR PAGE NO.
    0839 1E 00 07F3 D6         3884        ALC   LDFAP3(LDFPGL),LDFH01(,@BR)  *
                               3885 *
                               3886 * WAIT FOR TILE DIRECTORY TO BE READ INTO CORE
                               3887 *
    083E C0 87 0025            3888 LDF145 B     $DISKN                    WAIT TOR COMPLETION OR READ
    0842 057F             0843 3889        DC    AL(@CADDR)($WAITF)        WAIT PRAM
                               3890 *
                               3891 * SET PTR TO LAST ENTRY IN FILE DIR 1 AND TEST FOR FILE DEFINITION
                               3892 *
    0844 C2 02 1A00            3893 LDF150 LA    LDFBF1,@XR                ADDR FILE DIRECTORY 1 LH BYTE
                               3894 *
                               3895 * MODIFICATIONS DONE FOR MORE THAN 08 ALLOCATE COMMANDS
                               3896 *
    0848 BD 00 1F              3897        CLI   @$D1SW(,@XR),@ZERO        2 PAGES OF FILE DIRECTORY 1 ?
    084B F2 81 31              3898        JE    LDF160                    NO - CONTINUE
    084E 3D 00 07EF            3899        CLI   LDFAP1,@ZERO              IS SPACE AVAILABLE IN REGION 1 ?
    0852 F2 81 0E              3900        JE    LDF155                    NO - CHECK REGION 2
    0855 4C 00 E2 07EF         3901        MVC   LDFSAV(LDFPGL,@BR),LDFAP1  SAVE 1ST PAGE IN REGION 1
    085A 1E 00 07EF D6         3902        ALC   LDFAP1(LDFPGL),LDFH01(,@BR)  INCREMENT REGION 1 POINTER
    085F C0 87 0874            3903        B     LDF157                    GO SET UP DPL FOR 2ND D1 PAGE
    0863 3D 00 07F3            3904 LDF155 CLI   LDFAP3,@ZERO              IS SPACE AVAILABLE IN REGION 2 ?
    0867 F2 81 AD              3905        JE    LDF230                    NO - ERROR CONDITION
    086A 4C 00 E2 07F3         3906        MVC   LDFSAV(LDFPGL,@BR),LDFAP3   SAVE 1ST PAGE IN REGION 2
    086F 1E 00 07F3 D6         3907        ALC   LDFAP3(LDFPGL),LDFH01(,@BR)  INCREMENT REGION 2 POINTER
    0874 5C 00 FD E2           3908 LDF157 MVC   LDFF3W+@DSAD(,@BR),LDFSAV(,@BR) SET UP DPL FOR D1 PAGE 2
    0878 9C 00 1F E2           3909        MVC   @$D1SW(,@XR),LDFSAV(,@BR) SAVE PAGE NO IN D1 PAGE 1
    087C 7C 00 E2              3910        MVI   LDFSAV(,@BR),@ZERO        SET LOOP SW = 0
    087F E2 02 00              3911 LDF160 LA    *-*(,@XR),@XR             LOAD ADDR OF 1ST BYTE IN ENTRY
    0881                       3912        ORG   LDF160+@D1                * BEGINNING WITH THE LAST
    0881 E0               0881 3913        DC    AL1(LDFLFE)               * FILE DIRECTORY ONE
    0882                       3914        ORG   LDF160+3                  * ENTRY
    0882 BD 00 00              3915        CLI   LDFBY0(,@XR),LDFNUL       TEST ENTRY FOR ZERO (UNDEFINED)
    0885 F2 81 11              3916        JE    LDF190                    YES, TEST NEXT ENTRY
                               3917 *
                               3918 * DETERMINE DEVICE TYPE AND INCREMENT DEVICE COUNTER
                               3919 *
    0888 B9 C0 00              3920 LDF170 TBF   @$D1DC(,@XR),@$MBPD+@$MBSD  IS DEVICE TYPE DISK ?
    088B F2 90 07              3921        JF    LDF180                    YES, INCR DISK CTR
    088E 5E 00 DE D6           3922        ALC   LDFNDD(LDFCTB,@BR),LDFH01(,@BR)  NO, INCR NON-DISK CTR
    0892 F2 87 04              3923        J     LDF190                    TEST NEXT ENTRY
    0895 5E 00 DF D6           3924 LDF180 ALC   LDFDKD(LDFCTB,@BR),LDFH01(,@BR)  INCR DISK CTR
                               3925 *
                               3926 * DECREMENT FILE DIRECTORY 1 ENTRY POINTER TO ACCESS NEXT FILE ENTRY
                               3927 *
    0899 1F 00 0881 D8         3928 LDF190 SLC   LDF160+@D1,LDFD1R(LDFCTB,@BR) DECR FILE PT
    089E C0 02 08AF            3929        BNL   LDF192                    PROCESS UNTIL LAST ENTRY
                               3930 *
                               3931 * MODIFICATIONS DONE FOR MORE THAN 08 ALLOCATE COMMANDS
                               3932 *
    08A2 BD 00 1F              3933        CLI   @$D1SW(,@XR),@ZERO        IS 2 SECTOR SW ON ?
    08A5 F2 81 1D              3934        JE    LDF200                    NO - LAST ENTRY HAS BEEN PROC.
    08A8 3C 60 0881            3935        MVI   LDF160+@D1,LDFL2E         POINT AT LAST ENTRY IN 2ND PAGE
    08AC 7C 60 E2              3936        MVI   LDFSAV(,@BR),LDFL2E       SET LDFSAV NON-ZERO
    08AF 7D 00 E2              3937 LDF192 CLI   LDFSAV(,@BR),@ZERO        ESTABLISH INDEX REGISTER FOR
    08B2 F2 81 08              3938        JE    LDF194                    * PAGE 1 OR PAGE 2 OF FILE
    08B5 C2 02 1B00            3939        LA    LDFBF3,@XR                * DIRECTORY 1 BEING SEARCHED
    08B9 C0 87 087F            3940        B     LDF160                    * BY LOOP
    08BD C2 02 1A00            3941 LDF194 LA    LDFBF1,@XR
    08C1 C0 87 087F            3942        B     LDF160
                               3943 *
                               3944 * TOTAL AVAILABLE PAGES AND DEVICE TYPE COUNTERS AND
                               3945 * TEST DEVICE TOTAL FOR ZERO
                               3946 *
    08C5 4C 00 DD 07F1         3947 LDF200 MVC   LDFTAP(LDFCTB,@BR),LDFRIP  SHIFT REGION 1 PAGE SIZE
    08CA 4E 00 DD 07F5         3948        ALC   LDFTAP(LDFCTB,@BR),LDFR2P  INCR BY REGION 2 PAGE SIZE
    08CF 5C 00 E0 DF           3949        MVC   LDFTDT(LDFCTB,@BR),LDFDKD(,@BR)  SHIFT DISK CTR
    08D3 5E 00 E0 DE           3950        ALC   LDFTDT(LDFCTB,@BR),LDFNDD(,@BR)  INCR BY NON-DISK CTR
    08D7 7D 00 E0              3951        CLI   LDFTDT(,@BR),LDFNUL        ANY FILE BUFS TO ALLOCATE ?
    08DA F2 81 D7              3952        JE    LDF310                     WRITE DIRECTORY AND XIT
                               3953 *
                               3954 * DETERMINE IF ENOUGH VIRTUAL MEMORY PAGES ARE AVAILABLE IN REGION 1,
                               3955 * ALLOWING ONE PAGE FOR EACH FILE
                               3956 *
    08DD 1D 00 07F1 E0         3957 LDF202 CLC   LDFRIP,LDFTDT(LDFCTB,@BR)  ENOUGH PAGES ?
    08E2 F2 82 27              3958        JL    LDF220                    NO, TEST COMBINED REGIONS
```

```
                               3959 *
                               3960 * CALCULATE THE NUMBER OF SECTOR THAT CAN BE ALLOCATED EVENLY TO EACH
                               3961 * DISK DEVICE ENTRY
                               3962 *
    08E5 7D 00 DF              3963 LDF203 CLI   LDFDKD(,@BR),LDFNUL       ANY DISK DEVICES
    08E8 F2 81 25              3964        JE    LDF225                    NO, ALLOT I BFR
    08EB 1F 00 07F1 DE         3965        SLC   LDFRIP,LDFNDD(LDFCTB,@BR) DECR BY NON-DISK CTR
    08F0 1F 00 07F1 DF         3966 LDF204 SLC   LDFRIP,LDFDKD(LDFCTB,@BR) DECR BY NO. DISK ENTRIES
    08F5 F2 82 08              3967        JL    LDF206                    PROC UNTIL CTR LT 0
    08F8 5E 00 E1 D6           3968        ALC   LDFCNT(LDFCTB,@BR),LDFH01(,@BR)  INCR SECTOR COUNT
    08FC C0 87 08F0            3969        B     LDF204                    RECYCLE LOOP
                               3970 *
                               3971 * TEST SECTOR COUNTER FOR VALUE GREATER THAN EIGHT
                               3972 *
    0900 7D 08 E1              3973 LDF206 CLI   LDFCNT(,@BR),LDFX08       GT EIGHT ?
    0903 F2 04 65              3974        JNH   LDF250                    NO, PROC ALL ENTRIES
    0906 7C 08 E1              3975        MVI   LDFCNT(,@BR),LDFX08       SET CNT TO 8
    0909 F2 87 5F              3976        J     LDF250                    PROC ALL ENTRIES
                               3977 *
                               3978 * ENOUGH IN TOTAL AVAILABLE PAGES FOR EACH FILE
                               3979 *
    090C 5D 00 DD E0           3980 LDF220 CLC   LDFTAP(LDFCTB,@BR),LDFTDT(,@BR)  ENOUGH PAGES
    0910 C2 02 1940            3981 LDF225 LA    LDFFE2,@XR                FILE 2 1ST ENTRY
    0914 F2 02 0C              3982        JNL   LDF240                    YES, ALLOCATE PAGES
                               3983 *
                               3984 * ERROR CONDITION CODE AND EXIT TO SYSTEM ERROR ROUTINE
                               3985 *
    0917 3C B2 03CD            3986 LDF230 MVI   $CAERR,@@E613             SET ERROR COND CODE
    091B 3C A0 03CE            3987        MVI   $ERRPG,$$$NLN             SET RTN NOT TO PRINT LINE NO.
    091F C0 87 0469            3988        B     $CAERK                    ABORT LOADER, PRINT ERROR MSG
                               3989 *
                               3990 * SET FILE POINTER TO LAST ENTRY IN FILE DIRECTORY 1
                               3991 *
    0923 C2 01 1A00            3992 LDF240 LA    LDFBF1,@BR                FILE 1 1ST ENTRY
    0927 7D 00 00              3993 LDF244 CLI   LDFBY0(,@BR),LDFNUL       TEST ENTRY FOR ZERO (UNDEFINED)
    092A F2 81 87              3994        JE    LDF310                    WRITE TO VM
                               3995 *
                               3996 * ALLOCATE 1 SECTOR FOR EACH FILE AND COMPLETE FILE DIRECTORY 2
                               3997 *
    092D 0F 00 07F1 09ED       3998        SLC   LDFRIP(LDFCTB),LDFH01     REGION 1 ALLOCATED ?
    0933 F2 82 25              3999        JL    LDF249                    NO, PROCESS UNTIL ALL PROC
    0936 8C 00 02 0000         4000 LDF246 MVC   @$D2VB(LDFCTB,@XR),*-*    PAGE ALLOCATED FOR BFR
    0939                       4001        ORG   *-2                       * INITIALLY SET TO 1ST
    0939 07EF             093A 4002        DC    AL2(LDFAP1)               * AVAIL PAGE REGION 1
    093B 8C 00 03 09ED         4003        MVC   @$D2BS(LDFCTB,@XR),LDFH01  SECTORS ALLOCATED TO 1
    0940 0E 00 0000 09ED       4004 LDF247 ALC   *-*(LDFCTB),LDFH01        INCR TO NEW 1ST AVAIL PAGE
    0942                       4005        ORG   LDF247+2                  * INITIALLYIEF TO THE
    0942 07EF             0943 4006        DC    AL2(LDFAP1)               * 1S1 AVAILABLE PAGE
    0946                       4007        ORG   *+2                       * IN REGION 1
    0946 0F 00 09F7 09ED       4008        SLC   LDFTDT(LDFCTB),LDFH01     ALL FILES ALLOCATED
    094C F2 04 65              4009        JNH   LDF310                    YES, WRITE TO DISK AND EXIT
                               4010 *
                               4011 * INCREMENT FILE DIRECTORY POINTERS
                               4012 *
    094F 36 02 09F1            4013 LDF248 A     LDFD2R,@XR                INCR TO NEXT ENTRY
    0953 36 01 09EF            4014        A     LDFD1R,@BR                INCR TO NEXT ENTRY
    0957 C0 87 0927            4015        B     LDF244                    CONTINUE LOOP
                               4016 *
                               4017 * PREPARE LOOP FOR REGION 2 BASE ADDRESS
                               4018 *
    095B 0C 01 093A 09F3       4019 LDF249 MVC   LDFOP2(LDFADB),LDFPR2     SET LOOP FOR REGION 2
    0961 0C 01 0943 09F3       4020        MVC   LDFOP1(LDFADB),LDFPR2     SET LOOP FOR REGION 2
    0967 C0 87 0927            4021        B     LDF244                    PROCESS REGION 2
                               4022 *
                               4023 * SET FILE POINTER TO LAST ENTRY IN FILE DIRECTORY 1
                               4024 *
    096B C2 01 1A00            4025 LDF250 LA    LDFBF1,@BR                FILE 1 1ST ENTRY
    096F C2 02 1940            4026        LA    LDFFE2,@XR                FILE 2 1ST ENTRY
    0973 7D 00 00              4027 LDF260 CLI   LDFBY0(,@BR),LDFNUL       TEST ENTRY FOR ZERO (UNDEFINED)
    0976 F2 81 3B              4028        JE    LDF310                    WRITE TO VM
                               4029 *
                               4030 * DETERMINE DEVICE TYPE AND COMPLETE THAT ENTRY IN FILE 2
                               4031 *
    0979 79 C0 00              4032        TBF   @$D1DC(,@BR),@$MBPD+@$MBSD  IS DEVICE TYPE DISK ?
    097C F2 90 13              4033        JF    LDF280                    YES, INCR DISK CTR
                               4034 *
                               4035 * COMPLETE FILE DIRECTORY 2 ENTRY FOR A NON-DISK DEVICE TYPE
                               4036 *
    097F 8C 00 02 07EF         4037 LDF270 MVC   @$D2VB(LDFCTB,@XR),LDFAP1  PG ALLOCATED FOR BUFFER
    0984 8C 00 03 09ED         4038        MVC   @$D2BS(LDFCTB,@XR),LDFH01  SECTORS ALLOCATED TO 1
    0989 0E 00 07EF 09ED       4039        ALC   LDFAP1(LDFCTB),LDFH01     INCR 1ST AVAIL PG
    098F F2 87 10              4040        J     LDF290                    LOOP UNTIL ALL ENTRIES PROC
                               4041 *
                               4042 * COMPLETE FILE DIRECTORY 2 DISK DEVICE ENTRIES
                               4043 *
    0992 8C 00 02 07EF         4044 LDF280 MVC   @$D2VB(LDFCTB,@XR),LDFAP1  PG ALLOCATED FOR FILE BFR
    0997 8C 00 03 09F8         4045        MVC   @$D2BS(LDFCTB,@XR),LDFCNT  SECTORS ALLOCATED
    099C 0E 00 07EF 09F8       4046        ALC   LDFAP1(LDFCTB),LDFCNT     INCR 1ST AVAIL PG
                               4047 *
                               4048 * INCREMENT FILE DIRECTORY POINTERS
                               4049 *
    09A2 36 02 09F1            4050 LDF290 A     LDFD2R,@XR                INCR TO NEXT ENTRY
    09A6 36 01 09EF            4051        A     LDFD1R,@BR                INCR TO NEXT ENTRY
    09AA 0F 00 09F7 09ED       4052        SLC   LDFTDT,LDFH01(1)          ALL FILES ALLOCATED ?
    09B0 C0 84 0973            4053        BH    LDF260                    NO, CONTINUE LOOP
                               4054 *
                               4055 * PLEACE BOTH FILE DIRECTORIES IN VIRTUAL MEMORY
                               4056 *
    09B4 C0 87 17E7            4057 LDF310 B     DL4ICS                    DISK IOCR RTN
    09B8 0A06             09B9 4058        DC    AL(@CADDR)(LDFFDW)        ADDR DISK PARM LIST
    09BA C0 87 17E7            4059        B     DL4ICS                    DISK IOCR RTN
    09BE 0A0C             09BF 4060        DC    AL(@CADDR)(LDFF2W)        ADDR DISK PARM LIST
                               4061 *
                               4062 * MODIFICATIONS DONE FOR MORE THAN 08 ALLOCATE COMMANDS
                               4063 *
    09C0 C2 02 1A00            4064        LA    LDFBF1,@XR                SET POINTER TO PAGE 1 OF D1
    09C4 BD 00 1F              4065        CLI   @$D1SW(,@XR),@ZERO        IS 2 SECTOR SWITCH ON ?
    09C7 F2 81 06              4066        JE    LDF315                    NO - CONTINUE
    09CA C0 87 17E7            4067        B     DL4ICS                    WRITE PAGE 2 OF D1 TO VM
    09CE 0A12             09CF 4068        DC    AL(@CADDR)(LDFF3W)        ADDR DISK PAW LIST
                               4069 *
                               4070 * READ NE TRACE REFERENCE LIST INTO A CORE BUFFER IF NEEDED
                               4071 *
    09D0 38 04 03D0            4072 LDF315 TBN   $XIND1,$TRACE             TRACE SW ON ?
    09D4 F2 90 06              4073        JF    LDF317                    NO. SKIP READ
    09D7 C0 87 17E7            4074        B     DL4ICS                    READ TRACE REFERENCE INTO CORE
    09DB 0A00             09DC 4075        DC    AL(@CADDR)(LDFTRL)        * AND OVERLAY DIRECTORY 2
                               4076 *
                               4077 * RESTORE CORE BUFFER AREA AND EXIT FROM LDFILE TO LVINIT
                               4078 *
    09DD C0 87 0025            4079 LDF317 B     $DISKN                    RESTORE CORE BUFFER AREA
    09E1 0A1E             09E2 4080        DC    AL(@CADDR)(LDFRBF)        * 1A00-1BFF
    09E3 C0 87 0025            4081        B     $DISKN                    WAIT FOR READ COMPLETE
    09E7 057F             09E8 4082        DC    AL(@CADDR)($WAITF)        WAIT DPL ADDR
    09E9 C0 87 0A24            4083 LDF320 B     LVINIT                    EXIT LDFILE

                               4085 ***********************************************************************
                               4086 * LDFILE CONSTANTS, WORK AREAS AND EQUATES
                               4087 ***********************************************************************
                               4088 *
                               4089 * LDFILE EQUATES REFERENCING CONSTANTS
                               4090 *
                          0000 4091 LDFFN2 EQU   0                         DISP USER FILE NAME FILE 2
                          0000 4092 LDFNUL EQU   0                         TEST FOR 0 DISP
                          0000 4093 LDFBY0 EQU   0                         DISP OF STATUS BYTE IN ENTRY
                          0001 4094 LDFPGL EQU   1                         BYTES IN A PG NO.
                          0001 4095 LDFCTB EQU   1                         BYTES IN THE CTR
                          0002 4096 LDFADB EQU   2                         BYTES IN CADDR
                          0003 4097 LDFFN1 EQU   3                         USER FILE NAME FILE 1
                          0008 4098 LDFX08 EQU   8                         MAX SECTORS TO ALLOCATE
                          0010 4099 LDFLN2 EQU   16                        LENGTH FILE 2 ENTRY
                          0080 4100 LDFDMK EQU   X'80'                     DEVICE CODE MASK FOR DISK TYPE
                          0070 4101 LDFLE2 EQU   112                       DISP TO 1ST FILE 2 ENTRY - 16
                          00E0 4102 LDFLFE EQU   224                       DISP TO LAST FILE 1 ENTRY
                          0060 4103 LDFL2E EQU   96                        DISP TO LAST FILE ENTRY-D1 PG 2
                          00FF 4104 LDFLTH EQU   255                       BYTES TO ZERO IN BFR
                          1900 4105 LDFTLB EQU   X'1900'                   TRACE REFERENCE LIST BUFFER
                          1A00 4106 LDFBF1 EQU   X'1A00'                   FILE DIRECTORY 1-1ST PAGE
                          1900 4107 LDFBF2 EQU   X'1900'                   FILE DIRECTORY 2 BFR
                          1B00 4108 LDFBF3 EQU   X'1B00'                   FILE DIRECTORY 1-2ND PAGE
                          1A02 4109 LDFVA2 EQU   B$LDRP+2                  CADDR LAST BYTE REGION 1
                          1A06 4110 LDFVA4 EQU   B$LDRP+6                  CADDR LAST BYTE REGION 2
                          0955 4111 LDFTLA EQU   X'0955'                   DISK ADDR TO SAVE CADDR 1A00-1BFF
                               4112 *
                               4113 * LDFILE CONSTANTS
                               4114 *
    09ED 01               09ED 4115 LDFH01 DC    IL1'1'                    BINARY INTEGER 1
    09EE 0020             09EF 4116 LDFD1R DC    AL2(@$L1E)                LENGTH OF FILE 1 ENTRY
    09F0 0010             09F1 4117 LDFD2R DC    AL2(@$L2E)                LENGTH OF FILE 2 ENTRY
    09F2 07F3             09F3 4118 LDFPR2 DC    AL2(LDFAP3)               ADDR 1ST AVAIL PG IN REGION 2
                               4119 *
                               4120 * LDFILE WORK AREAS
                               4121 *
    09F4                  09F4 4122 LDFTAP DS    CL1                       TOTAL AVAIL PGS
    09F5                  09F5 4123 LDFNDD DS    CL1                       NON-DISK DEVICE CTR
    09F5                       4124        ORG   LDFNDD                    * INITIALLY SET TO
    09F5 00               09F5 4125        DC    XL1'00'                   * ZERO
    09F6                  09F6 4126 LDFDKD DS    CL1                       DISK DEVICE CTR
    09F6                     4127        ORG   LDFDKD                    * INITIALLY SET TO
    09F6 00               09F6 4128        DC    XL1'00'                   * ZERO
    09F7                  09F7 4129 LDFTDT DS    CL1                       TOTAL DEVICE TYPES DEFINED
    09F8                  09F8 4130 LDFCNT DS    CL1                       DISK SECTOR COUNTER
    09F8                     4131        ORG   LDFCNT                    * INITIALLY SET TO
    09F8 00               09F8 4132        DC    XL1'00'                   * ZERO
    09F9                  09F9 4133 LDFSAV DS    CL1                       SAVE AREA FOR 2ND D1 PAGE NO.
                               4134 *
                               4135 * LDFILE DISK PARAMETER LISTS
                               4136 *
                          09FA 4137 LDFFDR EQU   *                         ADDR DISK PARM LIST
    09FA 01               09FA 4138        DC    AL1(@DGET)                READ CODE
    09FB 0459             09FC 4139        DC    AL2(#@#IO1)               DADDR OF FIRST SECTOR OF D1
    09FD 02               09FD 4140        DC    AL1(#@@#SC)               SECTOR COUNT
    09FE 1A00             09FF 4141        DC    AL2(LDFBF1)               ADDR CORE INPUT AREA
                               4142 *
                          0A00 4143 LDFTRL EQU   *                         ADDR DISK PARAM LIST
    0A00 01               0A00 4144        DC    AL1(@DGET)                READ CODE
    0A01 07               0A01 4145        DC    AL1(@DVBCY)               BASE CYL FOR VM
    0A02 54               0A02 4146        DC    XL1'54'                   SECTOR DISP FROM BASE CIL
    0A03 01               0A03 4147        DC    XL1'01'                   SECTOR COUNT
    0A04 1900             0A05 4148        DC    AL(@CADDR)(LDFTLB)        ADDR CORE INPUT AREA
                               4149 *
                          0A06 4150 LDFFDW EQU   *                         ADDR DISK PARAM LIST
    0A06 02               0A06 4151        DC    AL1(@DPUT)                WRITE CODE
    0A07 07               0A07 4152        DC    AL1(@DVBCY)                BASE CYL FOR VM
    0A08 00               0A08 4153        DC    AL1(@VMFD1)               SECTOR DISP FROM BASE CYL
    0A09 01               0A09 4154        DC    XL1'01'                   SECTOR COUNT
    0A0A 1A00             0A0B 4155        DC    AL(@CADDR)(LDFBF1)              ADDR CORE OUTPUT AREA
                               4156 *
                          0A0C 4157 LDFF2W EQU   *                         ADDR DISK PARAM LIST
    0A0C 02               0A0C 4158        DC    AL1(@DPUT)                WRITE CODE
    0A0D 07               0A0D 4159        DC    AL1(@DVBCY)               BASE CYL FOR VM
    0A0E 01               0A0E 4160        DC    AL1(@VMFD2)               SECTOR DISP FROM BASE CYL
    0A0F 01               0A0F 4161        DC    XL1'01'                   SECTOR COUNT
    0A10 1900             0A11 4162        DC    AL(@CADDR)(LDFBF2)        ADDR CORE OUTPUT AREA
                          0A12 4163 LDFF3W EQU   *                         ADDR DISK PARAM LIST
    0A12 02               0A12 4164        DC    AL1(@DPUT)                WRITE CODE
    0A13 07               0A13 4165        DC    AL1(@DVBCY)               BASE CYL FOR OF
    0A14 01               0A14 4166        DC    AL1(@VMFD2)               SECTOR DISP FROM BASE CYL
    0A15 01               0A15 4167        DC    XL1'01'                   SECTOR COUNT
    0A16 1B00             0A17 4168        DC    AL(@CADDR)(LDFBF3)        ADDR CORE OUTPUT AREA
                               4169 *
                          0A18 4170 LDFSBF EQU   *                         DSK PARM LIST (SAVE CORE BFR)
    0A18 02               0A18 4171        DC    AL1(@DPUT)                WRITE CODE
    0A19 044D             0A1A 4172        DC    AL2(#@LDSV)               DADDR
    0A1B 02               0A1B 4173        DC    AL1(#@@#SC)               SECTOR COUNT
    0A1C 1A00             0A1D 4174        DC    AL(@CADDR)(LDFBF1)        CORE ADDR
                          0A1E 4175 LDFRBF EQU   *                         DSK PARM LIST (RESTORE CORE BFR)
    0A1E 01               0A1E 4176        DC    AL1(@DGET)                READ CODE
    0A1F 044D             0A20 4177        DC    AL2(#@LDSV)               DADDR
    0A21 02               0A21 4178        DC    AL1(#@@#SC)               SECTOR COUNT
    0A22 1A00             0A23 4179        DC    AL(@CADDR)(LDFBF1)        CORE ADDR
                               4180 *
                               4181 * LDFILE EQUATES REFERENCING PROGRAM
                               4182 *
                          07EF 4183 LDFAP1 EQU   LALVAP-7                  REGION 1 1ST PAGE
                          07F1 4184 LDFRIP EQU   LALVAP-5                  REMAINING PAGES REGION 1 CADDR
                          07F3 4185 LDFAP3 EQU   LALVAP-3                  REGION 2 1ST PAGE
                          07F5 4186 LDFR2P EQU   LALVAP-1                  REMAINING PAGES REGION 2 CADDR
                          1AFF 4187 LDFEB1 EQU   LDFBF1+255                RH BYTE BFR 1
                          1940 4188 LDFFE2 EQU   LDFBF2+@$D2E1             1ST FILE 2 ENTRY
                          19FF 4189 LDFEB2 EQU   LDFBF2+255                RH BYTE BFR 2
                          093A 4190 LDFOP2 EQU   LDF246+4                  2ND OPERAND CADDR
                          0943 4191 LDFOP1 EQU   LDF247+3                  ADDR 1ST OPERAND
                          19FE 4192 LDF2BP EQU   LDFEB2-1                  TO PROPAGATE ZEROS
```

```
                               4194 ***********************************************************************
                               4195 * 5703-XM1   COPYRIGHT IBM CORP 1970                                  *
                               4196 *            REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083      *
                               4197 *                                                                     *
                               4198 ***********************************************************************
                               4199 *STATUS -                                                             *
                               4200 *   VERSION 1 MODIFICATION 0                                          *
                               4201 *                                                                     *
                               4202 *FUNCTION -                                                           *
                               4203 *   * LVINIT INITIALIZES THE ARITHMETIC AND CHARACTER SCALAR          *
                               4204 *     VARIABLES USED IN THE BASIC PROGRAM                             *
                               4205 *   * THE ARITHMETIC AND CHARACTER ARRAYS REFERENCED IN THE BASIC     *
                               4206 *     PROGRAM ARE INITIALIZED                                         *
                               4207 *   * THE INTERNAL VARIABLES AND CONSTANTS ARE INITIALIZED AND MOVED  *
                               4208 *     TO THEIR VIRTUAL MEMORY LOCATIONS                               *
                               4209 *   * THE TRACE SITS ARE SET IN VARIABLES TO BE TRACED                *
                               4210 *   * THE PRECISION BIT IN ARITHMETIC ELEMENTS IS SET TO THE REQUIRED *
                               4211 *     PRECISION                                                       *
                               4212 *                                                                     *
                               4213 *ENTRY POINTS -                                                       *
                               4214 *   LVINIT HAS ONLY ONE ENTRY POINT                                   *
                               4215 *   CALLING SEQUENCE IS                                               *
                               4216 *        B     LVINIT                                                 *
                               4217 *                                                                     *
                               4218 *INPUT -                                                              *
                               4219 *   * LVIVA1 - 1 BYTE, FOR FIRST FREE VIRTUAL PAGE IN VIRTUAL         *
                               4220 *     MEMORY REGION 1 (END OF PMC)                                    *
                               4221 *   * LVIVA2 - 1 BYTE, FOR FIRST FREE VIRTUAL PAGE IN VIRTUAL         *
                               4222 *     MEMORY REGION 2 (END OF VARIABLES)                              *
                               4223 *   * LVIICA - 2 BYTES, FOR VIRTUAL ADDRESS OF THE FIRST BYTE OF      *
                               4224 *     INTERNAL CONSTANTS                                              *
                               4225 *   * LVIIVA - 2 BYTES, FOR VIRTUAL ADDRESS OF THE FIRST BYTE OR      *
                               4226 *     INTERNAL VARIABLES                                              *
                               4227 *   * SYMBOL AND ARRAY TABLE                                          *
                               4228 *      * LETTER VARIABLE TABLE - 58 BYTES, 29 2-BYTE ENTRIES (LVT)    *
                               4229 *        LETTER DIGIT TABLE - 580 BYTES, 290 2-BYTE ENTRIES (LDT)     *
                               4230 *      * CHARACTER VARIABLE TABLE - 58 BYTES, 29 2-BYTE ENTRIES (CVT) *
                               4231 *      * ARITHMETIC ARRAY SYMBOL TABLE - 58 BYTES, 29 2-BYTE          *
                               4232 *        ENTRIES                                                      *
                               4233 *      * CHARACTER ARRAY SYMBOL TABLE - 58 BYTES, 29 2-BYTE           *
                               4234 *        ENTRIES                                                      *
                               4235 *   * FUNCTION AND ARRAY TABLE (FAT) 406 BYTES                        *
                               4236 *      * ARITHMETIC ARRAY DOPE VECTORS - 29 8-BYTE ENTRIES            *
                               4237 *      * CHARACTER ARRAY DOPE VECTORS - 29 4-BYTE ENTRIES             *
                               4238 *   * TRACE REFERENCE LIST - 256 BYTES, CONTAINS THE TRACE            *
                               4239 *     COMMAND LINE                                                    *
                               4240 *                                                                     *
                               4241 *OUTPUT -                                                             *
                               4242 *   * VIRTUAL MEMORY                                                  *
                               4243 *      * INITIALIZED INTERNAL CONSTANTS                               *
                               4244 *      * INITIALTZED INTERNAL VARIABLES                               *
                               4245 *      * INITIALIZED ARITHMETIC SCALAR VARIABLES                      *
                               4246 *      * INITIALIZED CHARACTER SCALAR VARIABLES                       *
                               4247 *      * INITIALIZED ARITHMETIC SCALAR ELEMENTS                       *
                               4248 *      * INITIALIZED CHARACTER ARRAY ELEMENTS                         *
                               4249 *                                                                     *
                               4250 *EXTERNAL REFERENCES -                                                *
                               4251 *        $XIND1 - SYSTEM EXECUTION INDICATOR                          *
                               4252 *        DL4ICS - 4-TRACK LIOCS                                       *
                               4253 *        $DISKN - SYSTEM DISK IOCR                                    *
                               4254 *        $CAERK - SYSTEM ERROR MESSAGE ROUTINE ENTRY                  *
                               4255 *        $CAERR - ERROR ROUTINE ERROR CODE PARAMETER                  *
                               4256 *        $ERRPG - ERROR ROUTINE LINE NUMBER PARAMETER                 *
                               4257 *        C4BIN2 - DECIMAL TO BINARY CONVERSION                        *
                               4258 *        C4BVAL - C4BIN2 NUMBER PARAMETER AREA                        *
                               4259 *        LRADDR - LOADER ADDRESS RESOLUTION                           *
                               4260 *                                                                     *
                               4261 *EXITS, NORMAL -                                                      *
                               4262 *   LVINIT HAS ONE NORMAL EXIT                                        *
                               4263 *        LRADDR - AFTER VM INITIALIZATION                             *
                               4264 *                                                                     *
                               4265 *EXITS, ERROR -                                                       *
                               4266 *        $CAERK - WITH ERROR CODES:                                   *
                               4267 *           @@E250 - VARIABLE NOT IN PROGRAM                          *
                               4268 *           @@E252 - SUBSCRIPT EXCEEDS <ARRAY SIZE LIMIT>             *
                               4269 *           @@E253 - ARRAY NOT IN PROGRAM                             *
                               4270 *           @@E294 - NO NON-ARRAY <VARIABLES> IN PROGRAM              *
                               4271 *           @@E256 - INCONSISTENT NUMBER OF SUBSCRIPTS                *
                               4272 *                                                                     *
                               4273 *TABLES/WORK AREAS -                                                  *
                               4274 *   * THE CONSTANTS AND WORK AREAS RESIDE AT THE END OF EXECUTABLE    *
                               4275 *     CODE AND ARE REFERENCED BY @BR                                  *
                               4276 *   * COMPILER COMMON PARAMETER BLOCK, LOCATED AT CORE ADDRESS        *
                               4277 *     1A00 TO 1F00 CONTAINS:                                          *
                               4278 *      * VIRTUAL MEMORY REGION POINTERS (SIX)                         *
                               4279 *      * VARIABLE SYMBOL TABLES (FIVE)                                *
                               4280 *      * FUNCTION AND ARRAY TABLE                                     *
                               4281 *   * TRACE LIST - 58 BYTES, 29 2-BYTE ENTRIES (INTERNAL USE ONLY)    *
                               4282 *   * TRACE REFERENCE LIST                                            *
                               4283 *   * BUFFER 1 - 256 BYTES, FOR SINGLE ELEMENT INITIALIZATION         *
                               4284 *   * BUFFER 2 - 4 CORE PAGES, FOR ARRAY INITIALIZATION               *
                               4285 *                                                                     *
                               4286 *ATTRIBUTES -                                                         *
                               4287 *   N/A                                                               *
                               4288 *                                                                     *
                               4289 *CHARACTER CODE DEPENDENCY -                                          *
                               4290 *   THE OPERATION OF THIS MODULE DEPENDS UPON THE FOLLOWING           *
                               4291 *   PROPERTIES OF THE INTERNAL REPRESENTATION OF THE EXTERNAL         *
                               4292 *   CHARACTER SET                                                     *
                               4293 *      * MOST CODING HAS BEEN ARRANGED SO THAT REDEFINITION OF        *
                               4294 *        CHARACTER CONSTANTS BY REASSEMBLY, WILL-RESULT IN A CORRECT  *
                               4295 *        MODULE FOR THE NEW DEFINITION                                *
                               4296 *      * ALPHABETIC LETTERS A THROUGH Z ARE PRESUMED TO BE CODED IN   *
                               4297 *        INCREASING COLLATING SEQUENCE, AND THE RANGE OF CHARACTER    *
                               4298 *        CONSTANTS FOR THIS SERIES IS EXPECTED TO EXCLUDE ALL NUMERIC *
                               4299 *        CHARACTER CONSTANTS                                          *
                               4300 *      T NUMERIC CHARACTERS 0 - 9 ARE PRESUMED TO BE CODED IN         *
                               4301 *        INCREASING COLLATING SEQUENCE                                *
                               4302 *      * EXTENDED ALPHABETIC LETTERS ($, #, @) ARE PRESUMEMED TO BE   *
                               4303 *        IN INCREASING COLLATING SEQENCE, AND ARE ALL EXPECTED TO     *
                               4304 *        COLLATE LOWER THAN LETTER (A)                                *
                               4305 *      * DECIMAL NUMBERS MUST BE CODED SO THAT THE LOW ORDER FOUR     *
                               4306 *        BITS, WHEN CONSIDERED AS A BINARY INTEGER, IDENTIFY NE       *
                               4307 *        VALUE OF THE DIGIT                                            *
                               4308 *   THE SPECIFIC INSTRUCTIONS (INSTRUCTION SEQUENCES) WHICH REQUIRE   *
                               4309 *   MODIFICATION IF THESE PROPERTIES OF THE CHARACTER SET ARE CHANGED *
                               4310 *   MAY BE IDENTIFIED BY -                                             *
                               4311 *      * THE TABLE IDENTIFIED BY LABEL LVIATL                          *
                               4312 *                                                                      *
                               4313 *NOTES -                                                               *
                               4314 *   ERROR PROCEDURES                                                   *
                               4315 *      LVINIT HAS TWO ERROR PROCEDURES                                 *
                               4316 *        * PRINT UP ARROW BENEATH ERROR                                *
                               4317 *          SET ERROR CODE AT $CAERR                                    *
                               4318 *              @@E250                                                  *
                               4319 *              @@E252                                                  *
                               4320 *              @@E253                                                  *
                               4321 *              @@E256                                                  *
                               4322 *          SHIFT TRACE REFERENCE LIST T. PRIMARY INPUT BUFFER AREA    *
                               4323 *          SET @XR POINTER TO ERROR BYTE IN BUFFER                     *
                               4324 *          SET $ERRPG TO $ERKEY                                        *
                               4325 *        * PRINT-ERROR MESSAGE ONLY                                    *
                               4326 *          SET ERROR CODE AT SCAERR                                    *
                               4327 *              @@E254                                                  *
                               4328 *          SET $ERRPG TO $ERKEY                                        *
                               4329 *                                                                      *
                               4330 *   REGISTER USAGE                                                     *
                               4331 *      * BOTH REGISTERS ARE USED DURING EXECUTION                      *
                               4332 *      * THE REGISTERS ARE NOT SAVED OR RESTORED                       *
                               4333 *                                                                      *
                               4334 *   SAVED/RESTORED                                                     *
                               4335 *      N/A                                                             *
                               4336 *                                                                      *
                               4337 *   MODIFICATION CONSIDERATIONS                                        *
                               4338 *      LVINIT USES AS A BUFFER FOUR CORE PAGES WHICH OVERLAY A         *
                               4339 *      PORTION OF THE EXECUTION LOADER (INCLUDING THE FIRST            *
                               4340 *      PAGE OF LVINIT), 0700 - OAFF                                    *
                               4341 *                                                                      *
                               4342 *   REQUIRED MODULES                                                   *
                               4343 *        @SYSEQ - COMMON SYSTEM EQUATES                                *
                               4344 *        @FXDEQ - SYSTEM NUCLEUS ADDRESSES AND INDICATORS              *
                               4345 *        @VMDEQ - VM DIRECTORY EQUATES                                 *
                               4346 *        @ERMEQ - GENERAL ERROR MESSAGE EQUATES                        *
                               4347 *        $BSEQU - COMPILER FIXED EQUATES                               *
                               4348 *        $B@EQU - COMPILER SYSTEM EQUATES                              *
                               4349 *        DL4ICS - 4-TRACK LIOCS                                        *
                               4350 *        C4BIN2 - DECIMAL TO BINARY CONVERSION                         *
                               4351 *        LRADDR - LOADER ADDRESS RESOLUTION                            *
                               4352 *                                                                      *
                               4353 *   OTHER                                                              *
                               4354 *      N/A                                                             *
                               4355 ************************************************************************
                               4357 ************************************************************************
                               4358 ************************************************************************
                               4359 *                                                                      *
                               4360 *      LVINIT - VIRTUAL MEMORY INITIALIZATION                          *
                               4361 *                                                                      *
                               4362 ************************************************************************
                               4363 ************************************************************************
                               4364 *
                               4365 * LVINIT ENTRY POINT AND SET BASE ADDR
                               4366 *
                          0A24 4367 LVINIT EQU   *                         LVINIT ENTRY POINT
                          131F 4368        USING LVI945,@BR                SET BASE ADDR
    0A24 C2 01 131F            4369        LA    LVI945,@BR                LOAD LVINIT BASE
    0A28 7C 01 D5              4370        MVI   LVITSW(,@BR),LVISWO       SET ALL SW ON
                               4371 *
                               4372 * TEST FOR TRACE OPTIONS
                               4373 *
    0A2B 38 10 03D0            4374 LVI010 TBN   $XIND1,$TRALL             TRACE ALL SW ON
    0A2F F2 90 14              4375        JF    LVI012                    NO, TURN TRACE SW OFF
    0A32 7C 00 D5              4376        MVI   LVITSW(,@BR),LVINUL       SET ALL SW OFF
    0A35 38 20 03D0            4377 LVI014 TBN   $XIND1,$TRVAR             TRACE SELECTED VAR SW ON
    0A39 F2 10 18              4378        JT    LVI015                    YES SCAN FOR VAR IN REF LIST
    0A3C 38 10 03D0            4379        TBN   $XIND1,$TRALL             TRACE ALL SW ON
    0A40 F2 90 57              4380        JF    LVI045                    NO, SET ALL TRACE SWS OFF
    0A43 F2 87 5E              4381        J     LVI050                    YES, INIT INTERNAL CONSTANTS
                               4382 *
                               4383 * SET ALL SCALAR VARIABLE TRACE SWITCHES OFF
                               4384 *
    0A46 3C 00 1473            4385 LVI012 MVI   LVILTB,LVINUL             ZERO LAST TRACE LIST BYTE
    0A4A 0C 38 1472 1473       4386        MVC   LVILTB-1(LVITLL),LVILTB   PROPAGATE ZEROS THROUGH FIELD
    0A50 C0 87 0A35            4387        B     LVI014                    TEST IF SELECTED VARS TO TRACE
                               4388 *
                               4389 * INITIALIZE LINE SCAN ROUTINE AND SET LIST POINTER
                               4390 *
    0A54 C2 02 1900            4391 LVI015 LA    LVITRL,@XR                1ST BYTE TRACE REF LIST
    0A58 B6 02 01              4392        A     LVITD1(,@XR),@XR          ADD DISP TO LIST PT
    0A5B 34 02 1177            4393        ST    LVI784+@OP1,@XR           SAVE 1ST BYTE ADDR
    0A5F 34 02 1037            4394        ST    LVI710+@OP1,@XR           SAVE 1ST BYTE ADDR
                               4395 *
                               4396 * SCAN TRACE REFERENCE LIST AND DIRECT THE PROCESSING OF BACK ENTRY
                               4397 *
    0A63 BD 1E 00              4398 LVI020 CLI   LVITD0(,@XR),@EOS         AT END OF LIST
    0A66 F2 81 3B              4399        JE    LVI050                    YES, INIT INTERNAL CONSTANTS
    0A69 6C 00 D8 00           4400        MVC   LVILSA(LVIBYC,@BR),LVITD0(,@XR)  SAVE LETTER
    0A6D BD 4D 01              4401        CLI   LVITD1(,@XR),B@LPAR       IS BYTE A LEFT PAREN
    0A70 C0 81 0C19            4402        BE    LVI140                    YES, PROCESS ARITH ARRAY VAR
    0A74 BD 5B 01              4403        CLI   LVITD1(,@XR),B@CVAR       IS BYTE A DOLLAR SIGN
    0A77 F2 01 0B              4404        JNE   LVI030                    N0, TEST FOR MO
    0A7A BD 4D 02              4405        CLI   LVITD2(,@XR),B@LPAR       IS BYTE A LEFT PAREN
    0A7D C0 81 0C8C            4406        BE    LVI170                    YL$, PROCESS CHAR ARRAY
    0A81 C0 87 0BD1            4407        B     LVI120                    NO, PROCESS CHAR VAR
    0A85 BD F0 01              4408 LVI030 CLI   LVITD1(,@XR),B@DEC0       IS BYTE A DIGIT
    0A88 F2 82 6F              4409        JL    LVI080                    NO, PROCESS ARITH VAR
    0A8B 6C 00 E0 01           4410        MVC   LVIDSA(1,@BR),LVITD1(,@XR)  SAVE DIGIT
    0A8F F2 87 B1              4411        J     LVI100                    YES, PROCESS LETTER-DIGIT VAR
                               4412 *
                               4413 * INCREMENT LIST OINTER TO FIRST BYTE OF NEXT LIST VARIABLE
                               4414 *
    0A92 C0 87 0BB9            4415 LVI040 B     LVI110                    INCR LIST PT
    0A96 C0 87 0A63            4416        B     LVI020                    CONTINUE UNTIL (OS
                               4417 *
                               4418 * SET ALL TRACE SWITCHES IN THE TRACE TABLE OFF
                               4419 *
    0A9A 3C 00 1473            4420 LVI045 MVI   LVILTB,LVINUL             ZERO LAST TRACE LIST BYTE
    0A9E 0C 38 1472 1473       4421        MVC   LVILTB-1(LVITLL),LVILTB   PROPAGATE ZEROS THROUGH FIELD
                               4422 *
                               4423 * TEST FOR PRECISION - IF LONG MODIFY PROGRAM TO PROCESS LONG PRECISION
                               4424 *
    0AA4 38 40 03D0            4425 LVI050 TBN   $XIND1,$XPREC             IS PREC LONG ?
    0AA8 C0 90 0E22            4426        BF    LVI320                    NO, INIT ARRAYS
                               4427 *
                               4428 * CHANGE PRECISION SENSITIVE INSTRUCTIONS TO LONG PRECISION LENGTHS
                               4429 *
    0AAC 3C 35 0CEC            4430        MVI   LVI065+@Q,B@LILP*B@NICN-1    LNG OF CON TO MOVE TO VM
    0AB0 3C 48 0CED            4431        MVI   LVI065+@D1,B@LILP*B@NICN-1+B@LCRV  DISP OF CADDR
    0AB4 0C 01 0CEF 1375       4432        MVC   LVI065+@DOP2(@CADDR),LVIALC  SET ADDR OF CON TO BE USED
    0ABA 3C 08 0CFF            4433        MVI   LVI078+@Q,B@LILP*B@NIVR-1    LNG OF VAR TO MOVE TO VM
    0ABE 3C 08 0D00            4434        MVI   LVI078+@D1,B@LILP*B@NIVR-1   SET DISP OF VARS IN BFR
    0AC2 0C 01 0D02 1377       4435        MVC   LVI078+@DOP2(@CADDR),LVILAV  SET ADDR OF VAL TO MOVE
    0AC8 3C 08 0D5B            4436        MVI   LVI240+@Q,LVILUP          BYTES TO MOVE IN LONG PREC
    0ACC 3C 08 0D5F            4437        MVI   LVI242+@Q,B@LILP-1        SET VALUE LNG TO LONG
    0AD0 3C 08 0E2D            4438        MVI   LVI322+@Q,LVILUP          BYTES TO MOVE TO LONG PREC LNG
    0AD4 3C 09 0E31            4439        MVI   LVI326+@Q,B@LILP          SET VALUE LNG TO LONG
    0AD8 3C 08 0E34            4440        MVI   LVI328+@Q,B@LILP-1        INIT MOVE PT WITH LONG LNG
    0ADC 3C 08 0F11            4441        MVI   LVI482+@Q,LVILUP          BYTES TO MOVE TO LONG PREC LNG
    0AE0 3C 09 0F15            4442        MVI   LVI484+@Q,B@LILP          SET VALUE LNG TO LONG
    0AE4 3C 08 0F18            4443        MVI   LVI486+@Q,B@LILP-1        INIT MOVE PT WITH LONG LNG
    0AE8 7C 20 AF              4444        MVI   LVISPS(,@BR),LVILTF       SET LONG PREC STATUS BIT
    0AEB 7C 00 B3              4445        MVI   LVISPM(,@BR),LVINUL       0 SHORT PREC EXPONENT
    0AEE 5C 01 E6 58           4446        MVC   LVIAIV(@CADDR,@BR),LVILAV(,@BR)  SET VAL ADDR PARAM LONG
    0AF2 3C 08 10C1            4447        MVI   LVI738+@Q,B@LILP-1        SET EL LNG TO LONG PREC
    0AF6 C0 87 0E22            4448        B     LVI320                    INIT ARRAYS

                               4450 ***********************************************************************
                               4451 * ARITHMETIC VARIABLE PROCESSING ROUTINE
                               4452 ***********************************************************************
                               4453 *
                               4454 * SAVE POINTERS
                               4455 *
    0AFA 34 02 0B3B            4456 LVI080 ST    LVI098+@OP1,@XR           SAVE PT
    0AFE 74 02 39              4457        ST    LVI996+@OP1(,@BR),@XR     SAVE PT IN ERROR RTN
                               4458 *
                               4459 * SET POINTERS AND TEST FOR LETTER MATCH IN THE ALPHA TABLE
                               4460 *
    0B01 3C 00 0B23            4461        MVI   LVITT1,LVINUL             SET PT TO 0
    0B05 C2 02 141D            4462        LA    LVIATL,@XR                ADDR ALPHA TABLE
    0B09 2D 00 13F7 00         4463 LVI090 CLC   LVILSA,LVI0TD(LVIBYC,@XR) LETTERS MATCH ?
    0B0E F2 81 0C              4464        JE    LVI092                    YES, CHECK IF LETTER USED
                               4465 *
                               4466 * INCREMENT POINTERS
                               4467 *
    0B11 76 02 48              4468        A     LVIH01(,@BR),@XR          INCR TO NEXT ENTRY
    0B14 1E 00 0B23 4A         4469        ALC   LVITT1(LVIBYC),LVIH02(,@BR)  INCR TO NEXT ENTRY
    0B19 C0 87 0B09            4470        B     LVI090                    RECYCLE LOOP
                               4471 *
                               4472 * TEST IF VARIABLE USED IN BASIC PROGRAM - IF YES, SET TRACE SWITCH
                               4473 *
    0B1D C2 02 1A0C            4474 LVI092 LA    LVILVT,@XR                BASE ADDR ARITH VAR TBL
    0B21 E2 02 00              4475 LVI094 LA    *-*(,@XR),@XR             ADD DISP TO CORRECT ENTRY
    0B24 9D 01 01 46           4476 LVI095 CLC   LVI1TD(@VADDR,@XR),LVIH00(,@BR)  IS TBL ENTRY NULL
    0B28 D0 81 10              4477        BE    LVI990(,@BR)              YES, GO TO ERROR RTN
    0B2B C2 02 143B            4478        LA    LVIPTL+1,@XR              PT EQ TRACE TBL ADDR
    0B2F 0C 00 0B37 0B23       4479        MVC   LVI096+@D1,LVITT1(LVIBYC)  SET TRACE TBL ENTRY DISP
    0B35 BA 20 00              4480 LVI096 SBN   *-*(,@XR),LVILVM          SET TRACE BIT ON
                               4481 *
                               4482 * RESTORE POINTERS AND RETURN
                               4483 *
    0B38 C2 02 0000            4484 LVI098 LA    *-*,@XR                   RESTORE PT
    0B3C 76 02 48              4485        A     LVIH01(,@BR),@XR          INCR PT TO DELIMITER
    0B3F C0 87 0A92            4486        B     LVI040                    RETURN TO CALLING PROG

                               4488 ***********************************************************************
                               4489 * ARITHMETIC LETTERB7MVATIBTE PROCESSING ROUTINE
                               4490 ***********************************************************************
                               4491 *
                               4492 * SAVE POINTERS
                               4493 *
    0B43 34 02 0BB1            4494 LVI100 ST    LVI109+@OP1,@XR           SAVE PT
    0B47 74 02 39              4495        ST    LVI996+@OP1(,@BR),@XR     SAVE DT IN ERROR RTN
                               4496 *
                               4497 * SET SUBROUTINE POINTERS AND CHECK FOR LETTER MATCH IN ALPHA TABLE
                               4498 *
    0B4A C2 02 141D            4499        LA    LVIATL,@XR                ADDR ALPHA TBL
    0B4E 5F 01 DE DE           4500        SLC   LVILDP(,@BR),LVILDP(LVIBY2,@BR)  ZERO PT
    0B52 6D 00 D8 00           4501 LVI103 CLC   LVILSA(LVIBYC,@BR),LVI0TD(,@XR)  LETTERS MATCH ?
    0B56 F2 81 0B              4502        JE    LVI105                    YES, DETERMINE DISP TO TBL
                               4503 *
                               4504 * INCREMENT POINTERS
                               4505 *
    0B59 76 02 48              4506        A     LVIH01(,@BR),@XR          INCR TO NEXT LETTER
    0B5C 5E 01 DE 4A           4507        ALC   LVILDP(LVIBY2,@BR),LVIH02(,@BR)  INCR CTR
    0B60 C0 87 0B52            4508        B     LVI103                    LOOP UNTIL LETTER IS MATCHED
                               4509 *
                               4510 * CALULATE LETTER-DIGIT POINTER AND CHECK FOR DEFINITION
                               4511 *
    0B64 C2 02 1A46            4512 LVI105 LA    LVILDT,@XR                LETTER-DIGIT TBL BASE ADDR
    0B68 1C 00 0BAD DE         4513        MVC   LVI107+@D1(LVIBYC),LVILDP(,@BR)  SET TBL ENTRY DISP
    0B6D 5E 01 DE DE           4514        ALC   LVILDP(,@BR),LVILDP(LVIBY2,@BR)  MULTIPLY THE ALPHA TABLE
    0B71 76 02 DE              4515        A     LVILDP(,@BR),@XR          * INDEX BY TEN TO OBTAIN
    0B74 5E 01 DE DE           4516        ALC   LVILDP(,@BR),LVILDP(LVIBY2,@BR) * THE LETTERS INDEX
    0B78 5E 01 DE DE           4517        ALC   LVILDP(,@BR),LVILDP(LVIBY2,@BR) * PLUS TWO TIMES THE
    0B7C 76 02 DE              4518        A     LVILDP(,@BR),@XR          * DIGIT
    0B7F 7B F0 E0              4519        SBF   LVIDSA(,@BR),LVIDNM       SET ZONE BITS TO 0
    0B82 76 02 E0              4520        A     LVIDSA(,@BR),@XR          ADD THE DIGIT TO THE PT TWICE
    0B85 76 02 E0              4521        A     LVIDSA(,@BR),@XR          * TO ACCESS CORRECT ENTRY
    0B88 9D 01 01 46           4522        CLC   LVI1TD(@VADDR,@XR),LVIH00(,@BR)  IS TBL ENTRY NULL
    0B8C D0 81 10              4523        BE    LVI990(,@BR)              YES, GO TO ERROR RTN
                               4524 *
                               4525 * SET TRACE SWITCH IN TRACE TABLE
                               4526 *
    0B8F D2 02 CB              4527        LA    LVITM0(,@BR),@XR          ADDR TRACE MASK AREA
    0B92 76 02 E0              4528        A     LVIDSA(,@BR),@XR          INCR TO NEEDED MASK
    0B95 2C 00 0BAC 00         4529        MVC   LVI107+@Q(LVIBYC),LVI0TD(,@XR)  TRACE TBL ENTRY MASK
    0B9A 7D 08 E0              4530        CLI   LVIDSA(,@BR),LVIBDC       MASK THE 1ST ENTRY BYTE
    0B9D F2 82 07              4531        JL    LVI106                    YES, DECR PT TO 1ST ENTRY BYTE
    0BA0 C2 02 143B            4532        LA    LVIPTL+1,@XR              PT EO TRACE TBL ADDR
    0BA4 F2 87 04              4533        J     LVI107                    SKIP NEXT INST
    0BA7 C2 02 143A            4534 LVI106 LA    LVIPTL,@XR                SET TRACE TBL PT
    0BAB BA 00 00              4535 LVI107 SBN   *-*(,@XR),*-*             SET TRACE BIT ON
                               4536 *
                               4537 * RESTORE POINTERS AND RETURN
                               4538 *
    0BAE C2 02 0000            4539 LVI109 LA    *-*,@XR                   RESTORE PT
    0BB2 76 02 4A              4540        A     LVIH02(,@BR),@XR          INCR PT TO DELIMITER
    0BB5 C0 87 0A92            4541        B     LVI040                    RETURN TO CALLING PROGRAM

                               4543 ***********************************************************************
                               4544 * ROUTINE TO SCAN PAST BLANKS AND COMMAS IN THE REFERENCE LIST
                               4545 ***********************************************************************
                               4546 *
    0BB9 34 08 0BC9            4547 LVI110 ST    LVI114+@OP1,@ARR          SAVE RETURN ADDR
    0BBD BD 40 00              4548 LVI112 CLI   LVITD0(,@XR),B@BLNK       IS BYTE A BLANK ?
    0BC0 F2 81 07              4549        JE    LVI116                    YES, INCR PAST IT
    0BC3 BD 6B 00              4550        CLI   LVITD0(,@XR),B@CMMA       IS BYTE A COMMA ?
    0BC6 C0 01 0000            4551 LVI114 BNE   *-*                       NO, RETURN
    0BCA 76 02 48              4552 LVI116 A     LVIH01(,@BR),@XR          INCR TO NEXT BYTE
    0BCD C0 87 0BBD            4553        B     LVI112                    TEST NEXT BYTE

                               4555 ***********************************************************************
                               4556 * CHARACTER VARIABLE PROCESSING ROUTINE
                               4557 ***********************************************************************
                               4558 *
                               4559 * SAVE POINTERS
                               4560 *
    0BD1 34 02 0C11            4561 LVI120 ST    LVI135+@OP1,@XR           SAVE PT
    0BD5 74 02 39              4562        ST    LVI996+@OP1(,@BR),@XR     SAVE PT IN ERROR RTN
                               4563 *
                               4564 * SET SUBROUTINE POINTERS AND CHECK FOR LETTER MATCH IN ALPHA TABLE
                               4565 *
    0BD8 C2 02 141D            4566        LA    LVIATL,@XR                ADDR ALPHA TBL
    0BDC 3C 00 0BF9            4567        MVI   LVITT2,LVINUL             SET PT TO 0
    0BE0 6D 00 D8 00           4568 LVI125 CLC   LVILSA(LVIBYC,@BR),LVI0TD(,@XR)  LETTERS MATCH ?
    0BE4 F2 81 0C              4569        JE    LVI130                    YES, DETERMINE DISP TO TBL
                               4570 *
                               4571 * INCREMENT POINTERS
                               4572 *
    0BE7 76 02 48              4573        A     LVIH01(,@BR),@XR          INCR TO NEXT LETTER
    0BEA 1E 00 0BF9 4A         4574        ALC   LVITT2(LVIBYC),LVIH02(,@BR)  INCR CTR
    0BEF C0 87 0BE0            4575        B     LVI125                    LOOP UNTIL LETTER IS MATCHED
                               4576 *
                               4577 * CALCULATE CHARACTER VARIABLE TABLE POINTER AND CHECK FOR DEFINITION
                               4578 *
    0BF3 C2 02 1C8A            4579 LVI130 LA    LVICVT,@XR                BASE ADDR CHAR VAR TBL
    0BF7 E2 02 00              4580 LVI132 LA    *-*(,@XR),@XR             ADD DISP TO CORRECT ENTRY
    0BFA 9D 01 01 46           4581        CLC   LVI1TD(@VADDR,@XR),LVIH00(,@BR)  IS TBL ENTRY NULL ?
    0BFE D0 81 10              4582        BE    LVI990(,@BR)              YES, GO TO ERROR RTN
                               4583 *
                               4584 * SET TRACE SWITCH IN TRACE TABLE
                               4585 *
    0C01 C2 02 143B            4586        LA    LVIPTL+1,@XR              PT EQ TRACE TBL ADDR
    0C05 0C 00 0C0D 0BF9       4587        MVC   LVI134+@D1,LVITT2(LVIBYC) SET TRACE TBL ENTRY DISP
    0C0B BA 10 00              4588 LVI134 SBN   *-*(,@XR),LVICVM          SET TRACE BIT ON
                               4589 *
                               4590 * RESTORE POINTER AND RETURN
                               4591 *
    0C0E C2 02 0000            4592 LVI135 LA    *-*,@XR                   RESTORE PT
    0C12 76 02 4A              4593        A     LVIH02(,@BR),@XR          INCR TO DELIMITER
    0C15 C0 87 0A92            4594        B     LVI040                    RETURN TO CALLING PROGRAM

                               4596 ***********************************************************************
                               4597 * ARITHMETIC ARRAY VARIABLE PROCESSING ROUTINZ
                               4598 ***********************************************************************
                               4599 *
                               4600 * SAVE POINTER AND SET SUBROUTINE POINTERS
                               4601 *
    0C19 34 02 0C4C            4602 LVI140 ST    LVI155+@OP1,@XR           SAVE POINTER
    0C1D 74 02 39              4603        ST    LVI996+@OP1(,@BR),@XR    SAVE 0T IN ERROR RTN
    0C20 C2 02 141D            4604        LA    LVIATL,@XR                ADDR ALPHA TBL
    0C24 3C 00 0C41            4605        MVI   LVITT3,LVINUL             SET PT TO 0
                               4606 *
                               4607 * TEST FOR LETTER MATCH IN ALPHA TABLE
                               4608 *
    0C28 6D 00 D8 00           4609 LVI145 CLC   LVILSA(LVIBYC,@BR),LVI0TD(,@XR) LETTERS MATCH ?
    0C2C F2 81 0C              4610        JE    LVI150                    YES, CHECK FOR DEFINITION
                               4611 *
                               4612 * INCREMENT POINTERS
                               4613 *
    0C2F 76 02 48              4614        A     LVIH01(,@BR),@XR          INCR TO NEXT LETTER
    0C32 1E 00 0C41 4A         4615        ALC   LVITT3(LVIBYC),LVIH02(,@BR)  INCR CTR
    0C37 C0 87 0C28            4616        B     LVI145                    LOOP UNTIL LETTERS MATCH
                               4617 *
                               4618 * CALCULATE NUMERIC ARRAY TABLE POINTER AND CHECK FOR DEFINITION
                               4619 *
    0C3B C2 02 1CC4            4620 LVI150 LA    LVINAT,@XR                BASE ADDR NUN ARRAY TBL
    0C3F E2 02 00              4621 LVI153 LA    *-*(,@XR),@XR             ADD DISP TO CORRECT ENTRY
    0C42 9D 01 01 46           4622        CLC   LVI1TD(@VADDR,@XR),LVIH00(,@BR) IS TBL ENTRY NULL
    0C46 D0 81 1E              4623        BE    LVI992(,@BR)
                               4624 *
                               4625 * DETERMINE MASK TO SET IN TRACE TAKE
                               4626 *
    0C49 C2 02 0000            4627 LVI155 LA    *-*,@XR                   RESTORE PT
    0C4D BD F0 02              4628        CLI   LVITD2(,@XR),B@DEC0       IS BYTE A DIGIT ?
    0C50 0C 00 0C80 0C41       4629        MVC   LVI165+@D1,LVITT3(LVIBYC) SET TRACE TBL ENTRY DISP
    0C56 F2 82 15              4630        JL    LVI159                    NO, MASK ALL BIT
    0C59 3C 04 0C7F            4631        MVI   LVI165+@Q,LVIAAP          SET MASK TO USE A PARTIAL
    0C5D 3C 01 0FD8            4632        MVI   LVI675+@Q,LVISWO          SET ELEMENT INIT SW ON
    0C61 BD 5D 00              4633 LVI157 CLI   LVITD0(,@XR),B@RPAR       A RIGHT PAREN ?
    0C64 F2 81 0F              4634        JE    LVI160                    SET MASK
    0C67 76 02 48              4635        A     LVIH01(,@BR),@XR          INCR PT TO NEXT BYTE
    0C6A C0 87 0C61            4636        B     LVI157                    LOOP UNTIL RT PAREN
    0C6E 3C 08 0C7F            4637 LVI159 MVI   LVI165+@Q,LVIAAA          SET ALL MASK TO BE USED
    0C72 C0 87 0C61            4638        B     LVI157                    INCR IFIR TO-DEINITEP
                               4639 *
                               4640 * SET TRACE SWITCH IN TRACE TABLE
                               4641 *
    0C76 34 02 0C84            4642 LVI160 ST    LVI167+@OP1,@XR           SAVE PT
    0C7A C2 02 143B            4643        LA    LVIPTL+1,@XR              PT EQ TRACE TBL ADDR
    0C7E BA 00 00              4644 LVI165 SBN   *-*(,@XR),*-*             SET TRACE BIT ON
                               4645 *
                               4646 * RESTORE POINTER AND RETURN
                               4647 *
    0C81 C2 02 0000            4648 LVI167 LA    *-*,@XR                   RESTORE PT
    0C85 76 02 48              4649        A     LVIH01(,@BR),@XR R        INCR TO DELIMETER
    0C88 C0 87 0A92            4650        B     LVI040                    RETURN

                               4652 ***********************************************************************
                               4653 * CHARACTER ARRAY VARIABLE PROCESSINS ROUTINE
                               4654 ***********************************************************************
                               4655 *
                               4656 * SAVE POINTER AND SET SUBROUTINE POINTERS
                               4657 *
    0C8C 34 02 0CBF            4658 LVI170 ST    LVI185+@OP1,@XR           SAVE PT
    0C90 74 02 39              4659        ST    LVI996+@OP1(,@BR),@XR     SAVE PT IN ERROR RTN
    0C93 C2 02 141D            4660        LA    LVIATL,@XR                ADDR ALPHA TBL
    0C97 3C 00 0CB4            4661        MVI   LVITT4,LVINUL             SET PT TO 0
                               4662 *
                               4663 * TEST FOR LETTER MATCH IN ALPHA TABLE
                               4664 *
    0C9B 6D 00 D8 00           4665 LVI175 CLC   LVILSA(LVIBYC,@BR),LVI0TD(,@XR) LETTERS MATCH
    0C9F F2 81 0C              4666        JE    LVI180                    YES, CHECK FOR DEFINITION
                               4667 *
                               4668 * INCREMENT POINTERS
                               4669 *
    0CA2 76 02 48              4670        A     LVIH01(,@BR),@XR          INCR TO NEXT LETTER
    0CA5 1E 00 0CB4 4A         4671        ALC   LVITT4(LVIBYC),LVIH02(,@BR)  INCR CTR
    0CAA C0 87 0C9B            4672        B     LVI175                    LOOP UNTIL LETTERS MATCH
                               4673 *
                               4674 * CALCULATE CHARACTER ARRAY TABLE POINTER AND CHECK FOR DEFINITION
                               4675 *
    0CAE C2 02 1CFE            4676 LVI180 LA    LVICAT,@XR                BASE ADDR CHAR ARRAY TBL
    0CB2 E2 02 00              4677 LVI182 LA    *-*(,@XR),@XR             ADD DISP TO CORRECT ENTRY
    0CB5 9D 01 01 46           4678        CLC   LVI1TD(@VADDR,@XR),LVIH00(,@BR) IS TBL ENTRY NULL
    0CB9 D0 81 1E              4679        BE    LVI992(,@BR)
                               4680 *
                               4681 * DETERMINE MASK TO SET IN TRACE TABLE
                               4682 *
    0CBC C2 02 0000            4683 LVI185 LA    *-*,@XR                   RESTORE PT
    0CC0 BD F0 03              4684        CLI   LVITD3(,@XR),B@DEC0       IS BYTE A DIGIT ?
    0CC3 0C 00 0C80 0CB4       4685        MVC   LVI165+@D1,LVITT4(LVIBYC) SET TRA1 TBL ENTRY DIP
    0CC9 F2 82 0B              4686        JL    LVI187                    NO, MASK ALL BIT
    0CCC 3C 01 0C7F            4687        MVI   LVI165+@Q,LVICAP          SET PARTIAL MASK TO BE USED
    0CD0 3C 01 1111            4688        MVI   LVI760+@Q,LVISWO          SET ELEMENT INIT SW ON
    0CD4 F2 87 04              4689        J     LVI188                    INCR TO RIGHT PAREN
    0CD7 3C 02 0C7F            4690 LVI187 MVI   LVI165+@Q,LVICAA          SET ALL MASK TO BE USED
    0CDB C0 87 0C61            4691 LVI188 B     LVI157                    INCR PT TO DELIMITER
```

```
                               4693 ***********************************************************************
                               4694 *                                                                     *
                               4695 *      PLACE INTERNAL CONSTANTS AND VARIBLES INTO MEMORY              *
                               4696 *                                                                     *
                               4697 ***********************************************************************
                               4698 *
                               4699 * PLACE INTERNAL CONSTANTS IN VIRTUAL MEMORY
                               4700 *
    0CDF 4C 00 F3 1A08         4701 LVI060 MVC   LVIINN(,@BR),LVIICP       SET INT CONS PG NO.
    0CE4 C2 02 0700            4702        LA    LVIBF2,@XR                CADDIR I/O BFR
    0CE8 BC 40 00              4703        MVI   @ZERO(,@XR),@BLANK                                     1-5
    0CEB 8C 00 00 0000         4704 LVI065 MVC   *-*(@VQ,@XR),*-*          MOVE THE INTERNAL CONSTANTS
    0CEC                       4705        ORG   LVI065+@Q                 * TO THE VIRTUAL MEMORY
    0CEC 1D               0CEC 4706        DC    AL1(B@LISP*B@NICN-1)      * BUFFER, THE DISPLACEMENT
    0CED                       4707        ORG   LVI065+@D1                * AND CONSTANTS MOVED ARE
    0CED 30               0CED 4708        DC    AL1(B@LISP*B@NICN-1+B@LCRV)  * DEPENDENT ON PROGRAM    1-4
    0CEE 1397             0CEF 4709        DC    AL2(LVIASC)               * PRECISION
                               4710 *
                               4711 * TEST FOR INTERNAL VARIABLES
                               4712 *
    0CF0 3D 00 0001            4713 LVI070 CLI   B@NIVR,LVINUL             ANY INTERNAL VARS ?
    0CF4 C0 81 0D03            4714        BE    LVI200                    NO, INIT VARS
                               4715 *
                               4716 * MOVE INTERNAL VARIABLES TO VIRTUAL MEMORY
                               4717 *
    0CF8 0E 00 0D00 1A0B       4718 LVI075 ALC   LVI078+@D1(1),LVIIVD      SET DISP TO 1ST VAR
    0CFE 8C 00 00 0000         4719 LVI078 MVC   *-*(@VQ,@XR),*-*          MOVE THE INTERNAL CONSTANT(S)
    0CFF                       4720        ORG   LVI078+@Q                 * TO VM BUFFER, THE DISP AND
    0CFF 04               0CFF 4721        DC    AL1(B@LISP*B@NIVR-1)      * VARIABLE(S) ARE MOVED
    0D00 04               0D00 4722        DC    AL1(B@LISP*B@NIVR-1)      * DEPENDENT ON THE PROGRAM
    0D01 13D2             0D02 4723        DC    AL2(LVISPM)               * PRECISION

                               4725 ***********************************************************************
                               4726 * CHARACTER VARIABLE INITIALIZATION ROUTINE
                               4727 ***********************************************************************
                               4728 *
                               4729 * SET PARAMETERS FOR THE PUT ROUTINE
                               4730 *
    0D03 3C 12 12DC            4731 LVI200 MVI   LVI933+@D1,B@LCRV-1       1ST ENTRY DISP IN BFR
    0D07 3C 12 12DE            4732        MVI   LVI935+@Q,B@LCRV-1        BYTES IN CHAR VARIABLE
    0D0B 1C 01 12E1 5A         4733        MVC   LVI935+@DOP2(@CADDR),LVICHV(,@BR)  ADDR INIT VALUE
                               4734 *
                               4735 * SET CHARACTER VARIABLE TABLE POINTER AND TEST FOR DEFINITION
                               4736 *
    0D10 C2 02 1C8A            4737 LVI205 LA    LVICVT,@XR                ADDR CHAR VAR TBL
    0D14 E2 02 00              4738 LVI210 LA    *-*(,@XR),@XR             INCR POINTER TO THE LAST
    0D16                       4739        ORG   LVI210+@D1                * UNTESTED ENTRY, INITIALLY
    0D16 38               0D16 4740        DC    XL1'38'                   * SET TO THE LAST ENTRY
    0D17                       4741        ORG
    0D17 9D 01 01 46           4742        CLC   LVI1TD(,@XR),LVIH00(@CADDR,@BR) IS ENTRY DEFINED ?
    0D1B F2 01 0C              4743        JNE   LVI220                    YES, CHECK TRACE SWITCH
    0D1E 1F 00 0D16 4A         4744 LVI215 SLC   LVI210+@D1(LVIBYC),LVIH02(,@BR)  DECR TO NEXT ENTRY
    0D23 C0 02 0D10            4745        BNM   LVI205                    LOOP UNTIL DISP IS 0
    0D27 F2 87 30              4746        J     LVI240                    INIT ARITH VAR
                               4747 *
                               4748 * CHECK TRACE SWITCH
                               4749 *
    0D2A 2C 00 12DF 01         4750 LVI220 MVC   LVI935+@D1,LVI1TD(LVIBYC,@XR)  SET VALUE DISP IN PG
    0D2F 6C 00 D7 00           4751        MVC   LVIHLD(LVIBYC,@BR),LVI0TD(,@XR)  SET VALUE PG TO SAVE
    0D33 0C 00 0D3F 0D16       4752        MVC   LVI225+@D1,LVI210+@D1(LVIBYC)  SET DISP TO TRACE TBL BYTE
    0D39 C2 02 143A            4753        LA    LVIPTL,@XR                ADDR TRACE TBL
    0D3D E2 02 00              4754 LVI225 LA    *-*(,@XR),@XR             ADD DISP TO INDEX PT
    0D40 B8 10 01              4755        TBN   LVI1TD(,@XR),LVICVM       IS TRACE BIT ON
    0D43 F2 90 06              4756        JF    LVI230                    NO, SET STATUS BIT OFF
                               4757 *
                               4758 * SET STATUS BYTE IN INITIALIZATION VALUE AND BRANCH TO PUT ROUTINE
                               4759 *
    0D46 7C C0 B8              4760        MVI   LVICSB(,@BR),LVICTN       SET STATUS BYTE FOR TRACE
    0D49 F2 87 03              4761        J     LVI235                    GO PUT VALUE TO VM
    0D4C 7C 40 B8              4762 LVI230 MVI   LVICSB(,@BR),LVICTF       SET STATUS BYTE TO NOT TRACE
    0D4F C0 87 12AF            4763 LVI235 B     LVI900                    MOVE VALUE TO VM
    0D53 7C 01 D5              4764        MVI   LVITSW(,@BR),LVISWO       SET SCALAR SW ON
    0D56 C0 87 0D1E            4765        B     LVI215                    CHECK NEXT TBL ENTRY

                               4767 ***********************************************************************
                               4768 * ARITHMETIC LETTER VARIABLE INITIALIZATION ROUTINE
                               4769 ***********************************************************************
                               4770 *
                               4771 * SET PARAMETERS FOR THE PUT ROUTINE
                               4772 *
    0D5A 3C 00 12DE            4773 LVI240 MVI   LVI935+@Q,*-*             SET THE NUMBER OF BYTES
    0D5B                       4774        ORG   LVI240+@Q                 * TO MOVE IN THE PUT S@BROUTINE
    0D5B 04               0D5B 4775        DC    AL1(B@LISP-1)             * INITIALLY SHORT PREC LNG-1
    0D5E                       4776        ORG
    0D5E 3C 00 12DC            4777 LVI242 MVI   LVI933+@D1,*-*            SET THE DISP TO ISI ENTRY IN
    0D5F                       4778        ORG   LVI242+@Q                 * THE BFR, INITIALLY
    0D5F 04               0D5F 4779        DC    AL1(B@LISP-1)             * SET TO LNG OF SHORT PREC VAL
    0D62                       4780        ORG
    0D62 1C 01 12E1 E6         4781        MVC   LVI935+@DOP2,LVIAIV(@CADDR,@BR) SET ADDR ARITH VALUE
                               4782 *
                               4783 * SET THE LETTER VARIABLE TABLE POINTER AND CHECK FOR DEFINITION
                               4784 *
    0D67 C2 02 1A0C            4785 LVI245 LA    LVILVT,@XR                ADDR LETTER VAR TBL
    0D6B E2 02 00              4786 LVI250 LA    *-*(,@XR),@XR             INCR POINTER TO THE LAST
    0D6D                       4787        ORG   LVI250+@D1                * UNTESTED ENTRY, INITIALLY
    0D6D 38               0D6D 4788        DC    XL1'38'                   * SET TO LAST ENTRY
    0D6E                       4789        ORG
    0D6E 9D 01 01 46           4790        CLC   LVI1TD(,@XR),LVIH00(@CADDR,@BR)  IS ENTRY NULL ?
    0D72 F2 01 0C              4791        JNE   LVI260                    NO, CHECK TRACE SWITCH
    0D75 1F 00 0D6D 4A         4792 LVI255 SLC   LVI250+@D1(LVIBYC),LVIH02(,@BR)  DECR DISP TO NEXT ENTRY
    0D7A C0 02 0D67            4793        BNM   LVI245                    LOOP UNTIL DISP IS 0
    0D7E F2 87 30              4794        J     LVI280                    INIT LETTER-DIGIT VAR
                               4795 *
                               4796 * CHECK TRACE SWITCH
                               4797 *
    0D81 2C 00 12DF 01         4798 LVI260 MVC   LVI935+@D1,LVI1TD(LVIBYC,@XR)  VALUE DISP IN PG
    0D86 6C 00 D7 00           4799        MVC   LVIHLD(LVIBYC,@BR),LVI0TD(,@XR) VALUE PG TO SAVE
    0D8A 0C 00 0D96 0D6D       4800        MVC   LVI265+@D1,LVI250+@D1(LVIBYC) SET DISP TO TRACE TBL BYTE
    0D90 C2 02 143A            4801        LA    LVIPTL,@XR                ADDR TRACE TBL
    0D94 E2 02 00              4802 LVI265 LA    *-*(,@XR),@XR             ADD DISP TO INDEX PT
    0D97 B8 20 01              4803        TBN   LVI1TD(,@XR),LVILVM       IS TRACE BIT ON ?
    0D9A F2 90 06              4804        JF    LVI269                    NO, SET STATUS BIT OFF
```

```
                               4805 *
                               4806 * SET STATUS BYTE IN INITIALIZATION VALUE AND BRANCH TO PUT ROUTINE
                               4807 *
    0D9D 7A 80 AF              4808        SBN   LVISPS(,@BR),LVITMK       SET STATUS BYTE ON
    0DA0 F2 87 03              4809        J     LVI270                    PUT VALUE TO VM
    0DA3 7B 80 AF              4810 LVI269 SBF   LVISPS(,@BR),LVITMK       SET VALUE TRACE BIT OFF
    0DA6 C0 87 12AF            4811 LVI270 B     LVI900                    MOVE VALUE TO VM
    0DAA 7C 01 D5              4812        MVI   LVITSW(,@BR),LVISWO       SET SCALAR SW ON
    0DAD C0 87 0D75            4813        B     LVI255                    CHECK NEXT TBL ENTRY

                               4815 ***********************************************************************
                               4816 * ARITHMETIC LETTER-DIGIT VARIABLE INITIALIZATION ROUTINE
                               4817 ***********************************************************************
                               4818 *
                               4819 * SET LETTER-DIGIT TABLE POINTER AND TEST FOR DEFINITION
                               4820 *
    0DB1 C2 02 0000            4821 LVI280 LA    *-*,@XR                   SET POINTER TO LAST LETTER
    0DB3                       4822        ORG   LVI280+@D1                * DIGIT TBL ENTRY NOT TESTED,
    0DB3 1C88             0DB4 4823        DC    AL2(LVILET)               * INITIALLY THE LAST ENTRY
    0DB5                       4824        ORG
    0DB5 9D 01 01 46           4825        CLC   LVI1TD(@CADDR,@XR),LVIH00(,@BR)  IS ENTRY NULL ?
    0DB9 F2 01 1F              4826        JNE   LVI290                    NO, CHECK TRACE BIT
                               4827 *
                               4828 * MODIFY POINTERS
                               4829 *
    0DBC 1F 01 0DB4 4A         4830 LVI285 SLC   LVI280+@OP1,LVIH02(LVIBY2,@BR)  DECR LDT PT
    0DC1 1F 00 0DE9 48         4831        SLC   LVI292+@D1,LVIH01(1,@BR)  DECR DECIMAL PT
    0DC6 C0 02 0DB1            4832        BNL   LVI280                    LOOP UNTIL COUNT IS LT 0
    0DCA 3C 09 0DE9            4833        MVI   LVI292+@D1,LVIDPT         RESET DIGIT PT
    0DCE 1F 00 0DF5 4A         4834        SLC   LVI294+@D1,LVIH02(1,@BR)  DECR TRACE ENTRY PT
    0DD3 C0 82 0FD0            4835        BL    LVI670                    INIT ARRAY ELEMENTS
    0DD7 C0 87 0DB1            4836        B     LVI280                    LOOP UNTIL ALL ENTRIES PROC
                               4837 *
                               4838 * CHECK TRACE SWITCH
                               4839 *
    0DDB 2C 00 12DF 01         4840 LVI290 MVC   LVI935+@D1,LVI1TD(1,@XR)  SET VALUE DISP IN PG
    0DE0 6C 00 D7 00           4841        MVC   LVIHLD(LVIBYC,@BR),LVI0TD(,@XR)  SET VALUE PG TO SAVE
    0DE4 D2 02 CB              4842        LA    LVITM0(,@BR),@XR          ADDR LETTER-DIGIT MASK AREA
    0DE7 E2 02 00              4843 LVI292 LA    *-*(,@XR),@XR             INCREMENT BY THE DIGIT VALUE
    0DE9                       4844        ORG   LVI292+@D1                * TO OBTAIN THE PROPER MASK
    0DE9 09               0DE9 4845        DC    XL1'09'                   * TO CHECK FOR TRACE BIT ON
    0DEA                       4846        ORG
    0DEA 2C 00 0E09 00         4847        MVC   LVI298+@Q,LVI0TD(1,@XR)   SET Q CODE FOR TRACE MASK
    0DEF C2 02 143A            4848        LA    LVIPTL,@XR                SET POINTER TO PROPER ENTRY
    0DF3 E2 02 00              4849 LVI294 LA    *-*(,@XR),@XR             * IN TRACE TABLE, INITIALLY
    0DF5                       4850        ORG   LVI294+@D1                * SET TO THE 1ST BYTE IN THE
    0DF5 38               0DF5 4851        DC    XL1'38'                   * LAST ENTRY
    0DF6                       4852        ORG
    0DF6 3D 08 0DE9            4853        CLI   LVI292+@D1,LVIBDC         IS TRACE BIT IN 1ST BYTE
    0DFA F2 02 07              4854        JNL   LVI296                    NO, SET DISP TO 1
    0DFD 3C 00 0E0A            4855        MVI   LVI298+@D1,LVI0TD         SET DISP TO 0 IN ENTRY
    0E01 F2 87 04              4856        J     LVI298                    GO TEST TRACE BIT
    0E04 3C 01 0E0A            4857 LVI296 MVI   LVI298+@D1,LVI1TD         SET DISP TO 1 IN ENTRY
    0E08 B8 00 00              4858 LVI298 TBN   *-*(,@XR),*-*             IS TRACE BIT ON
    0E0B F2 90 06              4859        JF    LVI305                    NO, GO SET STATUS BIT OFF
                               4860 *
                               4861 * SET STATUS BYTE IN INITIALIZATION VALUE AND BRANCH TO PUT ROUTINE
                               4862 *
    0E0E 7A 80 AF              4863 LVI300 SBN   LVISPS(,@BR),LVITMK       SET STATUS BIT ON
    0E11 F2 87 03              4864        J     LVI310                    BYPASS 1 INSTRUCTION
    0E14 7B 80 AF              4865 LVI305 SBF   LVISPS(,@BR),LVITMK       SET STATUS BIT OFF
    0E17 C0 87 12AF            4866 LVI310 B     LVI900                    MOVE VALUE TO VM
    0E1B 7C 01 D5              4867        MVI   LVITSW(,@BR),LVISWO       SET SCALAR SW ON
    0E1E C0 87 0DBC            4868        B     LVI285                    CHECK NEXT TBL ENTRY
```

```
                               4870 ***********************************************************************
                               4871 *                                                                     *
                               4872 *      ARRAY INITIALIZATION - REGION 1                                *
                               4873 *                                                                     *
                               4874 ***********************************************************************
                               4875 *
                               4876 ***********************************************************************
                               4877 * ARITHMETIC ARRAY INITIALIZATIONREGION 1
                               4878 ***********************************************************************
                               4879 *
                               4880 * SET INITIALIZATION PUT ROUTINE PARAMETERS
                               4881 *
    0E22 4C 00 FA 1A00         4882 LVI320 MVC   LVIPCT(1,@BR),LVIVA1      SET 1ST PG REGION 1
    0E27 1C 01 122B E6         4883        MVC   LVI810+@DOP2(@CADDR),LVIAIV(,@BR)  SET ADDR ARITH VALUE
    0E2C 3C 00 1228            4884 LVI322 MVI   LVI810+@Q,*-*             SET BYTES IN VALUE TO MOVE,
    0E2D                       4885        ORG   LVI322+@Q                 * INITIALLY SET TO SHORT
    0E2D 04               0E2D 4886        DC    AL1(B@LISP-1)             * PRECISION LENGTH
    0E30                       4887        ORG
    0E30 7C 00 D6              4888 LVI326 MVI   LVIPLN(,@BR),*-*          SET LENGTH OF VALUE
    0E31                       4889        ORG   LVI326+@Q                 * INITIALLY SET TO
    0E31 05               0E31 4890        DC    AL1(B@LISP)               * SHORT VALUE LNG
    0E33                       4891        ORG
    0E33 7C 00 DA              4892 LVI328 MVI   LVICNT(,@BR),*-*          SET THE DISP TO THE LAST
    0E34                       4893        ORG   LVI328+@Q                 * BYTE OF THE INIT VALUE
    0E34 04               0E34 4894        DC    AL1(B@LISP-1)             * TO INIT THE MOVE PT
    0E36                       4895        ORG
                               4896 *
                               4897 * SELECT ARRAY SYMBOL TABLE ENTRY AND TEST FOR DEFINITION
                               4898 *
    0E36 C2 02 1CC4            4899 LVI330 LA    LVINAT,@XR                ADDR ARITH SYMBOL
    0E3A B5 02 00              4900 LVI335 L     *-*(,@XR),@XR             LOAD DOPE VECTOR VADDR FORM
    0E3C                       4901        ORG   LVI335+@D1                * ARITH SRM TBL, INITIALLY SET
    0E3C 39               0E3C 4902        DC    AL1(B@LL12-1)             * WITH LAST TBL ELEMENT
    0E3D                       4903        ORG
    0E3D 7B 80 AF              4904        SBF   LVISPS(,@BR),LVITMK       SET TRACE BIT OFF
    0E40 76 02 46              4905        A     LVIH00(,@BR),@XR          IS ENTRY NULL
    0E43 F2 81 35              4906        JE    LVI370                    YES, DECR TO NEXT ENTRY
                               4907 *
                               4908 * TEST IF ARRAY IN REGION 1
                               4909 *
    0E46 76 02 52              4910 LVI336 A     LVIAAC(,@BR),@XR          CONVERT D/V VADDR TO CADDR
    0E49 2D 01 1A03 07         4911        CLC   LVIRG1(@CADDR),B@ABAS(,@XR)  IS ARRAY IN REGION 1 ?
    0E4E F2 84 08              4912        JH    LVI340                    YES, CHECK TRACE SWITCH
    0E51 3C 01 0F01            4913        MVI   LVI470+@Q,LVISWO          SET REGION 2 SW
    0E55 C0 87 0E7B            4914        B     LVI370                    DECR TO NEXT TBL ENTRY
                               4915 *
                               4916 * SET PUT PARAMETERS AND TEST TRACE SWITCH
                               4917 *
    0E59 6C 01 E4 05           4918 LVI340 MVC   LVIELC(B@LDMN,@BR),B@AMAX(,@XR)  SET NO. ELEMENTS
    0E5D 0C 00 0E69 0E3C       4919        MVC   LVI345+@D1(1),LVI335+@D1  SET ENTRY DISP TO INCR PT BY
    0E63 C2 02 143A            4920        LA    LVIPTL,@XR                ADDR TRACE TBL
    0E67 E2 02 00              4921 LVI345 LA    *-*(,@XR),@XR             ADD DISP TO TRACE TBL ENTRY
    0E6A B8 08 00              4922        TBN   LVI0TD(,@XR),LVIAAA       IS TRACE ALL SW ON ?
    0E6D F2 90 03              4923        JF    LVI360                    NO, GO INITIALIZE ARRAY
                               4924 *
                               4925 * SET STATUS BYTE IN INITIALIZATION VALUE
                               4926 *
    0E70 7A 80 AF              4927 LVI355 SBN   LVISPS(,@BR),LVITMK       SET TRACE BIT ON IN INIT VALUE
                               4928 *
                               4929 * INITIALIZE ARRAY AND SET PRINT SWITCH ON
                               4930 *
    0E73 C0 87 120F            4931 LVI360 B     LVI800                    INIT ARRAY
    0E77 3C 01 0EE7            4932        MVI   LVI450+@Q,LVISWO          SET PRINT SW ON
                               4933 *
                               4934 * DECREMENT ARRAY TABLE POINTER TO ACCESS NEXT TABLE ENTRY
                               4935 *
    0E7B 1F 00 0E3C 4A         4936 LVI370 SLC   LVI335+@D1(1),LVIH02(,@BR)  DECR PT
    0E80 C0 02 0E36            4937        BNM   LVI330                    PROCESS UNTIL LAST C,RY
    0E84 7B 80 AF              4938        SBF   LVISPS(,@BR),LVITMK       SET TRACE BIT OFF IN INIT VALUE

                               4940 ***********************************************************************
                               4941 * CHARACTER ARRAY INITIALIZATION - REGION 1
                               4942 ***********************************************************************
                               4943 *
                               4944 * SET INITIALIZATION PUT ROUTINE PARAMETERS
                               4945 *
    0E87 1C 01 122B 5A         4946 LVI380 MVC   LVI810+@DOP2(@CADDR),LVICHV(,@BR)  SET ADDR CHAR VALUE
    0E8C 3C 12 1228            4947        MVI   LVI810+@Q,B@LCRV-1        BYTES IN CHAR VALUE
    0E90 7C 13 D6              4948        MVI   LVIPLN(,@BR),B@LCRV       LENGTH OF CHAR VALUE
    0E93 7C 12 DA              4949        MVI   LVICNT(,@BR),B@LCRV-1     INIT DISP IN MOVE PT
                               4950 *
                               4951 * SELECT ARRAY SYMBOL TABLE ENTRY AND TEST FOR DEFINITION
                               4952 *
    0E96 C2 02 1CFE            4953 LVI390 LA    LVICAT,@XR                ADDR CHAR SYM TBL
    0E9A B5 02 00              4954 LVI395 L     *-*(,@XR),@XR             LOAD DOPE VECTOR VADDR FROM
    0E9C                       4955        ORG   LVI395+@D1                * CHAR SYM TBL, INITIALLY SET
    0E9C 39               0E9C 4956        DC    AL1(B@LL13-1)             * FOR LAST TBL ENTRY
    0E9D                       4957        ORG
    0E9D 76 02 46              4958        A     LVIH00(,@BR),@XR          IS ENTRY NULL
    0EA0 F2 81 3A              4959        JE    LVI440                    YES, DECR TO NEXT ENTRY
                               4960 *
                               4961 * TEST IF ARRAY IS IN REGION 1
                               4962 *
    0EA3 76 02 52              4963 LVI400 A     LVIAAC(,@BR),@XR          CONVERT D/V VADDR TO CADDR
    0EA6 2D 01 1A03 03         4964        CLC   LVIRG1(@CADDR),B@CBAS(,@XR)  IS ARRAY IN REGION 1
    0EAB F2 84 07              4965        JH    LVI410                    YES, CHECK TRACE SW
    0EAE 3C 01 0F64            4966        MVI   LVI550+@Q,LVISWO          SET CHAR REGION 2 SW ON
    0EB2 F2 87 28              4967        J     LVI440                    DECR TO NEXT ENTRY
                               4968 *
                               4969 * SET PUT PARAMETERS AND TEST TRACE SWITCH
                               4970 *
    0EB5 6C 01 E4 01           4971 LVI410 MVC   LVIELC(LVIBY2,@BR),B@CDMN(,@XR)  SET NO. ELEMENTS
    0EB9 0C 00 0EC5 0E9C       4972        MVC   LVI415+@D1(1),LVI395+@D1  SET ENTRY DISP TO INCR PT BY
    0EBF C2 02 143A            4973        LA    LVIPTL,@XR                ADDR TRACE TBL
    0EC3 E2 02 00              4974 LVI415 LA    *-*(,@XR),@XR             ADD DISP TO TRACE TBL ENTRY
    0EC6 B8 02 00              4975        TBN   LVI0TD(,@XR),LVICAA       IS TRACE ALL SW ON
    0EC9 F2 10 06              4976        JT    LVI425                    YES, SET TRACE BIT ON
                               4977 *
                               4978 * SET STATUS BYTE IN INITIALIZATION VALUE
                               4979 *
    0ECC 7C 40 B8              4980 LVI420 MVI   LVICSB(,@BR),LVICTF       SET TRACE BIT OFF IN INIT VALUE
    0ECF F2 87 03              4981        J     LVI430                    MOVE VALUES TO VM
    0ED2 7C C0 B8              4982 LVI425 MVI   LVICSB(,@BR),LVICTN       SET TRACE BIT ON IN INIT VALUE
                               4983 *
                               4984 * INITIALIZE ARRAY AND SET PRINT SWITCH ON
                               4985 *
    0ED5 C0 87 120F            4986 LVI430 B     LVI800                    INIT ARRAY
    0ED9 3C 01 0EE7            4987        MVI   LVI450+@Q,LVISWO          SET PRINT SW ON
                               4988 *
                               4989 * DECREMENT ARRAY TABLE POINTER TO ACCESS NEXT TABLE ENTRY
                               4990 *
    0EDD 1F 00 0E9C 4A         4991 LVI440 SLC   LVI395+@D1,LVIH02(1,@BR)  DECK PT
    0EE2 C0 02 0E96            4992        BNM   LVI390                    PROCESS UNTIL LAST ENTRY
                               4993 *
                               4994 * TEST PRINT SWITCH
                               4995 *
    0EE6 7D 00 48              4996 LVI450 CLI   LVIH01(,@BR),*-*          IS SWITCH ON ?
    0EE9 F2 01 14              4997        JNE   LVI470                    INIT REGION TWO
                               4998 *
                               4999 * WRITE INITIAL1ZED BUFFERS TO VIRTUAL MEMORY
                               5000 *
    0EEC C0 87 129B            5001 LVI460 B     LVI850                    IOCR ROUTINE TO PUT TO VM
    0EF0 7C 01 FB              5002        MVI   LVIPIN(,@BR),LVIBYC       SET PAGE COUNT TO ONE
    0EF3 1C 01 121B 50         5003        MVC   LVI805+@OP1,LVIBRS(@CADDR,@BR)  RESET BFR 1 CADDR IN PT
    0EF8 3C 00 1229            5004        MVI   LVI810+@D1,LVINUL         ZERO BUFFER DISP
    0EFC 3C 01 1416            5005        MVI   LVISWC,LVIBYC             RESET SWITCH                 1-5
```

```
                               5007 ***********************************************************************
                               5008 *                                                                     *
                               5009 *      ARRAY INITIALIZATION - REGION 2                                *
                               5010 *                                                                     *
                               5011 ***********************************************************************
                               5012 *
                               5013 ***********************************************************************
                               5014 * ARITHMETIC ARRAY INITIALIZATION - REGION 2
                               5015 ***********************************************************************
                               5016 *
    0F00 7D 00 48              5017 LVI470 CLI   LVIH01(,@BR),*-*          IS SWITCH ON ?
    0F03 4C 00 FA 1A04         5018        MVC   LVIPCT(1,@BR),LVIVA2      SET 1ST PG PARAM TO REGION 2
    0F08 F2 01 58              5019        JNE   LVI550                    CHECK CHAR REGION 2 SW
                               5020 *
                               5021 * SET INITIALIZATION PUT ROUTINE PARAMETERS
                               5022 *
    0F0B 1C 01 122B E6         5023 LVI480 MVC   LVI810+@DOP2(@CADDR),LVIAIV(,@BR)  SET ADDR ARITH VALUE
    0F10 3C 00 1228            5024 LVI482 MVI   LVI810+@Q,*-*             SET BYTES IN VALUE TO MOVE
    0F11                       5025        ORG   LVI482+@Q                 * INITIALLY SET TO SHORT
    0F11 04               0F11 5026        DC    AL1(B@LISP-1)             * PRECISION LENGTH
    0F14                       5027        ORG
    0F14 7C 00 D6              5028 LVI484 MVI   LVIPLN(,@BR),*-*          SET LENGTH OF VALUE,
    0F15                       5029        ORG   LVI484+@Q                 * INITIALLY SET TO
    0F15 05               0F15 5030        DC    AL1(B@LISP)               * SHORT VALUE LNG
    0F17                       5031        ORG
    0F17 7C 00 DA              5032 LVI486 MVI   LVICNT(,@BR),*-*          SET THE DISP TO THE LAST
    0F18                       5033        ORG   LVI486+@Q                 * BYTE OF THE INIT VALUE
    0F18 04               0F18 5034        DC    AL1(B@LISP-1)             * TO INIT THE MOVE PT
    0F1A                       5035        ORG
                               5036 *
                               5037 * SELECT ARRAY SYMBOL TABLE ENTRY AND TEST FOR DEFINITION
                               5038 *
    0F1A C2 02 1CC4            5039 LVI490 LA    LVINAT,@XR
    0F1E B5 02 00              5040 LVI495 L     *-*(,@XR),@XR             LOAD DOPE VECTOR VADDR FROM
    0F20                       5041        ORG   LVI495+@D1                * ARITH SYM TBL, INITIALLY SET
    0F20 39               0F20 5042        DC    AL1(B@LL12-1)             * FOR LAST TBL ENTRY
    0F21                       5043        ORG
    0F21 7B 80 AF              5044        SBF   LVISPS(,@BR),LVITMK       SET TRACE BIT OFF
    0F24 76 02 46              5045        A     LVIH00(,@BR),@XR          IS ENTRY NULL
    0F27 F2 81 2D              5046        JE    LVI540                    YES, DECR TO NEXT ETRY
                               5047 *
                               5048 * TEST IF ARRAY IS IN REGION 2
                               5049 *
    0F2A 76 02 52              5050 LVI500 A     LVIAAC(,@BR),@XR          CONVERT DIV VADDR TO CADDR
    0F2D 2D 01 1A03 07         5051        CLC   LVIRG1(@CADDR),B@ABAS(,@XR)  IN REGION 2
    0F32 F2 84 22              5052        JH    LVI540                    NO, DECR TO NEXT ENTRY
                               5053 *
                               5054 * SET OUT PARAMETER AND TEST TRACE SWITCH
                               5055 *
    0F35 6C 01 E4 05           5056 LVI510 MVC   LVIELC(LVIBY2,@BR),B@AMAX(,@XR)  SET NO. ELEMENTS
    0F39 0C 00 0F45 0F20       5057        MVC   LVI515+@D1(1),LVI495+@D1  SET ENTRY DISP TO INCR PT BY
    0F3F C2 02 143A            5058        LA    LVIPTL,@XR                ADDR TRACE TBL
    0F43 E2 02 00              5059 LVI515 LA    *-*(,@XR),@XR             ADD DIP TO TRACE TBL ENTRY
    0F46 B8 08 00              5060        TBN   LVI0TD(,@XR),LVIAAA       IS TRACE ALL SW ON
    0F49 F2 90 03              5061        JF    LVI530                    NO, GO INITIALIZE ARRAY
                               5062 *
                               5063 * SET STATUS BYTE IN INITIALIZATION VALUE
                               5064 *
    0F4C 7A 80 AF              5065 LVI525 SBN   LVISPS(,@BR),LVITMK       SET TRACE BIT ON
                               5066 *
                               5067 * INITIALIZE ARRAY AND SET PRINT SWITCH ON
                               5068 *
    0F4F C0 87 120F            5069 LVI530 B     LVI800                    INIT ARRAY
    0F53 3C 01 0FC2            5070        MVI   LVI630+@Q,LVISWO          SET PRINT SW ON
                               5071 *
                               5072 * DECREMENT ARRAY TABLE POINTER TO ACCESS NEXT TABLE ENTRY
                               5073 *
    0F57 1F 00 0F20 4A         5074 LVI540 SLC   LVI495+@D1,LVIH02(1,@BR)  DECR PT
    0F5C C0 02 0F1A            5075        BNM   LVI490                    PROCESS UNTIL LAST ENTRY
    0F60 7B 80 AF              5076        SBF   LVISPS(,@BR),LVITMK       SET TRACE BIT OFF IN INIT VALUE

                               5078 ***********************************************************************
                               5079 * CHARACTER ARRAY INITIALIZATION
                               5080 ***********************************************************************
                               5081 *
                               5082 * TEST REGION 2 SWITCH
                               5083 *
    0F63 7D 00 48              5084 LVI550 CLI   LVIH01(,@BR),*-*          IS SW ON ?
    0F66 F2 01 58              5085        JNE   LVI630                    INIT ARRAY ELEMENTS
                               5086 *
                               5087 * SET INITIALIZATION PUT ROUTINE PARAMETERS
                               5088 *
    0F69 1C 01 122B 5A         5089 LVI560 MVC   LVI810+@DOP2(@CADDR),LVICHV(,@BR)  SET ADDR CHAR VALUE
    0F6E 3C 12 1228            5090        MVI   LVI810+@Q,B@LCRV-1        BYTES IN CHAR VALUE
    0F72 7C 13 D6              5091        MVI   LVIPLN(,@BR),B@LCRV       LENGTH OF CHAR VALUE
    0F75 7C 12 DA              5092        MVI   LVICNT(,@BR),B@LCRV-1     INIT DISP IN MOVE PT
                               5093 *
                               5094 * SELECT ARRAY SYMBOL TABLE ENTRY AND TEST FOR DEFINITION
                               5095 *
    0F78 C2 02 1CFE            5096 LVI570 LA    LVICAT,@XR                ADDR CHAR SYM TBL
    0F7C B5 02 00              5097 LVI575 L     *-*(,@XR),@XR             LOAD DOPE VECTOR VADDR FROM
    0F7E                       5098        ORG   LVI575+@D1                * CHAR SYM TBL, INITIALLY SET
    0F7E 39               0F7E 5099        DC    AL1(B@LL13-1)             * FOR LAST TBL ENTRY
    0F7F                       5100        ORG
    0F7F 76 02 46              5101        A     LVIH00(,@BR),@XR          IS ENTRY NULL
    0F82 F2 81 33              5102        JE    LVI620                    YES, DECR TO NEXT ENTRY
                               5103 *
                               5104 * TEST IF ARRAY IS IN REGION 2
                               5105 *
    0F85 76 02 52              5106 LVI580 A     LVIAAC(,@BR),@XR          CONVERT DIV VADDR TO CADDR
    0F88 2D 01 1A03 03         5107        CLC   LVIRG1(@CADDR),B@CBAS(,@XR)  IN REGION 2
    0F8D F2 84 28              5108        JH    LVI620                    NO, DECR TO NEXT ENTRY
                               5109 *
                               5110 * SET PUT PARAMETER AND TEST TRACE SWITCH
                               5111 *
    0F90 6C 01 E4 01           5112 LVI590 MVC   LVIELC(LVIBY2,@BR),B@CDMN(,@XR)  SET NO. ELEMENTS
    0F94 0C 00 0FA0 0F7E       5113        MVC   LVI595+@D1(1),LVI575+@D1  SET ENTRY DISP TO INCR PT BY
    0F9A C2 02 143A            5114        LA    LVIPTL,@XR                ADDR TRACE TBL
    0F9E E2 02 00              5115 LVI595 LA    *-*(,@XR),@XR             ADD DISP TO TRACE TBL ENTRY
    0FA1 B8 02 00              5116        TBN   LVI0TD(,@XR),LVICAA       IS TRACE ALL SW ON
    0FA4 F2 10 06              5117        JT    LVI605                    YES, SET TRACE BIT ON
                               5118 *
                               5119 * SET STATUS BYTE IN INITIALIZATION VALUE
                               5120 *
    0FA7 7C 40 B8              5121 LVI600 MVI   LVICSB(,@BR),LVICTF       SET TRACE BIT OFF IN INIT VALUE
    0FAA F2 87 03              5122        J     LVI610                    MOVE VALUES TO VM
    0FAD 7C C0 B8              5123 LVI605 MVI   LVICSB(,@BR),LVICTN       SET TRACE BIT ON IN INIT VALUE
                               5124 *
                               5125 * INITIALIZE ARRAY AND SET PRINT SWITCH ON
                               5126 *
    0FB0 C0 87 120F            5127 LVI610 B     LVI800                    INIT ARRAY
    0FB4 3C 01 0FC2            5128        MVI   LVI630+@Q,LVISWO          SET PRINT SW ON
                               5129 *
                               5130 * DECREMENT ARRAY TABLE POINTER TO ACCESS NEXT TABLE ENTRY
                               5131 *
    0FB8 1F 00 0F7E 4A         5132 LVI620 SLC   LVI575+@D1,LVIH02(1,@BR)  DECR PT
    0FBD C0 02 0F78            5133        BNM   LVI570                    PROCESS UNTIL LAST ENTRY
                               5134 *
                               5135 * TEST PRINT SWITCH
                               5136 *
    0FC1 7D 00 48              5137 LVI630 CLI   LVIH01(,@BR),*-*          IS SW ON
    0FC4 C0 01 0CDF            5138        BNE   LVI060                    INIT INTERNAL CONS
                               5139 *
                               5140 * WRITE INITIalIZED BUFFERS TO VIRTUAL MEMORY
                               5141 *
    0FC8 C0 87 129B            5142 LVI640 B     LVI850                    IOCR RTN FOR PUT TO VM
    0FCC C0 87 0CDF            5143        B     LVI060                    INIT INTERNAL CONSTANTS
```

```
                               5145 ***********************************************************************
                               5146 *                                                                     *
                               5147 *      ARRAY ELEMENT INITIALIZATION                                   *
                               5148 *                                                                     *
                               5149 ***********************************************************************
                               5150 *
                               5151 ***********************************************************************
                               5152 * ARITHMETIC ARRAY ELEMENT INITIALIZATION
                               5153 ***********************************************************************
                               5154 *
                               5155 * TEST FOR NO SCALAR REFERENCES IN A TRACE ALL CONDITION
                               5156 *
    0FD0 7D 00 D5              5157 LVI670 CLI   LVITSW(,@BR),LVINUL       IS SW OFF
    0FD3 C0 81 1336            5158        BE    LVI991                    YES, SET ERROR CODE
                               5159 *
                               5160 * TEST ELEMENT INITIALIZATION SWITCH
                               5161 *
    0FD7 7D 00 48              5162 LVI675 CLI   LVIH01(,@BR),*-*          IS SW ON
    0FDA C0 01 1110            5163        BNE   LVI760                    INIT CHAR ELEMENTS
    0FDE 3C 00 12DC            5164        MVI   LVI933+@D1,LVI0TD         DISP TO 1ST VALUE IN BFR
    0FE2 7A 80 AF              5165        SBN   LVISPS(,@BR),B@TRAC       SET VALUE TRACE BIT ON
                               5166 *
                               5167 * SELECT ARRAY SYMBOL TABLE ENTRY AND TEST FOR DEFINITION
                               5168 *
    0FE5 C2 02 1CC4            5169 LVI680 LA    LVINAT,@XR                ADDR ARITH SYM TBL
    0FE9 E2 02 00              5170 LVI685 LA    *-*(,@XR),@XR             LOAD DOPE VECTOR VADDR FROM
    0FEB                       5171        ORG   LVI685+@D1                * ARITH SYM TBL, INITIALLY SET
    0FEB 38               0FEB 5172        DC    AL1(B@LL12-2)             * FOR LAST TBL ENTRY
    0FEC                       5173        ORG
    0FEC 9D 01 01 46           5174        CLC   LVITD1(,@XR),LVIH00(@VADDR,@BR)  IS ENTRY NULL ?
    0FF0 C0 81 1107            5175        BE    LVI755                    YES, DECR TO NEXT ENTRY
                               5176 *
                               5177 * SAVE DOPE VECTOR VADDR AND CHECK TRACE TABLE SWITCHES
                               5178 *
    0FF4 34 02 1091            5179 LVI690 ST    LVI734+@OP1,@XR           SAVE DOPE VECTOR VADDR
    0FF8 0C 00 1004 0FEB       5180        MVC   LVI695+@D1(1),LVI685+@D1  SET DISP TO TRACE TBL ENTRY
    0FFE C2 02 143A            5181        LA    LVIPTL,@XR                ADDR TRACE TBL
    1002 E2 02 00              5182 LVI695 LA    *-*(,@XR),@XR             ADD DIP TO TRLINTATY
    1005 B8 04 01              5183        TBN   LVI1TD(,@XR),LVIAAP       IS TRACE ELEMENT SW ON
    1008 F2 90 FC              5184        JF    LVI755                    NO, DECR NAT DISP
    100B B8 08 01              5185        TBN   LVI1TD(,@XR),LVIAAA       IS TRACE ALL ON ?
    100E F2 10 F6              5186        JT    LVI755                    YES, DECR NAT DISP
                               5187 *
                               5188 * DETERMINE ALPHABETIC CHARACTER AND SAVE IT
                               5189 *
    1011 3C 00 102E            5190        MVI   LVI705+@D1,LVINUL         CLEAR DISP
    1015 1F 00 1004 4A         5191 LVI700 SLC   LVI695+@D1,LVIH02(1,@BR)  DIVIED THE TRACE TABLE DISP BY
    101A 1E 00 102E 48         5192        ALC   LVI705+@D1,LVIH01(1,@BR)  * 2 TO OBTAIN THE DISP TO THE
    101F 1D 00 1004 46         5193        CLC   LVI695+@D1,LVIH00(1,@BR)  * ALPHA CHAR IN THE ALPHA
    1024 C0 01 1015            5194        BNE   LVI700                    * TABLE
    1028 C2 02 141D            5195        LA    LVIATL,@XR                ADDR ALPHA TBL
    102C E2 02 00              5196 LVI705 LA    *-*(,@XR),@XR             ADD DISP TO CHAR
    102F 2C 00 103F 00         5197        MVC   LVI715+@Q,LVI0TD(1,@XR)   SAVE LETTER
                               5198 *
                               5199 * INCREMENT THROUGH TRACE REFERENCE LIST UNTIL LETTER IS FOUND
                               5200 *
    1034 C2 02 0000            5201 LVI710 LA    *-*,@XR                   ADDR 1ST BYTE TRACE REF LIST
    1038 BD 1E 00              5202 LVI713 CLI   LVITD0(,@XR),B@EOST       AT EOS ?
    103B F2 81 C9              5203        JE    LVI755                    YES, CHECK NEXT TBL ENTRY
    103E BD 00 00              5204 LVI715 CLI   LVITD0(,@XR),*-*          BYTE = THE LETTER ?
    1041 F2 01 0F              5205        JNE   LVI725                    NO, INCR TO NEXT BYTE
                               5206 *
                               5207 * DETERMINE IF LIST REFERENCE IS THE PROPER ARRAY REFERENCE
                               5208 *
    1044 76 02 48              5209 LVI720 A     LVIH01(,@BR),@XR          INCR TO NEXT BYTE
    1047 BD 4D 00              5210        CLI   LVITD0(,@XR),B@LPAR       FOLLOWING BYTE A LEFT PAREN
    104A F2 01 06              5211        JNE   LVI725                    NO, INCR TO NEXT BYTE
    104D BD F0 01              5212        CLI   LVITD1(,@XR),B@DEC0       IS FOLLOWING BYTE A NUMBER
    1050 F2 02 07              5213        JNL   LVI730                    YES, PROCESS SUBSCRIPTS
                               5214 *
                               5215 * INCR TRACE REF LIST POINTER TO NEXT BYTE
                               5216 *
    1053 76 02 48              5217 LVI725 A     LVIH01(,@BR),@XR          INCR DT 1 BYTE
    1056 C0 87 1038            5218        B     LVI713                    LOOP UNTIL LETTER FOUND
                               5219 *
                               5220 * CONVERT SUBSCRIPT(S) TO BINARY
                               5221 *
    105A 76 02 48              5222 LVI730 A     LVIH01(,@BR),@XR          INCR PT TO 1ST DIGIT
    105D C0 87 16DE            5223        B     C4BIN2                    CONVERT NO. TO BINARY
    1061 74 02 39              5224        ST    LVI996+@OP1(,@BR),@XR     SAVE PT IN ERROR RTN
    1064 4C 01 E8 1748         5225        MVC   LVISS1(LVIBY2,@BR),C4BVAL  SAVE BINARY SUBSCRIPT
    1069 BD 5D 00              5226        CLI   LVITD0(,@XR),B@RPAR       IS BYTE A RT PAREN ?
    106C F2 81 8C              5227        JE    LVI750                    YES, CNECK VALIDITY OF SUBSC
    106F BD 6B 00              5228        CLI   LVITD0(,@XR),B@CMMA       IS BYTE A COMMA ?
    1072 D0 01 2C              5229        BNE   LVI994(,@BR)              NO, SET ERROR CODE
    1075 76 02 48              5230        A     LVIH01(,@BR),@XR          INCR PT TO 1ST DIGIT
    1078 C0 87 16DE            5231        B     C4BIN2                    CONVERT NO. TO BINARY
    107C 74 02 39              5232        ST    LVI996+@OP1(,@BR),@XR     SAVE PT IN ERROR RTN
    107F 4C 01 EA 1748         5233        MVC   LVISS2(LVIBY2,@BR),C4BVAL  SAVE BINARY SUBSC
    1084 BD 5D 00              5234        CLI   LVITD0(,@XR),B@RPAR       A RIGHT PAREN ?
    1087 D0 01 2C              5235        BNE   LVI994(,@BR)              NO, SET ERROR CODE
    108A 34 02 10F6            5236 LVI733 ST    LVI749+@OP1,@XR           SAVE LINE PT
                               5237 *
                               5238 * TEST VALIDITY OF SUBSCRIPTS
                               5239 *
    108E C2 02 0000            5240 LVI734 LA    *-*,@XR                   VADDR DOPE VECTOR
    1092 B5 02 01              5241        L     LVITD1(,@XR),@XR          SELECT VADDR FROM ENTRY
    1095 76 02 52              5242        A     LVIAAC(,@BR),@XR          CONVERT DIV VADDR TO CADDR
    1098 6D 01 46 01           5243 LVI735 CLC   LVIH00(B@LBIN,@BR),B@ACD1(,@XR)  IS D/V A VECTOR ?
    109C F2 01 0B              5244        JNE   LVI736                    NO, CHECK IF INPUT IS MATRIX
    109F 5D 01 E8 46           5245        CLC   LVISS1(B@LBIN,@BR),LVIH00(,@BR)  IS INPUT SUBSC A VECTOR
    10A3 F2 81 0C              5246        JE    LVI737                    YES, DICK FOR VALID SUBSC
    10A6 C0 87 1344            5247        B     LVI993                    SET ERROR CODE
    10AA 5D 01 E8 46           5248 LVI736 CLC   LVISS1(B@LBIN,@BR),LVIH00(,@BR)  IS INPUT SUBSC A MATRIX ?
                               5249 *
    10AE C0 81 1344            5250        BE    LVI993                    NO, SET ERROR CODE
    10B2 6D 01 E8 01           5251 LVI737 CLC   LVISS1(LVIBY2,@BR),B@ACD1(,@XR)  IS SUBSC 1 VALID ?
    10B6 D0 84 2C              5252        BH    LVI994(,@BR)              NO, SET ERROR CODE
    10B9 5D 01 EA 03           5253        CLC   LVISS2(LVIBY2,@BR),B@ACD2(,@BR)  IS SUBSC 2 VALID ?
    10BD D0 84 2C              5254        BH    LVI994(,@BR)              NO, SET ERROR CODE
                               5255 *
                               5256 * DETERMINE ELEMENT DISPLACEMENT FROM ARRAY BASE ADDR
                               5257 *
    10C0 7C 00 DC              5258 LVI738 MVI   LVIECT(,@BR),*-*          SET ELEMENT LENGTS,
    10C1                       5259        ORG   LVI738+@Q                 * INITIALLY CONTAINS A VALUE
    10C1 04               10C1 5260        DC    AL1(B@LISP-1)             * FOR SNORT 01/EC
    10C3                       5261        ORG
    10C3 F2 87 04              5262        J     LVI742                    DECR SUBSC 1 BY 1
    10C6 6E 01 EA 03           5263 LVI740 ALC   LVISS2(LVIBY2,@BR),B@ACD2(,@XR)  ADD 2ND DIM TO SUBSC 2
    10CA 5F 01 E8 48           5264 LVI742 SLC   LVISS1(LVIBY2,@BR),LVIH01(,@BR) SUBSC 1 AND CONTINUE
    10CE C0 84 10C6            5265        BH    LVI740                    * UNTIL SUBSC EQ 0
    10D2 5C 01 E8 DC           5266        MVC   LVISS1(LVIBY2,@BR),LVIECT(,@BR)  INIT FOR LAST EL BYTE
    10D6 5E 01 E8 EA           5267 LVI746 ALC   LVISS1(LVIBY2,@BR),LVISS2(,@BR)  MULTIPLY BY THE LENGTH
    10DA 5F 00 DC 48           5268        SLC   LVIECT(1,@BR),LVIH01(,@BR)       * OF THE ARRAY ELEMENT
    10DE C0 02 10D6            5269        BNL   LVI746                           *
                               5270 *
                               5271 * INCREMENT ARRAY BASE ADDRESS, SET PUT ROUTINE PARAMETERS
                               5272 *
    10E2 6E 01 E8 07           5273 LVI748 ALC   LVISS1(@VADDR,@BR),B@ABAS(,@XR)  ADD BASE ADDR TO DISP
    10E6 1C 00 12DF E8         5274        MVC   LVI935+@D1,LVISS1(1,@BR)  VALUE DISP IN PG
    10EB 5C 00 D7 E7           5275        MVC   LVIHLD(1,@BR),LVISS1-1(,@BR)  PG THE ELEMENT IS IN
    10EF C0 87 12AF            5276        B     LVI900                    MOVE VALUE TO VM
    10F3 C2 02 0000            5277 LVI749 LA    *-*,@XR                   RESTORE LINE PT
    10F7 C0 87 1053            5278        B     LVI725                    INCR REF LIST PT
                               5279 *
                               5280 * SET BINARY SUBSCRIPT SAVE AREAS TO PROCESS AS TWO SUBCRIPTED ARRAY
                               5281 *
    10FB 5C 01 EA E8           5282 LVI750 MVC   LVISS2(LVIBY2,@BR),LVISS1(,@BR)  SHIFT SUBSC
    10FF 5F 01 E8 E8           5283        SLC   LVISS1(LVIBY2,@BR),LVISS1(,@BR)  SET SUBSC 1 TO ZERO
    1103 C0 87 108A            5284        B     LVI733                    TEST-FOR SUBSC VALIDITY
                               5285 *
                               5286 * DECREMENT NAT TABLE DISPLACEMENT TO NEXT ENTRY
                               5287 *
    1107 1F 00 0FEB 4A         5288 LVI755 SLC   LVI685+@D1,LVIH02(1,@BR)  DECR PT
    110C C0 84 0FE5            5289        BH    LVI680                    LOOP UNTIL DISP IS 0
```

```
                               5291 ***********************************************************************
                               5292 *                                                                     *
                               5293 *      CHARACTER ARRAY ELEMENT INITIALIZATION                         *
                               5294 *                                                                     *
                               5295 ***********************************************************************
                               5296 *
                               5297 * TEST ELEMENT INITIALIZATION SWITCH
                               5298 *
    1110 7D 00 48              5299 LVI760 CLI   LVIH01(,@BR),*-*          IS SW ON ?
    1113 F2 01 E5              5300        JNE   LVI798                    NO, MOVE BFR TO VM
    1116 7A 80 B8              5301        SBN   LVICSB(,@BR),B@TRAC       SET TRACE BIT IN VALUE
    1119 3C 12 12DC            5302        MVI   LVI933+@D1,B@LCRV-1       SET DISP TO VAL
                               5303 *
                               5304 * SET PUT ROUTINE PARAMETERS
                               5305 *
    111D 3C 12 12DE            5306 LVI762 MVI   LVI935+@Q,B@LCRV-1        BYTES IN CHAR VALUE
    1121 1C 01 12E1 5A         5307        MVC   LVI935+@DOP2,LVICHV(@CADDR,@BR)  ADDR INIT VALUE
                               5308 *
                               5309 * SELECT ARRAY SYMBOL TABLE ENTRY AND TEST FOR DEFINITION
                               5310 *
    1126 C2 02 1CFE            5311 LVI775 LA    LVICAT,@XR                ADDR CHAR SYM TBL
    112A E2 02 00              5312 LVI777 LA    *-*(,@XR),@XR             LOAD DOPE VECTOR VADDR FROM
    112B                       5313        ORG   LVI777+@Q                 * CHAR SYM TBL, INITIALLY SET
    112B 38               112B 5314        DC    AL1(B@LL13-2)             * FOR LAST TBL ENTRY
    112D                       5315        ORG
    112D 9D 01 01 46           5316        CLC   LVITD1(,@XR),LVIH00(@VADDR,@BR) IS ENTRY NULL ?
    1131 F2 81 BE              5317        JE    LVI797                    YES, DECR TBL DISP
                               5318 *
                               5319 * SAVE DOPE VECTOR VADDR AND CHECK TRACE TABLE SWITCHES
                               5320 *
    1134 34 02 11BC            5321 LVI778 ST    LVI792+@OP1,@XR           SAVE DOPE VECTOR VADDR
    1138 0C 00 1144 112C       5322        MVC   LVI779+@D1(1),LVI777+@D1  SET DISP TO TRACE TBL ENTRY
    113E C2 02 143A            5323        LA    LVIPTL,@XR                ADDR TRACE TBL
    1142 E2 02 00              5324 LVI779 LA    *-*(,@XR),@XR             ADD DISP TO TBL ENTRY
    1145 B8 01 01              5325        TBN   LVI1TD(,@XR),LVICAP       IS TRACE ELEMENT SW ON ?
    1148 F2 90 A7              5326        JF    LVI797                    NO, DECR CAT DISP
    114B B8 02 01              5327        TBN   LVI1TD(,@XR),LVICAA       IS TRACE ALL ON ?
    114E F2 10 A1              5328        JT    LVI797                    YES, DECR CAT DISP
                               5329 *
                               5330 * DETERMINE ALPHABETIC CHARACTER AND SAVE IT
                               5331 *
    1151 3C 00 116E            5332        MVI   LVI782+@D1,LVINUL         ZERO INST DISP
    1155 1F 00 1144 4A         5333 LVI780 SLC   LVI779+@D1,LVIH02(1,@BR)  DIVIDE THE TRACE TABLE DISP TO
    115A 1E 00 116E 48         5334        ALC   LVI782+@D1,LVIH01(1,@BR)  * OBTAIN THE DISP TO THE PROPER
    115F 1D 00 1144 46         5335        CLC   LVI779+@D1,LVIH00(1,@BR)  * ALPHA CHAR IN THE ALPHA
    1164 C0 01 1155            5336        BNE   LVI780                    * TABLE
    1168 C2 02 141D            5337        LA    LVIATL,@XR                ADDR ALPHA TBL
    116C E2 02 00              5338 LVI782 LA    *-*(,@XR),@XR             ADD DISP TO
    116F 2C 00 117F 00         5339        MVC   LVI786+@Q,LVI0TD(1,@XR)   SAVE LETTER
                               5340 *
                               5341 * INCREMENT THROUGH THE TRACE REFERENCE LIST UNTIL LETTER IS FOUND
                               5342 *
    1174 C2 02 0000            5343 LVI784 LA    *-*,@XR                   ADDR 1ST BYTE TRACE LIST
                               5344 *
    1178 BD 1E 00              5345 LVI785 CLI   LVITD0(,@XR),B@EOST       AT EOS ?
    117B F2 81 74              5346        JE    LVI797                    YES, CHECK NEXT TBL ENTRY
    117E BD 00 00              5347 LVI786 CLI   LVITD0(,@XR),*-*          BYTE EQ THE LETTER ?
    1181 F2 01 15              5348        JNE   LVI790                    NO, INCR TO NEXT BYTE
                               5349 *
                               5350 * DETERMINE IF LIST REFERENCE IS THE PROPER ARRAY REFERENCE
                               5351 *
    1184 76 02 48              5352 LVI788 A     LVIH01(,@BR),@XR          INCR OT TO NEXT BYTE
    1187 BD 5B 00              5353        CLI   LVITD0(,@XR),B@CVAR       IS REF CHAR
    118A F2 01 0C              5354        JNE   LVI790                    NO, INCR PT TO NEXT BYTE
    118D BD 4D 01              5355        CLI   LVITD1(,@XR),B@LPAR       IS CHAR REF AN ARRAY
    1190 F2 01 06              5356        JNE   LVI790                    NO, INCR PT TO NEXT BYTE
    1193 BD F0 02              5357        CLI   LVITD2(,@XR),B@DEC0       IS REF A CHAR ARRAY WITH SUBSC
    1196 F2 02 07              5358        JNL   LVI791                    YES, PROCESS SUBSCRIPTS
                               5359 *
                               5360 * INCR TRACE REF LIST POINTER TO NEXT BYTE
                               5361 *
    1199 76 02 48              5362 LVI790 A     LVIH01(,@BR),@XR          INCR PT
    119C C0 87 1178            5363        B     LVI785                    LOOP UNTIL LETTER IS FOUND
                               5364 *
                               5365 * CONVERT SUBSCRIPT TO BINARY
                               5366 *
    11A0 76 02 4A              5367 LVI791 A     LVIH02(,@BR),@XR          INCR OT TO 15T DIGIT
    11A3 C0 87 16DE            5368        B     C4BIN2                    CONVERT NO. TO BINARY
    11A7 4C 01 EA 1748         5369        MVC   LVISS2(LVIBY2,@BR),C4BVAL  SAVE BINARY SUBC
    11AC BD 5D 00              5370        CLI   LVITD0(,@XR),B@RPAR       AT RIGHT PAREN ?
    11AF 74 02 39              5371        ST    LVI996+@OP1(,@BR),@XR     SAVE PT IN ERROR RTN
    11B2 D0 01 2C              5372        BNE   LVI994(,@BR)              NO, SET ERROR CODE
    11B5 34 02 11ED            5373        ST    LVI796+@OP1,@XR           SAVE LINE PT
                               5374 *
                               5375 * TEST FOR VAL1DITY OF SUBSCRIPT
                               5376 *
    11B9 C2 02 0000            5377 LVI792 LA    *-*,@XR                   VADDR DOPE VECTOR
    11BD B5 02 01              5378        L     LVITD1(,@XR),@XR          SELECT VADDR FROM ENTRY
    11C0 76 02 52              5379        A     LVIAAC(,@BR),@XR          CONVERT DIV VADDR TO CADDR
    11C3 6D 01 EA 01           5380        CLC   LVISS2(LVIBY2,@BR),B@CDMN(,@XR)  IS SUBSC VALID
    11C7 D0 84 2C              5381        BH    LVI994(,@BR)              NO, SET ERROR CODE
                               5382 *
                               5383 * INCREMENT ARRAY BASE ADDRESS, SET PUT PARAMETERS
                               5384 *
    11CA 7C 13 DC              5385 LVI794 MVI   LVIECT(,@BR),B@LCRV       SET ELEMENT LNG CT
    11CD 6C 01 E8 03           5386        MVC   LVISS1(@CADDR,@BR),B@CBAS(,@XR)  ADD BASE ADDR TO ACCUM
    11D1 5E 01 E8 EA           5387 LVI795 ALC   LVISS1(LVIBY2,@BR),LVISS2(,@BR)  INCR BASE ADDR BY THE
    11D5 5F 00 DC 48           5388        SLC   LVIECT(1,@BR),LVIH01(,@BR) * DISPLACEMENT OF THE
    11D9 C0 84 11D1            5389        BH    LVI795                     * EL, EL LNG TIMES
    11DD 1C 00 12DF E8         5390        MVC   LVI935+@D1,LVISS1(1,@BR)  ELEMENT DISP TO DG PARAM
    11E2 5C 00 D7 E7           5391        MVC   LVIHLD(1,@BR),LVISS1-1(,@BR)  EL PG TO PG PARAM
    11E6 C0 87 12AF            5392        B     LVI900                    MOVE VALUE TO VM
    11EA C2 02 0000            5393 LVI796 LA    *-*,@XR                   RESTORE LINE PT
    11EE C0 87 1178            5394        B     LVI785                    RECYCLE LOOP
                               5395 *
                               5396 * DECREMENT CAT TABLE D1SPLACEQTNT TO NEXT ENTRY
                               5397 *
    11F2 1F 00 112C 4A         5398 LVI797 SLC   LVI777+@D1,LVIH02(1,@BR)  DECR PT
    11F7 C0 02 1126            5399        BNM   LVI775                    LOOP UNTIL DISP IS 0
                               5400 *
                               5401 * WRITE PUT BUFFER AND EXIT LVINIT TO LRADDR
                               5402 *
    11FB 5C 00 ED F3           5403 LVI798 MVC   LVIOUT(1,@BR),LVIINN(,@BR)  SET PUT PG NO.
    11FF C0 87 17E7            5404        B     DL4ICS                    PUT PG TO VM
    1203 140A             1204 5405        DC    AL(@CADDR)(LVIPUT)        ADDR DISK PARAM LIST
    1205 C0 87 0025            5406        B     $DISKN                    WAIT FOR COMPLETION
    1209 057F             120A 5407        DC    AL(@CADDR)($WAITF)        WAIT PAREM
    120B C0 87 1474            5408        B     LRADDR                    RESOLVE BR ADDR TABLE
```

```
                               5410 ************************************************************************
                               5411 *                                                                      *
                               5412 *      ARRAY INITIAL1ZATION PUT ROUTINE                                *
                               5413 *                                                                      *
                               5414 ************************************************************************
                               5415 *
                               5416 * SAVE RETURN ADDRESS
                               5417 *
    120F 34 08 129A            5418 LVI800 ST    LVI840+@OP1,@ARR          SAVE RETURN ADDRESS
    1213 1E 01 121B DA         5419        ALC   LVI805+@OP1,LVICNT(LVIBY2,@BR) INCR PT TO LAST BYTE ADDR
    1218 C2 02 0000            5420 LVI805 LA    *-*,@XR                   ADDR OF 1ST AVAIL BYTE IN BFR
    121A                       5421        ORG   *-2                       * INITIALLY SET TO THE FIRST
    121A 0700             121B 5422        DC    AL(@CADDR)(LVIIB1)        * BUFFER ADDR
    121C 7D 04 F7              5423        CLI   LVISWC(,@BR),LVIBOF       SWITCH = 4                   1-5
    121F C0 01 1242            5424        BNE   LVI812                    @BR IF NOT TO FILL BUFFER    1-5
    1223 C0 87 125A            5425        B     LVI815                    GO TEST FOR OVERFLOW         1-5
                               5426 *
                               5427 * PREPARE ARRAY INITIALIZATION BUFFER WITH INITIALIZING VALUES
                               5428 *
    1227 8C 00 00 0000         5429 LVI810 MVC   *-*(@VQ,@XR),*-*          MOVE 1 ELEMENT TO ARRAY INIT
    1229                       5430        ORG   LVI810+@D1                * BUFFER, INITIALLY SET TO
    1229 00               1229 5431        DC    XL1'00'                   * THE FIRST VALUE
    122C                       5432        ORG
    122C C0 87 048D            5433        B     $UNMSK                    UNMASK THE LOADER AT THIS POINT
    1230 7D 02 F7              5434        CLI   LVISWC(,@BR),LVIBY2       IF SWITCH NO = 2             1-5
    1233 F2 01 16              5435        JNE   LVI814                    * GO ADD LENGTH TO INST      1-5
    1236 7C 01 F7              5436        MVI   LVISWC(,@BR),LVIBYC       SET SWITHC - 1               1-5
    1239 1E 00 1229 D6         5437        ALC   LVI810+@D1,LVIPLN(1,@BR)  INCR PT BY VALUE LNG
    123E C0 02 125A            5438        BNL   LVI815                    GO TEST OVERFLOW (>=0)       1-5
                          1242 5439 LVI812 EQU   *                         * VERY VERY TEMP !!! HJS 2020
                               5440 *       SLC   LVIELC(LVIBY2,@BR),LVIH01(,@BR)  ELEMENT CNT LESS 1   1-5
    1242 F2 04 03              5441        JNH   LVI813                    IF <=0 DON'T RESET SW        1-5
    1245 7C 02 F7              5442        MVI   LVISWC(,@BR),LVIBY2       SET SWITCH = 2               1-5
    1248 C0 87 1227            5443 LVI813 B     LVI810                    GO MOVE ELEMENT              1-5
    124C 1E 00 1229 D6         5444 LVI814 ALC   LVI810+@D1,LVIPLN(1,@BR)                               1-5
    1251 F2 04 37              5445        JNH   LVI833                                                 1-5
    1254 7C 04 F7              5446        MVI   LVISWC(,@BR),LVIBOF                                    1-5
    1257 F2 87 34              5447        J     LVI835                                                 1-5
                               5448 *
                               5449 * TEST FOR OVERFLOW OF THE FOUR BUFFER WORK AREA
                               5450 *
    125A 7D 04 FB              5451 LVI815 CLI   LVIPIN(,@BR),LVIBOF       OVRFLO 4TH BFR               1-5
    125D F2 81 0A              5452        JE    LVI820                    YES, WRITE THEM TO DISK
    1260 5E 00 FB 48           5453        ALC   LVIPIN(LVIBYC,@BR),LVIH01(,@BR)  INCR SECTOR CNT
    1264 76 02 4E              5454        A     LVIH64(,@BR),@XR          INCR TO NEXT BFR
    1267 F2 87 1D              5455        J     LVI830                    DECR ELEMENT CTR
                               5456 *
                               5457 * WRITE BUUFERS TO DISK AND MOVE OVERFLOW AREA TO BUFFER 1
                               5458 *
    126A C0 87 129B            5459 LVI820 B     LVI850                    GO TO PUT ROUTINE
    126E 5E 00 FA FB           5460        ALC   LVIPCT(1,@BR),LVIPIN(,@BR)  INCR PG BY SECTORS WRITTEN
    1272 0C 11 0711 0B11       5461        MVC   LVIBIO(LVIOBC),LVIBOA     MOVE OVERFLOW TO BFR 1
    1278 7C 01 FB              5462        MVI   LVIPIN(,@BR),LVID01       SECTOR COUNT
    127B 1C 00 1286 DA         5463        MVC   LVI828+@D1,LVICNT(1,@BR)  SET DISP FOR RESTORING ADDR
    1280 C2 02 0700            5464 LVI825 LA    LVIIB1,@XR                RESTORE BFR 1 ADDR
    1284 E2 02 00              5465 LVI828 LA    *-*(,@XR),@XR             RESTORE BASE ADDR OF BUFFERS
```

```
                               5466 *
                               5467 * DECREMENT ELEMENT COUNTER
                               5468 *
    1287 C0 87 1242            5469 LVI830 B     LVI812                    GO DECR ELEMENT COUNT        1-5
    128B 7C 01 F7              5470 LVI833 MVI   LVISWC(,@BR),LVIBYC       SET SWITCN = 1               1-5
    128E 34 02 121B            5471 LVI835 ST    LVI805+@OP1,@XR           SAVE NEXT AVAIL BYTE         1-5
    1292 1F 01 121B DA         5472        SLC   LVI805+@OP1,LVICNT(LVIBY2,@BR)  DECR BY VALUE LNG
    1297 C0 87 0000            5473 LVI840 B     *-*                       RETURN
                               5474 *
                               5475 * WRITE INITIAL1ZED BUFFERS TO VIRTUAL MEMORY
                               5476 *
    129B 34 08 12AE            5477 LVI850 ST    LVI860+@OP1,@ARR          SAVE RETURN ADDR
    129F C0 87 17E7            5478        B     DL4ICS                    IOCR RTN
    12A3 1417             12A4 5479        DC    AL2(LVIVMI)               ADDR DISK PARAM LIST
    12A5 C0 87 0025            5480        B     $DISKN                    WAIT FOR COMPLETION
    12A9 057F             12AA 5481        DC    AL2($WAITF)               COMPLETION PARAM
    12AB C0 87 0000            5482 LVI860 B     *-*                       RETURN
```

```
                               5484 ************************************************************************
                               5485 *                                                                      *
                               5486 *      LVINIT PUT SUBROUTINE                                           *
                               5487 *                                                                      *
                               5488 ************************************************************************
                               5489 *
                               5490 * SAVE REGISTERS
                               5491 *
    12AF 74 02 0B              5492 LVI900 ST    LVI950+@OP1(,@BR),@XR     SAVE PT
    12B2 74 08 0F              5493        ST    LVI955+@OP1(,@BR),@ARR    SAVE RETURN ADDR
                               5494 *
                               5495 * TEST IF NEEDED SECTOR IS IN THE CORE INPUT BUFFER
                               5496 *
    12B5 5D 00 D7 F3           5497 LVI910 CLC   LVIHLD(LVIBYC,@BR),LVIINN(,@BR)  IS PAGE IN CORE ?
    12B9 F2 81 1A              5498        JE    LVI930                    YES, MODIFY BFR
                               5499 *
                               5500 * PLACE PRESENT SECTOR IN VIRTUAL MEMORY AND GET THE REQUESTED SECTOR
                               5501 *
    12BC 5C 00 ED F3           5502 LVI920 MVC   LVIOUT(LVIBYC,@BR),LVIINN(,@BR)  SET PUT PG
    12C0 C0 87 17E7            5503        B     DL4ICS                    DISK IOCR RTN
    12C4 140A             12C5 5504        DC    AL(@CADDR)(LVIPUT)        ADDR DISK PARAM LIST
    12C6 5C 00 F3 D7           5505        MVC   LVIINN(LVIBYC,@BR),LVIHLD(,@BR)  SET GET PG
    12CA C0 87 17E7            5506        B     DL4ICS                    DISK IOCR ROUTINE
    12CE 1410             12CF 5507        DC    AL(@CADDR)(LVIGET)        ADDR DISK PARAM LIST
    12D0 C0 87 0025            5508        B     $DISKN                    WAIT FOR COMPLETION
    12D4 057F             12D5 5509        DC    AL(@CADDR)($WAITF)        WAIT PARAM
                               5510 *
                               5511 * MOVE VALUE TO THE BUFFER AND TEST FOR OVERFLOW
                               5512 *
    12D6 C2 02 0700            5513 LVI930 LA    LVIBF2,@XR                ADDR INPUT BFR
    12DA E2 02 00              5514 LVI933 LA    *-*(,@XR),@XR             INCR BY VALUE LNG
    12DD 8C 00 00 0000         5515 LVI935 MVC   *-*(@VQ,@XR),*-*          MOVE VALUE TO BFR
    12E2 0E 00 12DF 12DC       5516        ALC   LVI935+@D1,LVI933+@D1(1)  TEST FOR OVERFLOW
    12E8 F2 82 3C              5517        JL    LVI950                    NO, RETURN
                               5518 *
                               5519 * ON OVERFLOW WRITE PRESENT SECTOR TO VM AND GET NEXT CONTIGUOUS SECTOR
                               5520 *
    12EB 5C 00 ED F3           5521 LVI940 MVC   LVIOUT(LVIBYC,@BR),LVIINN(,@BR) SET PUT PG
    12EF C0 87 17E7            5522        B     DL4ICS                    DISK IOCR RTN
    12F3 140A             12F4 5523        DC    AL(@CADDR)(LVIPUT)        ADDR DISK PARAM LIST
    12F5 5E 00 F3 F4           5524        ALC   LVIINN(LVIBYC,@BR),LVISIN(,@BR) INCR TO NEXT PG
    12F9 C0 87 17E7            5525        B     DL4ICS                    DISK IOCR PIN
    12FD 1410             12FE 5526        DC    AL(@CADDR)(LVIGET)        ADDR DISK PARAM LIST
    12FF C0 87 0025            5527        B     $DISKN                    WAIT FOR COMPLETION
    1303 057F             1304 5528        DC    AL(@CADDR)($WAITF)        WAIT PARAM
                               5529 *
                               5530 * MOVE OVERFLOW TO INPUT BUFFER
                               5531 *
    1305 4C 00 01 12DF         5532        MVC   LVI945+@Q(1,@BR),LVI935+@D1   SET Q CODE LNG
    130A 4C 00 02 12DF         5533        MVC   LVI945+@D1(1,@BR),LVI935+@D1  BFR DISP FOR VALUE
    130F 4C 00 03 12DF         5534        MVC   LVI945+@DD2(1,@BR),LVI935+@D1  OVERFLOW AREA DISP
    1314 74 01 07              5535        ST    LVI948+@OP1(,@BR),@BR     SAVE PT
    1317 C2 02 0700            5536        LA    LVIBF2,@XR                ADDR INPUT BFR LH BYTE
    131B C2 01 0800            5537        LA    LVIBF2+256,@BR            ADDR OVRFLO AREA LH BYTE
    131F 9C 00 00 00           5538 LVI945 MVC   *-*(@VQ,@XR),*-*(,@BR)    MOVE OVRFLO TO BFR
    1323 C2 01 0000            5539 LVI948 LA    *-*,@BR                   RESTORE PT
```

```
                               5540 *
                               5541 * RESTORE REGISTERS AND RETURN TO CALLING PROGRAM
                               5542 *
    1327 C2 02 0000            5543 LVI950 LA    *-*,@XR                  RESTORE PT
    132B C0 87 0000            5544 LVI955 B     *-*                      RETURN TO CALLING PROGRAM
```

```
                               5546 ************************************************************************
                               5547 ************************************************************************
                               5548 *                                                                      *
                               5549 *      LVINIT ERROR SUBROUTINES                                        *
                               5550 *                                                                      *
                               5551 ************************************************************************
                               5552 ************************************************************************
                               5553 *
    132F 3C 3C 03CD            5554 LVI990 MVI   $CAERR,@@E250             SET ERROR CODE
    1333 F2 87 19              5555        J     LVI995                    GO TO SYS ERROR RTN
    1336 3C 40 03CD            5556 LVI991 MVI   $CAERR,@@E254             SET ERROR CODE
    133A F2 87 1F              5557        J     LVI997                    GO TO SYS ERROR RTN
    133D 3C 3F 03CD            5558 LVI992 MVI   $CAERR,@@E253             SET ERROR CODE
    1341 F2 87 0B              5559        J     LVI995                    GO TO SYS ERROR RTN
    1344 3C 42 03CD            5560 LVI993 MVI   $CAERR,@@E256             SET ERROR CODE
    1348 F2 87 04              5561        J     LVI995                    GO TO SYS ERROR RTN
                               5562 *
    134B 3C 3E 03CD            5563 LVI994 MVI   $CAERR,@@E252             SET ERROR CODE
    134F 0C FE 06FF 19FF       5564 LVI995 MVC   LVIPIB(LVIRLL),LVITRL+255  LIST TO PRIMARY INPUT BFR
    1355 C2 02 0000            5565 LVI996 LA    *-*,@XR                   RESTORE LINE PT
    1359 76 02 54              5566        A     LVIECC(,@BR),@XR          CONVERT PT TO NEW BFR PT
    135C 3C 80 03CE            5567 LVI997 MVI   $ERRPG,$ERKEY             SET INVALID LINE NO.
    1360 C0 87 0469            5568 LVI998 B     $CAERK                    ABORT LOADER, PRINT ERROR MSG
```

```
                               5570 ************************************************************************
                               5571 ************************************************************************
                               5572 *                                                                      *
                               5573 *      LVINIT CONSTANTS, WORK AREA AND EQUATES                         *
                               5574 *                                                                      *
                               5575 ************************************************************************
                               5576 ************************************************************************
                               5577 *
                               5578 * LVINIT EQUATES REFERENCING CONSTANTS
                               5579 *
                          0000 5580 LVITD0 EQU   0                         TRACE REF LIST DISP OF 0
                          0000 5581 LVINUL EQU   0                         NULL COMPARISON CODE
                          0000 5582 LVI0TD EQU   0                         TABLE DISP OF 0
                          0001 5583 LVI1TD EQU   1                         TABLE DISP OF 1
                          0001 5584 LVISWO EQU   1                         SET SWITCH FOR REGION 2
                          0001 5585 LVID01 EQU   1                         SET SECTOR COUNT
                          0001 5586 LVIBYC EQU   1                         LENGTH OF 1 BYTE
                          0001 5587 LVITD1 EQU   1                         TRACE REF LIST DISP OF 1
                          0002 5588 LVITD2 EQU   2                         TRACE REF LIST DISP OF 2
                          0002 5589 LVIBY2 EQU   2                         BYTES IN TWO BYTE COUNTER
                          0003 5590 LVITD3 EQU   3                         TRACE REF LIST DISP OF 3
                          0004 5591 LVIBOF EQU   4                         TO TEST FOR WORK AREA OVRFLO
                          0008 5592 LVIBDC EQU   8                         TO DETERMINE TRACE BYTE TO MASK
                          0009 5593 LVIDPT EQU   9                         TO RESET DIGIT PT
                          0008 5594 LVILUP EQU   B@LILP-1                  A CODE TO MOVE LONG PREC VALUE
                          0010 5595 LVICVM EQU   X'10'                     CHAR VAR TRACE MASK
                          0012 5596 LVIOBC EQU   18                        BYTE IN OVERFLOW AREA
                          0020 5597 LVILVM EQU   X'20'                     LETTER VAR TRACE MASK
                          0030 5598 LVIMKT EQU   X'30'                     TRACE MASK
                          0039 5599 LVITLL EQU   57                        LENGTH OF TRACE LIST
                          0080 5600 LVITMK EQU   X'80'                     TRACE BIT MASK
                          00F0 5601 LVIDNM EQU   X'F0'                     EBCDIC TO DECIMAL MASK FOR NUM
                          00FF 5602 LVIRLL EQU   255                       LNG OF LIST BFR TO SHIFT
                          06FF 5603 LVIPIB EQU   X'06FF'                   LAST BYTE PRIMARY INPUT BFR
                          0700 5604 LVIIB1 EQU   X'0700'                   1ST INIT BFR
                          1900 5605 LVIBF1 EQU   X'1900'                   TRACE REF LIST BUFFER ADDRESS
                          0700 5606 LVIBF2 EQU   X'0700'                   I/O INITIALIZATION BFR
                               5607 *
                               5608 * LVINIT CONSTANTS
                               5609 *
    1364 0000             1365 5610 LVIH00 DC    1XL2'00'                  ZERO FOR NULL CHECK
    1366 0001             1367 5611 LVIH01 DC    1XL2'01'                  TO INCR PTS BY ONE
    1368 0002             1369 5612 LVIH02 DC    1XL2'02'                  TO INCR PTS BY TWO
    136A 0003             136B 5613 LVIH03 DC    XL2'03'                   TO INCR PTS BY 3
    136C 0100             136D 5614 LVIH64 DC    XL2'0100'                 TO INCR BY BFR LNG
    136E 0700             136F 5615 LVIBRS DC    AL(@CADDR)(LVIIB1)        CADDR 1ST ARRAY INIT BFR
    1370 1F08             1371 5616 LVIAAC DC    AL(@CADDR)(B$LDRP+B@DL16+1) ARITH ARREY DOPE VECTOR
                               5617 *                                        * VIRTUAL TO CORE ADDR
                               5618 *                                        * CONVERSION CONSTANT
    1372 EE00             1373 5619 LVIECC DC    XL(@CADDR)'EE00'          REF LIST PT TO PRIMARY INPUT
                               5620 *                                      * BFR CONVERSION CONSTANT
    1374 13CD             1375 5621 LVIALC DC    AL2(LVINEL)               ADDR LONG PREC INTNL CONS
    1376 13D6             1377 5622 LVILAV DC    AL2(LVILPE)               ADDR LONG PREC ACTH VALUE
    1378 13E9             1379 5623 LVICHV DC    AL(@CADDR)(LVICMB)        ADDR CHAR VALUE
                               5624 *
                               5625 * SHORT PRECISION INTERNAL CONSTANTS
                               5626 *
    137A 0141421481       137E 5627 LVIS2S DC    XL(B@LISP)'0141421481'    CONSTANT FOR &SCR2
                               5628 *
    137F 0314159381       1383 5629 LVIP1S DC    XL(B@LISP)'0314159381'    CONSTANT FOR &PI
                               5630 *
    1384 0271828281       1388 5631 LVICES DC    XL(B@LISP)'0271828281'    CONSTANT FOR &E
                               5632 *
    1389 1141421481       138D 5633 LVIN2S DC    XL(B@LISP)'1141421481'    CONSTANT FOR -&SCR2
                               5634 *
    138E 1314159381       1392 5635 LVIN1S DC    XL(B@LISP)'1314159381'    CONSTANT FOR -&PI
                               5636 *
    1393 1271828281       1397 5637 LVINES DC    XL(B@LISP)'1271828281'    CONSTANT FOR -&E
                               5638 *
                               5639 * LONG PRECISION INTERNAL CONSTANTS
                               5640 *
    1398 2141421356237310 13A0 5641 LVIS2L DC    XL(B@LILP)'214142135623731081'  CONSTANT FOR &SQR2
                               5642 *
    13A1 2314159265358979 13A9 5643 LVIP1L DC    XL(B@LILP)'231415926535897981'  CONSTANT FOR &PI
                               5644 *
    13AA 2271828182845905 13B2 5645 LVICFL DC    XL(B@LILP)'227182818284590581'  CONSTANT FOR &E
                               5646 *
    13B3 3141421356237310 13BB 5647 LVIN2L DC    XL(B@LILP)'314142135623731081'  CONSTANT FOR -&SQR2, LONG
                               5648 *
    13BC 3314159265358979 13C4 5649 LVINIL DC    XL(B@LILP)'331415926535897981'  CONSTANT FOR -&PI, LONG
                               5650 *
    13C5 3271828182845905 13CD 5651 LVINEL DC    XL(B@LILP)'327182818284590581'  CONSTANT FOR -&E, LONG

                               5653 *
                               5654 * INITIALIZATION VALUE AREAS
                               5655 *
                          13CE 5656 LVISPS EQU   *                         1ST BYTE OF INIT VALUE
    13CE                  13D6 5657 LVILPE DS    CL9                       INITIALIZATION VALUE AREA,
    13CE                       5658        ORG   LVISPS                    * INIALLY SET WITH AN EXPONENT
    13CE 00000000         13D1 5659        DC    4XL1'00'                  * FOR BOTH LONG AND SHORT PREC
    13D2 1E0000001E       13D6 5660        DC    XL5'1E0000001E'           * SHORT ZEROED IN LONG PREC
                               5661 *
    13D7                  13D7 5662 LVICSB DS    CL1                       STATUS BYTE CHAR VALUE
    13D8 4040404040404040 13E9 5663 LVICMB DC    18XL1'40'                 REMAINDER OF CHAR VALUE

                               5665 *
                               5666 * TRACE TABLE MASKS FOR LETTER-DIGIT VARIABLES
                               5667 *
    13EA 80               13EA 5668 LVITM0 DC    XL1'80'                   TRACE MASK FOR DIGIT 0
    13EB 40               13EB 5669 LVITM1 DC    XL1'40'                   TRACE MASK FOR DIGIT 1
    13EC 20               13EC 5670 LVITM2 DC    XL1'20'                   TRACE MASK FOR DIGIT 2
    13ED 10               13ED 5671 LVITM3 DC    XL1'10'                   TRACE MASK ROR DIGIT 3
    13EE 08               13EE 5672 LVITM4 DC    XL1'08'                   TRACE MASK FOR DIGIT 4
    13EF 04               13EF 5673 LVITM5 DC    XL1'04'                   TRACE MASK FOR DIGIT 5
    13F0 02               13F0 5674 LVITM6 DC    XL1'02'                   TRACE MASK FOR DIGIT 6
    13F1 01               13F1 5675 LVITM7 DC    XL1'01'                   TRACE MASK FOR DIGIT 7
    13F2 80               13F2 5676 LVITM8 DC    XL1'80'                   TRACE MASK FOR DIGIT 8
    13F3 40               13F3 5677 LVITM9 DC    XL1'40'                   TRACE MASK FOR DIGIT 9
                               5678 *
                               5679 * TRACE TABLE MASKS FOR ARITHMETIC AND CHARACTER ARRAY VARIABLES
                               5680 *
                          0008 5681 LVIAAA EQU   X'08'                     TRACE MASK FOR ARITH 'ALL'
                          0004 5682 LVIAAP EQU   X'04'                     TRACE MASK FOR ARITH 'PARTIAL'
                          0002 5683 LVICAA EQU   X'02'                     TRACE MASK FOR CHAR 'ALL'
                          0001 5684 LVICAP EQU   X'01'                     TRACE MASK FOR CHAR 'PARTIAL'
                               5685 *
                               5686 * STATUS BYTE MASK MATES
                               5687 *
                          0020 5688 LVILTF EQU   B@PREC                    STATUS MASK FOR TRACE OFF, LONG
                          0040 5689 LVICTF EQU   B@DTYP                    STATUS MASK FOR TRACE OFF, CHAR
                          0080 5690 LVISTN EQU   B@TRAC                    STATUS MASK FOR TRACE ON, SHORT
                          00A0 5691 LVILTN EQU   B@TRAC+B@PREC             STATUS MASK FOR TRACE ON, LONG
                          00C0 5692 LVICTN EQU   B@TRAC+B@DTYP             STATUS MASK FOR TRACE ON, CHAR
                               5693 *
                               5694 * LVINIT WORK AREAS
                               5695 *
    13F4                  13F4 5696 LVITSW DS    CL1                       TRACE ALL SW
    13F5                  13F5 5697 LVIPLN DS    CL1                       CONTAINS THE LENGTH OF VALUE
    13F6                  13F6 5698 LVIHLD DS    CL1                       PG PARM FOR PUT RTN
    13F7                  13F7 5699 LVILSA DS    CL1                       LETTER SAVE AREA
    13F8                  13F9 5700 LVICNT DS    CL2                       SECTOR COUNTER
    13F8                       5701        ORG   *-2                       * INITIALLY CONTAINS THE
    13F8 0000             13F9 5702        DC    XL2'00'                   * VALUE ZERO
    13FA                  13FB 5703 LVIECT DS    CL2                       ELEMENT COUNT
    13FA                       5704        ORG   *-2                       * INITIALLY SET TO CONTAIN
    13FA 0000             13FB 5705        DC    XL2'0000'                 * ZEROS
    13FC                  13FD 5706 LVILDP DS    CL2                       LETTER-DIGIT PT
    13FE                  13FF 5707 LVIDSA DS    CL2                       DIGIT SAVE AREA
    13FE                       5708        ORG   *-2                       * INITIALLY SET TO
    13FE 0000             13FF 5709        DC    2XL1'00'                  * ZERO
    1400                  1401 5710 LVICTR DS    CL2                       AREA USED TO DETERMINE
    1400                       5711        ORG   *-2                       * THE DISP TO 1ST BYTE OF
    1400 0038             1401 5712        DC    XL2'38'                   * THE NEEDED DIGIT
    1402                  1403 5713 LVIELC DS    CL2                       ELEMENT CTR FOR PUT RTN
    1404                  1405 5714 LVIAIV DS    CL(@CADDR)                ADDRESS OF THE ARITHMETIC
    1404                       5715        ORG   *-@CADDR                  * VALUE, INITIALLY SET TO THE
    1404 13D2             1405 5716        DC    AL(@CADDR)(LVISPM)        * SHORT PRECISION VALUE
    1406                  1407 5717 LVISS1 DS    CL2                       SAVE AREA 1ST BINARY SUBSC
    1408                  1409 5718 LVISS2 DS    CL2                       SAVE AREA 2ND BINARY SUBSC
                               5719 *
                               5720 * LVINIT DISK PARAMETER LIST
                               5721 *
                          140A 5722 LVIPUT EQU   *                         ADDR DISK PARM LIST
    140A 02               140A 5723        DC    AL1(@DPUT)                WRITE CODE
    140B 07               140B 5724        DC    AL1(@DVBCY)               BASE CYL FOR VM
    140C                  140C 5725 LVIOUT DS    1CL1                      SECTOR DISP FROM BASE CYL
    140D 01               140D 5726        DC    1XL1'01'                  SECTOR CNT
    140E 0700             140F 5727        DC    AL2(LVIBF2)               ADDR CORE OUTPUT AREA
                               5728 *
                          1410 5729 LVIGET EQU    *                        ADDR DISK PARM LIST
    1410 01               1410 5730        DC    AL1(@DGET)                READ CODE
    1411 07               1411 5731        DC    AL1(@DVBCY)               BASE CIL FOR VM
    1412                  1412 5732 LVIINN DS    1CL1                      SECTOR DISP FROM THE BASE
    1412                       5733        ORG   LVIINN                    * CYL, INITIALLY SET
    1412 00               1412 5734        DC    1XL1'00'                  * TO ZERO
    1413 01               1413 5735 LVISIN DC    1XL1'01'                  SECTOR LNT
    1414 0700             1415 5736        DC    AL2(LVIBF2)               ADDR CORE INPUT AREA
                               5737 *
    1416                  1416 5738 LVISWC DS    XL1                                                     1-5
                          1417 5739 LVIVMI EQU   *                         ADDR DISK PARAM LIST
    1417 02               1417 5740        DC    AL1(@DPUT)                WRITE CODE
    1418 07               1418 5741        DC    AL1(@DVBCY)               BASE CYL FOR VM
    1419                  1419 5742 LVIPCT DS    CL1                       SECTOR DISP
    141A                  141A 5743 LVIPIN DS    CL1                       NUMBER OF SECTORS TO WRITE
    141A                       5744        ORG   LVIPIN                    * INITIALLY SET
    141A 01               141A 5745        DC    XL1'01'                   * TO ONE
    141B 0700             141C 5746        DC    AL2(LVIIB1)               ADDR CORE OUTPUT AREA
                               5747 *
                               5748 * ALPHA TABLE
                               5749 *
                          141D 5750 LVIATL EQU   *                         ADDR 1ST BYTE ALPHA TABLE
    141D 5B7B7CC1C2C3C4C5 1439 5751        DC    1CL29'$#@ABCDEFGHIJKLMNOPORSTUVWXYZ'  ALPHA TABLE
                               5752 *
                               5753 * TRACE LIST
                               5754 *
                          143A 5755 LVIPTL EQU   *                         ADDR 1ST BYTE TRACE LIST
    143A                  1473 5756        DS    58CL1                     TRACE LIST, CONTAINS BIT SW
    143A                       5757        ORG   LVIPTL                    * INITIALLY SET WITH ALL LETTER
    143A FFF0FFF0FFF0FFF0 1473 5758        DC    29XL2'FFF0'               * LETTER-DIGIT AND CHAR SW ON
                               5759 *
                               5760 * LVINIT EQUATES REFERENCING PROGRAM
                               5761 *
                          1A00 5762 LVIVA1 EQU   B$LDRP                    1ST PG NO. REGION 1
                          1A03 5763 LVIRG1 EQU   B$LDRP+B@DL02             ADDR LAST PG REGION 1
                          1A08 5764 LVIICP EQU   B$LDRP+B@DL05-1           PG INTERNAL CONSTANTS
                          1A0B 5765 LVIIVD EQU   B$LDRP+B@DL06             DISP TO 1ST INTERNAL VAR IN PG
                          1A04 5766 LVIVA2 EQU   B$LDRP+4                  CADDR 1ST PG REGION 2
                          1A0C 5767 LVILVT EQU   B$LDRP+B@DL06+1           ADDR 1ST ENTRY LVT
                          1A46 5768 LVILDT EQU   B$LDRP+B@DL07+1           ADDR 1ST ENTRY LDT
                          1C8A 5769 LVICVT EQU   B$LDRP+B@DL10+1           ADDR 1ST ENTRY CVT
                          1CC4 5770 LVINAT EQU   B$LDRP+B@DL11+1           ADDR 1ST ENTRY NAT
                          1CFE 5771 LVICAT EQU   B$LDRP+B@DL12+1           ADDR 1ST ENTRY CAT
                          0006 5772 LVIDVP EQU   B@ABAS-1                  D/V DISP TO PG NO.
                          0B23 5773 LVITT1 EQU   LVI094+@D1                DISP IN TRACE TBL FOR MASK
                          0BF9 5774 LVITT2 EQU   LVI132+@D1                DISP TO CHAR TBL ENTRY FIELD
                          0C41 5775 LVITT3 EQU   LVI153+@D1                DISP TO NVM ARRAY ENTRY FIELD
                          0CB4 5776 LVITT4 EQU   LVI182+@D1                DISP TO CHAR ARRAY ENTRY FIELD
                          1C88 5777 LVILET EQU   LVICVT-2                  LAST BYTE LETTER-DIGIT TBL
                          1397 5778 LVIASC EQU   LVINES                    ADDR SHORT PREC INTNL CONS
                          1900 5779 LVITRL EQU   LVIBF1                    1ST BYTE IN TRACE REF LIST
                          0B11 5780 LVIBOA EQU   LVIIB1+1041               LAST BYTE OVERFLOW AREA
                          0711 5781 LVIBIO EQU   LVIIB1+17                 BFR ADDR TO MOVE OVRFLO TO
                          1473 5782 LVILTB EQU   LVIPTL+57                 LAST BYTE TRACE LIST
                          13D2 5783 LVISPM EQU   LVISPS+4                  LAST BYTE SHORT PREC VALUE
                               5784 *
                               5786 ***********************************************************************
                               5787 * 5703-XM1   COPYRIGHT IBM CORP 1970                                  *
                               5788 *            REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083      *
                               5789 *                                                                     *
                               5790 ***********************************************************************
                               5791 *STATUS -                                                             *
                               5792 *   VERSION 1 MODIFICATION 0                                          *
                               5793 *                                                                     *
                               5794 *FUNCTION -                                                           *
                               5795 *   * LRADDR RESOLVES THE REMAINING UNKNOWN PSUEDO INSTRUCTION        *
                               5796 *     VIRTUAL ADDRESS OPERANDS THAT THE COMPILER WAS UNABLE TO        *
                               5797 *     RESOLVE.  THIS IS DONE BY PASSING THE BRANCH TABLE AGAINST      *
                               5798 *     A STATEMENT NUMBER TABLE.                                       *
                               5799 *   * LRADDR PLACES THE COMPILER COMMON PARAMETER AREA TO DISK        *
                               5800 *                                                                     *
                               5801 *ENTRY POINTS -                                                       *
                               5802 *   LRADDR HAS ONLY ONE ENTRY POINT                                   *
                               5803 *   THE CALLING SEQUENCE IS:                                          *
                               5804 *        B     LRADDR                                                 *
                               5805 *                                                                     *
                               5806 *INPUT -                                                              *
                               5807 *   * BRANCH ADDRESS TABLE - (1-16 SECTORS), CONTAINS 64 4-BYTE       *
                               5808 *     ENTRIES PER SECTOR                                              *
                               5809 *   * STATEMENT TABLE - (1-16 SECTORS), CONTAINS 64 4-BYTE ENTRIES    *
                               5810 *     PER SECTOR                                                      *
                               5811 *   * VIRTUAL MEMORY - PMC GENERATED BY THE COMPILER                  *
                               5812 *   * COMPILER COMMON PARAMETER AREA                                  *
                               5813 *                                                                     *
                               5814 *OUTPUT -                                                             *
                               5815 *   * VIRTUAL MEMORY - LRADDR CAUSES MODIFICATION OF PSUEDO MACHINE   *
                               5816 *     CODE AREA UNDER CERTAIN CONDITIONS                              *
                               5817 *   * COMPILER COMMON PARAMETER AREA                                  *
                               5818 *      * LALVA1                                                       *
                               5819 *      * LALVA2                                                       *
                               5820 *      * LALVA3                                                       *
                               5821 *      * LALVA4                                                       *
                               5822 *      * LVIICA                                                       *
                               5823 *      * LVIIVA                                                       *
                               5824 *      * LETTER VARIABLE TABLE (LVT)                                  *
                               5825 *      * LETTER DIGIT TABLE                                           *
                               5826 *      * CHARACTER VARIABLE TABLE (CVT)                               *
                               5827 *      * ARITHMETIC ARRAY TABLE                                       *
                               5828 *      * CHARACTER ARRAY TABLE (CAT)                                  *
                               5829 *      * FUNCTION AND ARRAY TABLE (FAT)                               *
                               5830 *                                                                     *
                               5831 *EXTERNAL REFERENCES -                                                *
                               5832 *        DL4ICS - 4-TRACK LIOCS                                       *
                               5833 *        $DISKN - SYSTEM DISK IOCR                                    *
                               5834 *        LSORTA - LOADER ADDRESS SORT ROUTINE                         *
                               5835 *        LALLOC - LOADER ARRAY ALLOCATION                             *
                               5836 *                                                                     *
                               5837 *EXITS, NORMAL -                                                      *
                               5838 *   LRADDR HAS ONE NORMAL EXIT                                        *
                               5839 *      LAL000 - AFTER BRANCH ADDRESS RESOLUTION                       *
                               5840 *                                                                     *
                               5841 *EXITS, ERROR                                                         *
                               5842 *   N/A                                                               *
                               5843 *                                                                     *
                               5844 *TABLES/WORK AREAS -                                                  *
                               5845 *   * THE CONSTANTS AND WORK AREAS RESIDE AT THE END OF EXECUTION     *
                               5846 *     CODE                                                            *
                               5847 *   * BUFFER 1 - 256 BYTES, FOR BRANCH TABLE                          *
                               5848 *   * BUFFER 2 - 256 BYTES, USED FOR                                  *
                               5849 *      * STATEMENT TABLE                                              *
                               5850 *      * VIRTUAL MEMORY SECTOR                                        *
                               5851 *                                                                     *
                               5852 *ATTRIBUTES -                                                         *
                               5853 *   N/A                                                               *
                               5854 *                                                                     *
                               5855 *CHARACTER CODE DEPENDENCY -                                          *
                               5856 *   N/A                                                               *
                               5857 *                                                                     *
                               5858 *NOTES -                                                              *
                               5859 *   ERROR PROCEDURES                                                  *
                               5860 *      N/A                                                            *
                               5861 *                                                                     *
                               5862 *   REGISTER USAGE                                                    *
                               5863 *      * BOTH REGISTERS ARE USED DURING EXECUTION                     *
                               5864 *      * THE REGISTERS ARE NOT SAVED OR RESTORED                      *
                               5865 *                                                                     *
                               5866 *   SAVED/RESTORED AREAS                                              *
                               5867 *      N/A                                                            *
                               5868 *                                                                     *
                               5869 *   MODIFICATION CONSIDERATIONS                                       *
                               5870 *      N/A                                                            *
                               5871 *                                                                     *
                               5872 *   REQUIRED MODULES                                                  *
                               5873 *        @SYSEQ - COMMON SYSTEM EQUATES                               *
                               5874 *        @VMDEQ - VM DIRECTORY EQUATES                                *
                               5875 *        $B$EQU - COMPILER SYSTEM EQUATES                             *
                               5876 *        DL4ICS - 4-TRACK LIOCS                                       *
                               5877 *        LSORTA - LOADER ADDRESS SORT ROUTINE                         *
                               5878 *        LALLOC - LOADER ARRAY ALLOCATION                             *
                               5879 *        @FXDEQ - SYSTEM NUCLEUS ADDRESSES AND INDICATORS             *
                               5880 *                                                                     *
                               5881 *   OTHER                                                             *
                               5882 *      N/A                                                            *
                               5883 ***********************************************************************
                               5885 ************************************************************************
                               5886 *                                                                      *
                               5887 *      CONVERT BRANCH TABLE LINE NO. ENTRIES TO CORRESPONDING VADDRS  *
                               5888 *                                                                      *
                               5889 ************************************************************************
                               5890 *
                               5891 * READ A BRANCH TABLE SECTOR
                               5892 *
    1474 3C 50 1610            5893 LRADDR MVI   LRAPNO,@DCBT1             1ST BR TABLE SECTOR
    1478 C0 87 17E7            5894 LRA010 B     DL4ICS                    TO DISK IOCR FOR GET
    147C 160E             147D 5895        DC    AL(@CADDR)(LRAPLB)        ADDR DISK PARM LIST
    147E C0 87 0025            5896        B     $DISKN                    WAIT FOR COMPLETION
    1482 057F             1483 5897        DC    AL(@CADDR)($WAITF)        WAIT PARM
    1484 C0 87 048D            5898        B     $UNMSK                    UNMASK THE LOADER AT THIS POINT
                               5899 *
                               5900 * CHECK FOR NULL BRANCH TABLE SECTOR
                               5901 *
    1488 0D 01 06A1 1604       5902 LRA012 CLC   LRABEQ(LRABCT),LRAH00     IS SECTOR NULL ?
    148E C0 81 15F2            5903        BE    LRA280                    SAVE LOADER PARAMETERS
                               5904 *
                               5905 * READ FIRST SECTOR OF STATEMENT TABLE
                               5906 *
    1492 3C 40 1616            5907 LRA015 MVI   LRASPG,@DCST1             1ST STMT TABLE SECTOR
    1496 C0 87 17E7            5908        B     DL4ICS                    TO DISK IOCR FOR GET
    149A 1614             149B 5909        DC    AL(@CADDR)(LRAPLS)        ADDR DISK PARM LIST
                               5910 *
                               5911 * DETERMINE END OF TABLE ENTRIES IN SECTOR
                               5912 *
    149C 3C FC 14A6            5913        MVI   LRA020+@D1,LRALST         PRESET LAST PG NO. AS DISP
    14A0 C2 02 06A0            5914        LA    LRABB1,@XR                GET ADDR BR TABLE BFR
    14A4 BD 00 00              5915 LRA020 CLI   *-*(,@XR),LRAX00          IS ENTRY 0
    14A7 F2 01 19              5916        JNE   LRA030                    NO, INITLZ SORT RTN
    14AA 0F 00 14A6 1608       5917        SLC   LRA020+@D1(1),LRAP04      DECK DISP 1 ENTRY
    14B0 C0 84 14A4            5918        BH    LRA020                    RECYCLE UNTIL
    14B4 3A 07 154C            5919        SBN   LRABSW,LRABMK             SET BRANCH SWITCH ON
    14B8 C2 02 06A0            5920        LA    LRABB1,@XR                CADDR LAST ENTRY IN 1 ENTRY BFR
    14BC 34 02 16D3            5921        ST    LSOBOT,@XR                SAVE LAST ADDR PARM
    14C0 F2 87 1D              5922        J     LRA037                    BRANCH AROUND FIRST SORT
                               5923 *
                               5924 * SET LSORTA SORT PARAMETERS FOR FIRST SORT MODE
                               5925 *
    14C3 0C 00 14CB 14A6       5926 LRA030 MVC   LRA035+@D1(1),LRA020+@D1  DISP TO LAST ENTRY
    14C9 E2 02 00              5927 LRA035 LA    *-*(,@XR),@XR             ADDR OF LAST ENTRY
    14CC 36 02 160A            5928        A     LRAN04,@XR                PT TO 2ND LAST ENTRY
    14D0 34 02 16D3            5929        ST    LSOBOT,@XR                SET LSORTA PARM
    14D4 C2 02 06A0            5930        LA    LRABB1,@XR                ADDR 1ST ENTRY
    14D8 C2 01 06A0            5931        LA    LRABB1,@BR                ADDR 1ST ENTRY
    14DC C0 87 162C            5932        B     LSORTA                    SORT BFR
                               5933 *
                               5934 * WAIT FOR STATEMENT TABLE SECTOR READ COMPLETION
                               5935 *
    14E0 C0 87 0025            5936 LRA037 B     $DISKN                    WAIT FOR READ COMPLETION
    14E4 057F             14E5 5937        DC    AL(@CADDR)($WAITF)        WAIT PARM
                               5938 *
                               5939 * INITIALIZE THE BUFFER POINTERS
                               5940 *
    14E6 C2 02 06A0            5941 LRA040 LA    LRABB1,@XR                ADDR 1ST ENTRY BRABFR
    14EA C2 01 1900            5942 LRA045 LA    LRASB1,@BR                ADDR 1ST ENTRY STMT BFR
                               5943 *
                               5944 * CHECK IF LINE NO. IS IN THE PRESENT STATEMENT SECTOR
                               5945 *
    14EE 2D 01 19FF 03         5946 LRA050 CLC   LRASBE,LRAPGD(LRAACT,@XR)  LINE NO. IN THIS SECTOR ?
    14F3 F2 82 2F              5947        JL    LRA110                    NO, CHECK NEXT SECTOR
                               5948 *
                               5949 * DETERMINE IF A VIRTUAL ADDRESS IS IN LINE NO. POSITION OF BRANCH BFR
                               5950 *
    14F6 BD 56 02              5951 LRA060 CLI   LRADPG(,@XR),@VENTA       LINE NO. A VADDR ?
    14F9 F2 02 4F              5952        JNL   LRA200                    YES, BEGIN NEXT SORT
                               5953 *
                               5954 * CHECK BRANCH TABLE ENTRY LINE NO. FOR MATCH IN STATEMENT TABLE
                               5955 *
    14FC 6D 01 03 03           5956 LRA070 CLC   LRAPGD(LRAACT,@BR),LRAPGD(,@XR)  LINE NOS. EQUAL
    1500 F2 84 40              5957        JH    LRA130                    HIGH, ZERO BRANCH ADDR
    1503 F2 82 18              5958        JL    LRA100                    LOW, CHECK NEXT ENTRY
    1506 9C 01 03 01           5959 LRA080 MVC   LRAPGD(LRAACT,@XR),LRASVA(,@BR)  TRANSFER VADDR
    150A 34 02 160D            5960 LRA090 ST    LRASAV,@XR                SAVE @XR FOR COMPARE
    150E 0D 01 16D3 160D       5961        CLC   LSOBOT(LRAACT),LRASAV     AT LAST ENTRY
    1514 F2 82 34              5962        JL    LRA200                    YES, BEGIN NEXT SORT
    1517 E2 02 04              5963        LA    LRAINC(,@XR),@XR          INCR TO NEXT ENTRY
    151A C0 87 14EE            5964        B     LRA050                    PROCESS NEXT ENTRY
                               5965 *
                               5966 * INCREMENT STATEMENT TABLE POINTER ONE ENTRY
                               5967 *
    151E D2 01 04              5968 LRA100 LA    LRAINC(,@BR),@BR          INCR TO NEXT ENTRY
    1521 C0 87 14F6            5969        B     LRA060                    CHECK ENTRY FOR MATCH
                               5970 *
                               5971 * REPLACE PRESENT STATEMENT TABLE SECTOR WITH NEXT CONTIGUOUS STATEMENT
                               5972 * TABLE SECTOR
                               5973 *
    1525 0E 00 1616 1611       5974 LRA110 ALC   LRASPG(1),LRAPCT          INCR TABLE PG DISP
    152B 34 02 1538            5975        ST    LRA120+@OP1,@XR           SAVE BFR PT
    152F C0 87 17E7            5976        B     DL4ICS                    IOCR RTN
    1533 1614             1534 5977        DC    AL(@CADDR)(LRAPLS)        ADDR DISK PARM LIST
    1535 C2 02 0000            5978 LRA120 LA    *-*,@XR                   RESTORE PT
    1539 C0 87 0025            5979        B     $DISKN                    WAIT FOR READ COMPLETION
    153D 057F             153E 5980        DC    AL(@CADDR)($WAITF)        WAIT PARM
    153F C0 87 14EA            5981        B     LRA045                    CHECK IF ENTRY IN THIS SECTOR
                               5982 *
                               5983 * ZERO THE UNRESOLVED VIRTUAL ADDRESS POSITION
                               5984 *
    1543 AF 01 03 03           5985 LRA130 SLC   LRAPGD(LRAACT,@XR),LRAPGD(,@XR)  ZERO VADDR POSITION
    1547 C0 87 150A            5986        B     LRA090                    CONTINUE LOOP
```

```
                               5988 ***********************************************************************
                               5989 *                                                                     *
                               5990 *      LOAD UNRESOLVED PSUEDO CODE ADDRESSES INTO VIRTUAL MEMORY      *
                               5991 *                                                                     *
                               5992 ***********************************************************************
                               5993 *
                               5994 * SET LSORTA SORT PARAMETERS FOR SECOND SORT MODE
                               5995 *
    154B F2 00 12              5996 LRA200 JC    LRA210,*-*                BRANCH IF ONLY ONE ENTRY
    154C                       5997        ORG   LRA200+@Q                 INITIALIZE BRANCH
    154C 80               154C 5998        DC    AL1(@NOP)                 SWITCH TO NOT
    154E                       5999        ORG   LRA200+3                  * BRANCH
    154E 0F 01 16D3 1606       6000        SLC   LSOBOT(@CADDR),LRAX02     DECR SORT TERMINATION ADDR
    1554 C2 02 069E            6001        LA    LRABB1-LRAN02,@XR         ADDR 1ST ENTRY -2
    1558 C2 01 06A0            6002        LA    LRABB1,@BR                ADDR 1ST ENTRY
    155C C0 87 162C            6003        B     LSORTA                    SORT BFR
                               6004 *
                               6005 * DETERMINE AND READ THE DESIRED SECTOR OF VIRTUAL MEMORY
                               6006 *
    1560 3B 07 154C            6007 LRA210 SBF   LRABSW,LRABMK             SET BRANCH SW OFF
    1564 C2 01 06A0            6008        LA    LRABB1,@BR                ADDR FIRST ENTRY
    1568 0F 00 160B 160B       6009        SLC   LRACTR(1),LRACTR          ZERO CTR
    156E 1C 00 161C 00         6010 LRA220 MVC   LRAVPG,LRAPDP(1,@BR)      SET PG NO, IN LIST
    1573 C0 87 17E7            6011        B     DL4ICS                    DISK IOCR
    1577 161A             1578 6012        DC    AL(@CADDR)(LRAPLV)        ADDR DISK PARM LIST
    1579 C0 87 0025            6013        B     $DISKN                    WAIT FOR READ COMPLETION
    157D 057F             157E 6014        DC    AL(@CADDR)($WAITF)        WAIT PARM
    157F C2 02 1900            6015        LA    LRASB1,@XR                ADDR PSUEDO CODE BFR
                               6016 *
                               6017 * INCREMENT ENTRY COUNTER
                               6018 *
    1583 0E 00 160B 1602       6019 LRA230 ALC   LRACTR(1),LRACIN          INCR ENTRY CTR
                               6020 *
                               6021 * MOVE THE VM ADDR FROM THE BRANCH TABLE TO THE UNRESOLVED
                               6022 * ADDR IN THE PSUEDO CODE BFR
                               6023 *
    1589 1C 00 1590 01         6024        MVC   LRA240+@D1,LRADIS(1,@BR)  PLACE BFR DISP IN MOVE INST
    158E 9C 01 00 03           6025 LRA240 MVC   *-*(LRAACT,@XR),LRAVMD(,@BR)  TRANSFER ADDR
                               6026 *
                               6027 * TEST IF ALL ENTRIES HAVE BEEN PROCESSED
                               6028 *
    1592 3D 00 160B            6029        CLI   LRACTR,LRACT0            IS CTR 0 ?
    1596 F2 81 3D              6030        JE    LRA260                   WRITE TO DISK
                               6031 *
                               6032 * INCREMENT TO NEXT ENTRY IN THE BRANCH TABLE AND CHECK FOR THE
                               6033 * CORRECT SECTOR OF PSUEDO CODE IN THE BUFFER
                               6034 *
    1599 D2 01 04              6035        LA    LRAINC(,@BR),@BR         INCR TO NEXT ENTRY
    159C 7D 56 00              6036        CLI   LRAPDP(,@BR),@VENTA      LT 1ST PSUEDO CODE PG NO.
    159F F2 82 1F              6037        JL    LRA250                   YES, WRITE PROCESSED BFR
    15A2 1D 00 161C 00         6038        CLC   LRAVPG,LRAPDP(1,@BR)     IS THIS PG NO. ALREADY IN CORE ?
    15A7 C0 81 1583            6039        BE    LRA230                   YES, PROCESS ENTRY
                               6040 *
                               6041 * PLACE PRESENT UNWANTED VM SECTOR IN VIRTUAL MEMORY
                               6042 *
    15AB 0C 00 1622 161C       6043        MVC   LRAPGV(1),LRAVPG         MOVE PRESENT SECTOR NO. TO DPL
    15B1 C0 87 17E7             6044        B     DL4ICS                   IOCR RTN FOR PUT
    15B5 1620                15B6 6045        DC    AL(@CADDR)(LRAVPL)       ADDR DISK PARM LIST
    15B7 C0 87 0025             6046        B     $DISKN                   WAIT FOR COMPLETION
    15BB 057F                15BC 6047        DC    AL(@CADDR)($WAITF)       WAIT PARM
    15BD C0 87 156E             6048        B     LRA220                   PROCESS NEXT ENTRY
                                6049 *
                                6050 * PLACE LAST VM SECTOR TO VIRTUAL MEMORY AND EXIT LRADDR
                                6051 *
    15C1 0C 00 1622 161C        6052 LRA250 MVC   LRAPGV(1),LRAVPG         MOVE PRESENT SECTOR NO. TO DPL
    15C7 C0 87 17E7             6053        B     DL4ICS                   IOCR RTN FOR PUT
    15CB 1620                15CC 6054        DC    AL(@CADDR)(LRAVPL)       ADDR DISK PARM LIST
    15CD C0 87 0025             6055        B     $DISKN                   WAIT FOR COMPLETION
    15D1 057F                15D2 6056        DC    AL(@CADDR)($WAITF)       WAIT PARM
    15D3 F2 87 1C               6057        J     LRA280                   SAVE LOADER PARAMETERS
                                6058 *
                                6059 * PLACE PRESENT VM SECTOR IN VIRTUAL MEMORY PREPARATORY TO ACCESSING
                                6060 * THE NEXT BRANCH TABLE SECTOR
                                6061 *
    15D6 0C 00 1622 161C        6062 LRA260 MVC   LRAPGV(1),LRAVPG         MOVE PRESENT SECTOR NO. TO DPL
    15DC C0 87 17E7             6063        B     DL4ICS                   IOCR RTN FOR PUT
    15E0 1620                15E1 6064        DC    AL(@CADDR)(LRAVPL)       ADDR DISK PARM LIST
    15E2 C0 87 0025             6065        B     $DISKN                   WAIT FOR COMPLETION
    15E6 057F                15E7 6066        DC    AL(@CADDR)($WAITF)       WAIT PARM
                                6067 *
                                6068 * INCREMENT THE SECTOR COUNT TO THE NEXT PAGE OF THE BRANCH TABLE
                                6069 * AND BEGIN LRADDR PROCESSING
                                6070 *
    15E8 0E 00 1610 1611        6071 LRA270 ALC   LRAPNO(1),LRAPCT         INCR TO NEXT PG IN BRATBL
    15EE C0 87 1478             6072        B     LRA010                   PROCESS THAT SECTOR
                                6073 *
                                6074 * MOVE THE LOADER PARAMETERS TO THE STATEMENT TABLE FOR LATTER USE
                                6075 *
    15F2 C0 87 17E7             6076 LRA280 B     DL4ICS                   IOCR RTN FOR PUT
    15F6 1626                15F7 6077        DC    AL(@CADDR)(LRAPUT)       ADDR DISK PARM LIST
    15F8 C0 87 0025             6078        B     $DISKN                   WAIT FOR COMPLETION
    15FC 057F                15FD 6079        DC    AL(@CADDR)($WAITF)       WAIT PARM
    15FE C0 87 060B             6080        B     LAL000                   TO VM FUNCTION LOAD
```

```
                               6082 ************************************************************************
                               6083 ************************************************************************
                               6084 *                                                                      *
                               6085 *      LRADDR CONSTANTS, WORK AREAS AND EQUATES                        *
                               6086 *                                                                      *
                               6087 ************************************************************************
                               6088 ************************************************************************
                               6089 *
                               6090 * LRADDR EQUATES REFERENCING PROGRAM
                               6091 *
                          154C 6092 LRABSW EQU   LRA200+@Q                 BRANCH AROUND SORT SWITCH
                               6093 *
                               6094 * LRADDR EQUATES REFERENCING CONSTANTS
                               6095 *
                          0000 6096 LRAX00 EQU   0                         ZERO FOR COMPARISONS
                          0000 6097 LRABMT EQU   0                         CHECK FOR NULL ENTRY
                          0000 6098 LRACT0 EQU   0                         TO ZERO ENTRY CTR
                          0000 6099 LRAPDP EQU   0                         PG NO. DISP IN VADDR
                          0001 6100 LRADIS EQU   1                         SECTOR DISP IN ENTRY
                          0001 6101 LRASVA EQU   1                         DISP VADDR IN STMT TABLE
                          0002 6102 LRAN02 EQU   2                         DISP TO DECR 2
                          0002 6103 LRAACT EQU   2                         BYTES IN PG NO.
                          0002 6104 LRABCT EQU   2                         BYTES IN AN ENTRY ARGUMENT
                          0002 6105 LRADPG EQU   2                         LINE NO. PG DISP IN BR TBL
                          0003 6106 LRAVMD EQU   3                         VADDR DISP IN ENTRY
                          0003 6107 LRAPGD EQU   3                         PG NO. DISP IN ENTRY
                          0004 6108 LRAINC EQU   4                         DISP BETWEEN ENTRIES
                          0007 6109 LRABMK EQU   X'07'                     MASK FOR BRANCH SW
                          00FC 6110 LRALST EQU   X'FC'                     LAST PG NO. IN A SECTOR
                          06A0 6111 LRABB1 EQU   X'06A0'                   LH BYTE BRANCH TABLE BUFFER
                          1900 6112 LRASB1 EQU   X'1900'                   LH BYTE STMT TBL BFR
                          19FF 6113 LRASBE EQU   X'19FF'                   RH BYTE STMT TBL BFR
                          06A1 6114 LRABEQ EQU   LRABB1+1                  TO CHECK OR A NULL SECTOR
                               6115 *
                               6116 * LRADDR CONSTANTS
                               6117 *
    1602 04               1602 6118 LRACIN DC    XL1'04'                   COUNTER INCR
    1603 0000             1604 6119 LRAH00 DC    XL2'0000'                 TO CHICK FOR NULL BFR
    1605 0002             1606 6120 LRAX02 DC    XL2'0002'                 CONSTANT TO DECR 2
    1607 0004             1608 6121 LRAP04 DC    XL2'0004'                 CONSTANT TO DECR DISP
    1609 FFFC             160A 6122 LRAN04 DC    XL2'FFFC'                 DISP TO DECR 4
                               6123 *
                               6124 * LRADDR WORK AREAS
                               6125 *
    160B                  160B 6126 LRACTR DS    CL1                       END OF ENTRIES CTR
    160C                  160D 6127 LRASAV DS    CL2                       HOLD FOR XR TO COMPARE IT
                               6128 *
                               6129 * LRADDR DISK PARAMETER LISTS
                               6130 *
                          160E 6131 LRAPLB EQU   *                         ADDR DISK PARM LIST
    160E 01               160E 6132        DC    AL1(@DGET)                READ CODE
    160F 09               160F 6133        DC    AL1(@DCBCY)               BASE CYL FOR TABLES
    1610                  1610 6134 LRAPNO DS    CL1                       PG NO.
    1611 01               1611 6135 LRAPCT DC    XL1'01'                   SECTORS TO READ
    1612 06A0             1613 6136        DC    AL2(LRABB1)               ADDR DISK INPUT AREA
                               6137 *
                          1614 6138 LRAPLS EQU   *                         ADDR DISK PARM LIST
    1614 01               1614 6139        DC    AL1(@DGET)                READ CODE
    1615 09               1615 6140        DC    AL1(@DCBCY)               BASE CYL FOR TABLES
    1616                  1616 6141 LRASPG DS    CL1                       PG OF STMT TABLE
    1617 01               1617 6142        DC    XL1'01'                   SECTORS TO READ
    1618 1900             1619 6143        DC    AL2(LRASB1)               ADDR CORE INPUT AREA
                               6144 *
                          161A 6145 LRAPLV EQU   *                         ADDR DISK PARM LIST
    161A 01               161A 6146        DC    AL1(@DGET)                READ CODE
    161B 07               161B 6147        DC    AL1(@DVBCY)               BASE CYL FOR VM
    161C                  161C 6148 LRAVPG DS    CL1                       SECTOR ADDR IN VM
    161D 01               161D 6149        DC    XL1'01'                   SECTORS TO READ
    161E 1900             161F 6150        DC    AL2(LRASB1)               ADDR CORE INPUT AREA
                               6151 *
                          1620 6152 LRAVPL EQU   *                         ADDR DISK PARM LIST
    1620 02               1620 6153        DC    AL1(@DPUT)                WRITE CODE
    1621 07               1621 6154        DC    AL1(@DVBCY)               BASE CYL FOR VM
    1622                  1622 6155 LRAPGV DS    CL1                       SECTOR DISP IN VM
    1623 01               1623 6156        DC    XL1'01'                   SECTORS TO WRITE
    1624 1900             1625 6157        DC    AL2(LRASB1)               ADDR CORE OUTPUT AREA
                               6158 *
                          1626 6159 LRAPUT EQU   *                         ADDR DISK PARM LIST
    1626 02               1626 6160        DC    AL1(@DPUT)                WRITE CODE
    1627 09               1627 6161        DC    AL1(@DCBCY)               BASE CYL FOR TABLES
    1628 40               1628 6162        DC    AL1(@DCST1)               1ST PG OF STATEMENT TABLE
    1629 06               1629 6163        DC    XL1'06'                   SECTORS TO WRITE
    162A 1A00             162B 6164        DC    AL(@CADDR)(B$LDRP)        ADDR CORE OUTPUT AREA
                               6165 *
                               6167 ***********************************************************************
                               6168 * 5703-XM1   COPYRIGHT IBM CORP 1970                                  *
                               6169 *            REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083     *
                               6170 *                                                                     *
                               6171 ***********************************************************************
                               6172 *STATUS -                                                             *
                               6173 *   VERSION 1 MODIFICATION 0                                          *
                               6174 *                                                                     *
                               6175 *FUNCTION -                                                           *
                               6176 *   * LSORTA IS DESIGNED TO SORT THE FOUR BYTE ENTRIES IN THE         *
                               6177 *     BRANCH TABLE FOR VIRTUAL ADDRESS RESOLUTION ROUTINE LRADDR      *
                               6178 *      * SORT MODE 1 - SORTS 1 SECTOR OF ENTRIES BY THE PAGE NUMBER   *
                               6179 *        IN THE LAST TWO BYTES                                        *
                               6180 *      * SORT MODE 2 - SORTS 1 SECTOR OF ENTRIES BY THE VIRTUAL       *
                               6181 *        ADDRESS IN THE FIRST TWO BITES                               *
                               6182 *                                                                     *
                               6183 *ENTRY POINTS -                                                       *
                               6184 *   LSORTA HAS ONLY ONE ENTRY POINT                                   *
                               6185 *   CALLING SEQUENCE                                                  *
                               6186 *        B     LSORTA                                                 *
                               6187 *                                                                     *
                               6188 *INPUT -                                                              *
                               6189 *   * REGISTER PT1 (@BR), CONTAINS THE CORE ADDRESS OF THE FIRST      *
                               6190 *     BYTE IN THE BUFFER                                              *
                               6191 *   * RESISTER PT2 (@BR), CONTAINS THE CORE ADDRESS OF THE FIRST      *
                               6192 *     TWO BYTE ARGUMENT TO BE USED AS THE SORT CRITERION              *
                               6193 *   * LSOBOT - 2 BYTES, CONTAINS THE CORE ADDRESS OF THE NEXT TO      *
                               6194 *     THE LAST TWO BYTE ARGUMENT TO BE SORTED                         *
                               6195 *   * BRANCH TABLE BUFFER                                             *
                               6196 *                                                                     *
                               6197 *OUTPUT -                                                             *
                               6198 *   BRANCH ADDRESS TABLE BUFFER - WITH ENTRIES IN ASCENDING ORDER     *
                               6199 *                                                                     *
                               6200 *EXTERNAL REFERENCES -                                                *
                               6201 *   N/A                                                               *
                               6202 *                                                                     *
                               6203 *EXITS, NORMAL -                                                      *
                               6204 *   LSORTA HAS ONE NORMAL EXIT, TO THE FIRST INSTRUCTION FOLLOWING    *
                               6205 *   THE CALLING SEQUENCE.  THE REGISTERS ARE NOT RESTORED.            *
                               6206 *   THE RETURN ADDRESS IS IN THE ADDRESS RETURN REGISTER (@ARR)       *
                               6207 *                                                                     *
                               6208 *EXITS, ERROR -                                                       *
                               6209 *   N/A                                                               *
                               6210 *                                                                     *
                               6211 *TABLES/WORK AREA -                                                   *
                               6212 *   * THE CONSTANTS AND WORK AREAS RESIDE AT THE END OF EXECUTABLE    *
                               6213 *     CODE                                                            *
                               6214 *   * BRANCH ADDRESS TABLE BUFFER - SUPPLIED BY CALLING ROUTINE       *
                               6215 *                                                                     *
                               6216 *ATTRIBUTES -                                                         *
                               6217 *   LSORTA IS REUSABLE                                                *
                               6218 *                                                                     *
                               6219 *CHARACTER CODE DEPENDENCY -                                          *
                               6220 *   N/A                                                               *
                               6221 *                                                                     *
                               6222 *NOTES -                                                              *
                               6223 *   ERROR PROCEDURES                                                   *
                               6224 *      N/A                                                             *
                               6225 *                                                                      *
                               6226 *   REGISTER BASE                                                      *
                               6227 *      * REGISTERS PT1 AND PR2 (@BR, @XR) ARE USED IS INPUT            *
                               6228 *        PARAMETERS AND ARE USED DURING EXECUTION                      *
                               6229 *      * THE REGISTERS ARE NOT SAVED OR RESTORED                       *
                               6230 *                                                                      *
                               6231 *    SAVED/RESTORED AREAS                                              *
                               6232 *      N/A                                                             *
                               6233 *                                                                      *
                               6234 *    MODIRICATION CONSIDERATIONS                                       *
                               6235 *      N/A                                                             *
                               6236 *                                                                      *
                               6237 *    REQUIRED MODULES                                                  *
                               6238 *        @SYSEQ - COMMON SYSTEM EQUATES                                *
                               6239 *        LRADDR - LOADER ADDRESS RESOLUTION                            *
                               6240 *                                                                      *
                               6241 *    OTHER                                                             *
                               6242 *      N/A                                                             *
                               6243 ************************************************************************
                               6245 ***********************************************************************
                               6246 *                                                                     *
                               6247 *      EXECUTION LOADER SORT ROUTINE                                  *
                               6248 *                                                                     *
                               6249 ***********************************************************************
                               6250 *
                               6251 * SORT ROUTINE COMMON INITIALIZATION
                               6252 *
    162C 34 08 16CC            6253 LSORTA ST    LSO900+@OP1,@ARR           SAVE RETURN ADDR
    1630 3C 00 16D1            6254        MVI   LSOBSW,LSOB00              CLEAR BOTTON SWITCH
    1634 34 02 16D5            6255        ST    LSOTOP,@XR                 SAVE STARTING ADDR
    1638 36 02 16D0            6256        A     LSODEC,@XR                 DECR FOR FIRST PASS
    163C 36 01 16D0            6257        A     LSODEC,@BR                 DECR FOR FIRST PASS
                               6258 *
                               6259 * SIFTING DOWN ROUTINE
                               6260 *
    1640 36 02 16CE            6261 LSO100 A     LSOINC,@XR                 INCR TO NEXT ENTRY
    1644 36 01 16CE            6262        A     LSOINC,@BR                 INCR TO NEXT ENTRY
    1648 34 02 16D7            6263        ST    LSOBEY,@XR                 STORE PRESENT ENTRY
    164C 0D 01 16D7 16D3       6264        CLC   LSOBEY(LSOBCT),LSOBOT      AT LAST ENTRY
    1652 F2 02 68              6265        JNL   LSO800                     IF YES, CHECK FOR CHANGE
    1655 AD 01 07 03           6266        CLC   LSO2ND(LSOBCT,@XR),LSO1ST(,@XR)  ARE ENTRIES IN ORDER
    1659 C0 02 1640            6267        BNL   LSO100                     YES, GO TO NEXT PAIR ENTRIES
                               6268 *
                               6269 * SWITCHING ENTRIES
                               6270 *
    165D 34 01 169E            6271        ST    LSO500+@OP1,@BR            SAVE PRESENT ADDR 2ND PT
    1661 34 02 169A            6272        ST    LSO400+@OP1,@XR            SAVE PRESENT ADDR 1ST PT
    1665 34 02 16D9            6273 LSO200 ST    LSOTEY,@XR                 SAVE PRESENT ADDR
    1669 0D 01 16D9 16D5       6274        CLC   LSOTEY(LSOBCT),LSOTOP      AT FIRST ENTRY
    166F F2 81 31              6275        JE    LSO600                     YES, DO TOP RTN
    1672 1C 03 16DD 03         6276 LSO210 MVC   LSOHLD(LSOECT),LSO1ST(,@BR)  TEMP SAVE OF ADDR 1
    1677 5C 03 03 07           6277        MVC   LSO1ST(LSOECT,@BR),LSO2ND(,@BR)  CHANGE ADDR 2
    167B 4C 03 07 16DD         6278        MVC   LSO2ND(LSOECT,@BR),LSOHLD  CHANGE ADDR 1
                               6279 *
                               6280 * BUBBLING UP ROUTINE
                               6281 *
    1680 36 02 16D0            6282 LSO250 A     LSODEC,@XR                DECR 1ST PT 1 ENTRY UP
    1684 36 01 16D0            6283        A     LSODEC,@BR                DECR 2ND PT 1 ENTRY UP
    1688 AD 01 07 03           6284        CLC   LSO2ND(LSOBCT,@XR),LSO1ST(,@XR)  ARE ENTRIES IN ORDER
    168C C0 82 1665            6285        BL    LSO200                    NO, SWITCH ENTRIES
    1690 3D 01 16D1            6286 LSO300 CLI   LSOBSW,LSOB01             IS END SWITCH ON ?
    1694 F2 81 32              6287        JE    LSO900                    YES, END RTN
    1697 C2 02 0000            6288 LSO400 LA    *-*,@XR                   RESTORE SAVED ADDR 1 PT
    169B C2 01 0000            6289 LSO500 LA    *-*,@BR                   RESTORE SAVED ADDR 2 PT
    169F C0 87 1640            6290        B     LSO100                    CONTINUE SIFTING DOWN
                               6291 *
                               6292 * AT FIRST ENTRY ROUTINE
                               6293 *
    16A3 AD 01 07 03           6294 LSO600 CLC   LSO2ND(LSOBCT,@XR),LSO1ST(,@XR)  ARE ENTRIES IN ORDER ?
    16A7 C0 02 1690            6295        BNL   LSO300                    YES, CHECK IF DONE
    16AB 1C 03 16DD 03         6296 LSO650 MVC   LSOHLD(LSOECT),LSO1ST(,@BR)  TEMP SAVE OF ADDR 1
    16B0 5C 03 03 07           6297        MVC   LSO1ST(LSOECT,@BR),LSO2ND(,@BR) CHANGE ADDR 2
    16B4 4C 03 07 16DD         6298        MVC   LSO2ND(LSOECT,@BR),LSOHLD  CHANGE ADDR 1
    16B9 C0 87 1690            6299        B     LSO300                    CHECK IF DONE
                               6300 *
                               6301 * AT LAST ENTRY ROUTINE
                               6302 *
    16BD 3C 01 16D1            6303 LSO800 MVI   LSOBSW,LSOB01             TURN LAST ENTRY SW ON
    16C1 AD 01 07 03           6304        CLC   LSO2ND(LSOBCT,@XR),LSO1ST(,@XR)  ARE ENTRIES IN ORDER ?
    16C5 C0 82 1665            6305        BL    LSO200                    NO, REVERSE THE ENTRIES
    16C9 C0 87 0000            6306 LSO900 B     *-*                       RETURN TO CALLING ROUTINE

                               6308 ***********************************************************************
                               6309 * LSORTA CONSTANTS, WORK AREAS AND EQUATES
                               6310 ***********************************************************************
                               6311 *
                               6312 * LSORTA EQUATES REFERENCING CONSTANTS
                               6313 *
                          0000 6314 LSOB00 EQU   0                         BINARY ZERO
                          0001 6315 LSOB01 EQU   1                         BINARY ONE
                          0002 6316 LSOBCT EQU   2                         BYTES IN AN ARGUMENT
                          0003 6317 LSO1ST EQU   3                         DISP PG NO. 1ST ENTRY
                          0004 6318 LSOECT EQU   4                         BYTES IN AN ENTRY
                          0007 6319 LSO2ND EQU   7                         DISA PG NO. 2ND ENTRY
                               6320 *
                               6321 * LSORTA CONSTANTS
                               6322 *
    16CD 0004             16CE 6323 LSOINC DC    XL2'0004'                 INCR BETWEEN ENTRIES
    16CF FFFC             16D0 6324 LSODEC DC    XL2'FFFC'                 DECR BETWEEN ENTRIES
                               6325 *
                               6326 * LSORTA WORK AREAS
                               6327 *
    16D1                  16D1 6328 LSOBSW DS    CL1                       BOTTOM SWITCH
    16D2                  16D3 6329 LSOBOT DS    CL2                       BOTTOM ADDR OF SORT SAVE AREA
    16D4                  16D5 6330 LSOTOP DS    CL2                       TOP ADDR OF SORT SAVE AREA
    16D6                  16D7 6331 LSOBEY DS    CL2                       LAST ENTRY COMPARE AREA
    16D8                  16D9 6332 LSOTEY DS    CL2                       FIRST ENTRY COMPARE AREA
    16DA                  16DD 6333 LSOHLD DS    CL4                       TEMP ENTRY SAVE AREA
                               6334 *
                               6335 *      $C4BD
                               6337+*                                                                    *
                               6338+*                  INITIALIZATION                                    *
                               6339+*                                                                    *
                          16DE 6340+C4BIN2 EQU   *                         ENTRY POINT
                          16DE 6341+       USING C4BIN2,@BR                BASE VALUE
                               6342+*
    16DE 34 01 1740            6343+       ST    C4B800+@OP1,@BR           SAVE CALLERS BASE REGISTER
    16E2 C2 01 16DE            6344+       LA    C4BIN2,@BR                LOAD BASE VALUE
                               6345+*
    16E6 74 08 66              6346+       ST    C4B850+@OP1(,@BR),@ARR    SAVE RETURN ADDRESS
                               6347+*
    16E9 74 02 6E              6348+       ST    C4BSAV(,@BR),@XR          SAVE VALUE OF POINTER
    16EC 3C 0C 03CD            6349+       MVI   $CAERR,@@E122             SET ERROR CODE IN CASE
    16F0 5C 01 6A 6B           6350+       MVC   C4BVAL(C4BLVL,@BR),C4BINI(,@BR) INIT VALUE TO ZERO
    16F4 3C 04 174D            6351+C4B100 MVI   C4B900,4                  INITLZ CHAR. COUNT
                               6352+*
                               6353+***    DETERMINE IF CHAR NUMERIC AND DECR CHAR COUNT
                               6354+*
    16F8 F2 80 32              6355+C4B200 JC    C4B600,@NOP               SET TO UCB IF IMBEDDED BLANKS
                               6356+*                                      * ALLOWED
    16FB BD F0 00              6357+C4B300 CLI   0(,@XR),C4BLOW            THIS CHAR NUMERIC ?
    16FE F2 82 35              6358+       JL    C4B700                    NO, GOTO RETURN
                               6359+*
    1701 5F 00 6F 4E           6360+       SLC   C4B900(1,@BR),C4B590+@D1(,@BR) DECR CHAR COUNT
    1705 F2 82 35              6361+       JL    C4B800                    BR TO ERROR EXIT IF TOO MANY
                               6362+*
                               6363+***    MULTIPLY PREVIOUS VALUE BY TEN
                               6364+*
    1708 5E 01 6A 6A           6365+       ALC   C4BVAL(C4BLVL,@BR),C4BVAL(,@BR) DOUBLE PREVIOUS VALUE
    170C 5C 01 68 6A           6366+       MVC   C4BWRK(C4BLVL,@BR),C4BVAL(,@BR) SAVE DOUBLE VALUE
    1710 5E 01 6A 6A           6367+       ALC   C4BVAL(C4BLVL,@BR),C4BVAL(,@BR) QUADRUPLE PREVIOUS VALUE
    1714 5E 01 6A 6A           6368+       ALC   C4BVAL(C4BLVL,@BR),C4BVAL(,@BR) OCTUPLE PREVIOUS VALUE
    1718 5E 01 6A 68           6369+       ALC   C4BVAL(C4BLVL,@BR),C4BWRK(,@BR) ADD IN SAVED DOUBLE
                               6370+*
                               6371+***    ADD IN VALUE OF THIS CHAR AND INCR POINTER
                               6372+*
    171C 68 03 6C 00           6373+       MNN   C4BCHR(,@BR),0(,@XR)      FETCH NEMERIC VALUE OF NEW CHAR
    1720 5E 01 6A 6C           6374+       ALC   C4BVAL(C4BLVL,@BR),C4BCHR(,@BR) INCR VALU BY THIS CHAR
                               6375+*
    1724 E2 02 01              6376+       LA    @B1(,@XR),@XR             INCR POINTER TO NEXT CHAR
    1727 D0 87 1A              6377+       B     C4B200(,@BR)              GOTO DO IT AGAIN
                               6378+*                                                                    *
                               6379+*                  ROUTINE TO SCAN BLANKS                            *
                               6380+*                                                                    *
    172A E2 02 01              6381+C4B590 LA    @B1(,@XR),@XR             INCR POINTER TO NEXT CHAR
    172D BD 40 00              6382+C4B600 CLI   0(,@XR),@BLANK            IS THIS CHAR A BLANK ?
    1730 D0 01 1D              6383+       BNE   C4B300(,@BR)              RETURN IF NOT
    1733 D0 87 4C              6384+       B     C4B590(,@BR)              GET NEXT CHAR IF YES
```

```
                               6386+*
                               6387+***    ENDING ROUTINE
                               6388+*
    1736 74 02 68              6389+C4B700 ST    C4BLEN(,@BR),@XR          PLACE VALUE OF POINTER
    1739 5F 01 68 6E           6390+       SLC   C4BLEN(2,@BR),C4BSAV(,@BR) SUBRACT ENTERING VALUE
                               6391+*
    173D C2 01 0000            6392+C4B800 LA    *-*,@BR                   RESTORE CALLERS BR
                               6393+*
    1741 C0 87 0000            6394+C4B850 B     *-*                       RETURN TO CALLING ROUTINE
                               6395+*                                                                    *
                               6396+*                  WORK AREA AND CONSTANT                            *
                               6397+*                                                                    *
    1745                  1746 6398+C4BWRK DS    CL2                       SAVE AREA FOR DOUBLED VALUE
                               6399+*
                          1747 6400+C4BYT1 EQU   *                         FIRST BYTE OF BINARY VALUE
    1747                  1748 6401+C4BVAL DS    CL2                       SAVE AREA FOR BINARY VALUE
                               6402+*
    1749 00               1749 6403+C4BINI DC    XL1'00'                   INITIALIZE WA TO ZERO
                               6404+*
    174A                  174A 6405+C4BCHR DS    CL1                       SAVE AREA FOR EACH NEW CHAR
    174A                       6406+       ORG   *-1                       INITIALIZE
    174A 00               174A 6407+       DC    XL1'00'                   * TO ZERO
                               6408+*
    174B                  174C 6409+C4BSAV DS    CL2                       SAVE AREA FOR XR
                               6410+*
    174D                  174D 6411+C4B900 DS    CL1                       SAVE AREA FOR CHAR COUNTER
                               6412+*                                                                    *
                               6413+*                  EQUATES FOR C4BIN2                                *
                               6414+*                                                                    *
                          1746 6415+C4BLEN EQU   C4BWRK                    ON RETURN WILL CONTAIN COUNT
                               6416+*                                      * @XR INCREMENTED BY
                          0004 6417+C4BCHC EQU   4                         NUMBER OF CHAR TO CONVERT
                               6418+*
                          00F0 6419+C4BLOW EQU   C'0'                      LOWEST NUMERIC CHARACTER
                               6420+*
                          0002 6421+C4BLVL EQU   C4BVAL-C4BWRK             LENGTH OF BINARY VALUE
                               6422+*
                          16F9 6423+C4BLNK EQU   C4B200+@Q                 LOCATION OF IMBEDDED BLANK IND
                               6424+*
                          0087 6425+C4BSPC EQU   @UCB                      MOVED TO C4BLNK TO ALLOW BLANKS
                               6426+*
                          16F5 6427+C4BNMC EQU   C4B100+@Q                 LOCATION OF CONVERSION COUNT
                               6428+*
                          0080 6429+C4BNOP EQU   @NOP                      CHANGED IF IMBEDDED BLANK OK
                          174E 6430+C4END  EQU   *                         DEFINE END OF CODE
                               6431+***                END OF C4BIN2                                   ***
                               6432 *
                               6433 *      $DL2P
                               6435+***********************************************************************
                               6436+*  5703-XM1  COPYRIGHT IBM CORP 1970                                  *
                               6437+*            REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE. 120-2083      *
                               6438+*                                                                     *
                               6439+***********************************************************************
                               6440+*STATUS -                                                             *
                               6441+*   VERSION 1 MODIFICATION 0                                          *
                               6442+*                                                                     *
                               6443+*FUNCTION                                                             *
                               6444+*   * DL2ICS CONVERTS A RELATIVE DISK ADDRESS TO A PHYSICAL DISK      *
                               6445+*     ADDRESS AND COMBINES IT WITH A BASE ADDRESS PLACED IN DL2RAD    *
                               6446+*     BY THE CALLER.                                                  *
                               6447+*   * THE RELATIVE DISK ADDRESS IS A TWO BYTE CYLINDER SECTOR COUNT   *
                               6448+*     IN THE CALLERS DISK PARAMETER LIST (DPL).                       *
                               6449+*   * THE COUNT IS A CYLINDER SECTOR DISPLACEMENT FROM THE BASE       *
                               6450+*     ADDRESS PLACED IN DL2RAD                                        *
                               6451+*   * DL2ICS IS USED TO PROCESS DATA ON THE FIXED OR REMOVABLE DISK   *
                               6452+*     ON EITHER DRIVE AND PROVIDES THE INTERFACE TO $DISKN.           *
                               6453+*   * THE PHYSICAL DISK ADDRESS IS PLACED IN A COPY OF THE USERS DPL  *
                               6454+*     IN DL2ICS AND A CALL IS MADE TO $DISKN TO PERFORM THE REQUESTED *
                               6455+*     OPERATION.                                                      *
                               6456+*                                                                     *
                               6457+*ENTRY POINTS                                                         *
                               6458+*   * THE ENTRY IS DL2ICS. THE BASE REGISTER IS SAVED AND RESTORED    *
                               6459+*     ON RETURN. THE INDEX REGISTER IS NOT USED.                      *
                               6460+*   * THE FORMAT OF THE CALLING SEQUENCE IS AS FOLLOWS:               *
                               6461+*        B   DL2ICS                                                   *
                               6462+*        DC  AL2(PARMLT)                                              *
                               6463+*     WHERE PARMLT IS THE ADDR OF THE PARAMETER LIST TO BE PROCESSED. *
                               6464+*                                                                     *
                               6465+*INPUT                                                                *
                               6466+*   * THE INPUT IS A TWO BYTE BASE DISK ADDRESS PLACED IN             *
                               6467+*     DL2RAD AND A SIX BYTE DPL. THE SAME FORMAT AS THE DPL FOR       *
                               6468+*     $DISKN EXCEPT FOR THE DISK ADDRESS WHICH IS A RELATIVE CYLINDER *
                               6469+*     AND SECTOR DISPLACEMENT FROM THE BASE ADDRESS IN DL2RAD.        *
                               6470+*                                                                     *
                               6471+*OUTPUT                                                               *
                               6472+*     NONE.                                                           *
                               6473+*                                                                     *
                               6474+*EXTERNAL REFERENCES                                                  *
                               6475+*     $DISKN - ENTRY TO PHYSICAL DISK ROUTINE IS THE SYSTEM NUCLEUS.  *
                               6476+*                                                                     *
                               6477+*EXITS, NORMAL                                                        *
                               6478+*     NORMAL - EXIT IS TO THE FIRST INSTRUCTION FOLLOWING THE POINTER *
                               6479+*     TO THE DPL. THE BASE REGISTER IS RESTORED. THE RETURN ADDRESS   *
                               6480+*     IS THE ADDRESS RECALL REGISTER (ARR) +2.                        *
                               6481+*                                                                     *
                               6482+*EXITS, ERROR                                                         *
                               6483+*     NONE                                                            *
                               6484+*                                                                     *
                               6485+*TABLES/WORK AREAS                                                    *
                               6486+*   * THE CONSTANTS AND WORK AREAS RESIDE AT THE END OF THE EXECUTABLE*
                               6487+*     CODE AND ARE REFERENCED BY A DISPLACEMENT RELATIVE TO THE VALUE *
                               6488+*     IN INDEX REGISTER 1 (@BR).                                      *
                               6489+*   * DL2SEC AND DL2SAD ARE EQUATED TO OPERAND LOCATIONS IN THE       *
                               6490+*     EXECUTABLE CODE TO ELIMINATE EXCESS WORKING STORAGE.            *
                               6491+*                                                                     *
                               6492+*ATTRIBUTES                                                           *
                               6493+*   * DL2ICS IS REUSABLE                                              *
                               6494+*                                                                     *
                               6495+*CHARACTER CODE DEPENDENCY                                            *
                               6496+*     THE OPERATION OF THIS MODULE DOES NOT DEPEND UPON A PARTICULAR  *
                               6497+*     INTERNAL REPRESENTATION OF THE EXTERNAL CHARACTER SET.          *
                               6498+*                                                                     *
                               6499+*NOTES                                                                *
                               6500+*   ERROR PROCEDURES                                                  *
                               6501+*      NONE                                                           *
                               6502+*                                                                     *
                               6503+*   REGISTER USAGE                                                    *
                               6504+*      INDEX REGISTER 1 (@BR) IS SAVED AND RESTORED. THIS REGISTER IS *
                               6505+*      USED DURING EXECUTION. REGISTER 2 (@BR) IS NOT USED.           *
                               6506+*                                                                     *
                               6507+*   SAVED/RESTORED AREAS                                              *
                               6508+*      NONE                                                           *
                               6509+*                                                                     *
                               6510+*   MODIFICATION CONSIDERATIONS                                       *
                               6511+*      NONE                                                           *
                               6512+*                                                                     *
                               6513+*   REQUIRED MODULES                                                  *
                               6514+*     @SYSEQ - COMMON SYSTEM EQUATES.                                 *
                               6515+*     @FXDEQ - SYSTEM NUCLEUS ADDRESSES AND INDICATORS VALUES EQUATES *
                               6516+*                                                                     *
                               6517+*   OTHER                                                             *
                               6518+*      DL2ICS MAY BE USED TO CONVERT THE DISK ADDRESS ONLY AND NOT TO *
                               6519+*      CALL $DISKN IF THE USER MOVES A UCB CODE TO DL2SWH.            *
                               6520+*      THIS OPTION IS NOT STANDARD USAGE.                             *
                               6521+***********************************************************************
                          1752 6522+       USING DL2000,@BR                ESTABLISH ADDRESSABILITY
                               6523+*
                          0001 6524+DL2E01 EQU   X'01'                     FIELD LENGTH OF 1
                          0002 6525+DL2E02 EQU   X'02'                     FIELD LENGTH OF 2
                          0018 6526+DL2E18 EQU   X'18'                     HEX TRACK SECTOR COUNT
                          0060 6527+DL2E60 EQU   X'60'                     PHYSICAL SECTOR COUNT
                          0083 6528+DL2TSD EQU   X'83'                     MASK OFF TRACK SPINDLE DISK
                          007C 6529+DL2E7C EQU   X'7C'                     MASK OUT SECTOR COUNT
                          174E 6530+DL2ICS EQU   *                         ENTRY POINT
    174E 34 01 17CF            6531+       ST    DL2900+@OP1,@BR           SAVE OLD BASE
                          1752 6532+DL2000 EQU   *                         START PROCESSING
    1752 C2 01 1752            6533+       LA    DL2000,@BR                SET BASE ADORESS
    1756 76 08 8A              6534+       A     DL2C01(,@BR),@ARR         BUMP TO RIGHT BYTE OF ADDR
    1759 74 08 14              6535+       ST    DL2001+@DOP2(,@BR),@ARR   ADDR OF PARAM
    175C 76 08 8A              6536+       A     DL2C01(,@BR),@ARR         BUMP TO RETURN ADDR
    175F 74 08 81              6537+       ST    DL2910+@OP1(,@BR),@ARR    SAVE RETURN ADDR
                               6538+*
    1762 4C 01 1D 0000         6539+DL2001 MVC   DL2002+@DOP2(@DADDR,@BR),*-* SETUP ADDR OF DPL
    1767 5E 01 1D 8C           6540+       ALC   DL2002+@DOP2(@CADDR,@BR),DL2C05(,@BR) DUMP TO RIGHT END
    176B 4C 05 92 0000         6541+DL2002 MVC   DL2DPL(@DPLNG,@BR),*-*    MOVE USER DPL TO WORK AREA
    1770 5F 00 8F 86           6542+DL2005 SLC   DL2LST+@DSAD(DL2E01,@BR),DL2C48(,@BR) ADJUST SCTR/CYL
    1774 F2 82 07              6543+       JM    DL2006                    GO TO RESTORE TO CONTINUE
    1777 5E 00 8E 8A           6544+       ALC   DL2LST+@DCYL(DL2E01,@BR),DL2C01(,@BR) BUMP CYLINDER COUNT
    177B D0 87 1E              6545+       B     DL2005(,@BR)              BACK FOR NEXT CYLINDER
    177E 5E 00 8F 86           6546+DL2006 ALC   DL2LST+@DSAD(DL2E01,@BR),DL2C48(,@BR) RESTORE POSITIVE
```

```
                               6547+*
                               6548+*      GET THE LOGICAL SECTOR FROM THE DPL. THE NUMBER IS LEFT ADJUSTED
                               6549+*      TO COMAE IT MTN THE POINTER ESTABLISHED PRIOR TO AN ENTRY.
    1782 5C 00 1D 8F           6550+       MVC   DL2SEC(DL2E01,@BR),DL2LST+@DSAD(,@BR) GET SECTOR NUMBER
    1786 7C 00 8F              6551+       MVI   DL2LST+@DSAD(,@BR),@ZERO  CLEAR SECTOR BYTE
                               6552+*
                               6553+*      MOVE THE RELATIVE START TO THE DFL
                               6554+*
    1789 5E 01 8F 94           6555+       ALC   DL2LST+@DSAD(DL2E02,@BR),DL2RAD(,@BR) DL2RAD TO DPL
    178D 7D 18 1D              6556+       CLI   DL2SEC(,@BR),DL2E18       IS COUNT OVER A TRACK
    1790 F2 82 08              6557+       JL    DL2008                    NO GO CHANGE A PHYSICAL ADOR
    1793 5E 01 8F 85           6558+       ALC   DL2LST+@DSAD(DL2E02,@BR),DL2K80(,@BR) BUMP TRACK VALUE
    1797 5F 00 1D 88           6559+       SLC   DL2SEC(1,@BR),DL2K18(,@BR) DECR BY TRACK VALUE
    179B 5E 00 1D 1D           6560+DL2008 ALC   DL2SEC(1,@BR),DL2SEC(,@BR) SHIFT LEFT 1
    179F 5E 00 1D 1D           6561+       ALC   DL2SEC(1,@BR),DL2SEC(,@BR) SHIFT LEFT
    17A3 5C 00 14 8F           6562+       MVC   DL2SAD(DL2E01,@BR),DL2LST+@DSAD(,@BR) GET SECTOR ADDRESS
                               6563+*
                               6564+*      ZERO OUT THE SECTOR COUNT AND LEAVE THE DISK. SPINDLE AND
                               6565+*      TRACK BITS AS IS TO BE RE INSERTED AFTER THE SECTOR HAS BEEN
                               6566+*      LOCATES.
                               6567+*
    17A7 7B 7C 8F              6568+       SBF   DL2LST+@DSAD(,@BR),DL2E7C TURN OFF
    17AA 7B 83 14              6569+       SBF   DL2SAD(,@BR),DL2TSD       OFF TRACK SPINDLE DISK
    17AD 5E 00 14 1D           6570+       ALC   DL2SAD(DL2E01,@BR),DL2SEC(,@BR) COMBINE SECTOR COUNTS
    17B1 7D 60 14              6571+DL2010 CLI   DL2SAD(,@BR),DL2E60       TEST IF TRACK CROSSED
    17B4 F2 82 08              6572+       JL    DL2100
                               6573+*
                               6574+*      INCREMENT TRACK BIT. OVERFLOW INTO THE CYLINDER COUNT.
                               6575+*
    17B7 5E 01 8F 85           6576+       ALC   DL2LST+@DSAD(DL2E02,@BR),DL2K80(,@BR)
    17BB 5F 00 14 83           6577+       SLC   DL2SAD(1,@BR),DL2K60(,@BR) DECR BY TRACK VALUE
                               6578+*
    17BF 5E 00 8F 14           6579+DL2100 ALC   DL2LST+@DSAD(1,@BR),DL2SAD(,@BR) INSERT SECTOR COUNT
                               6580+*
    17C3 F2 80 06              6581+DL2110 JC    DL2900,@NOP               CONVERSION SWITCH
                          17C4 6582+DL2SWH EQU   DL2110+@Q                 ADDR OF Q CODE FOR SWITCH
    17C6 C0 87 0025            6583+       B     $DISKN                    GO PROCESS I/O
    17CA 17DF             17CB 6584+       DC    AL2(DL2LST)               ADDRESS OF DPL
    17CC C2 01 0000            6585+DL2900 LA    *-*,@BR                   RESTORE CALLERS BASE
    17D0 C0 87 0000            6586+DL2910 B     *-*
                               6587+**********************************************************************
                               6588+*      CONSTANTS
                               6589+**********************************************************************
    17D4 0060             17D5 6590+DL2K60 DC    XL2'0060'                 SECTOR COUNT OF 24 LEFT ADJUSTD
    17D6 0080             17D7 6591+DL2K80 DC    XL2'0080'                 BIT FOR INCREMENTING TRACK
    17D8 30               17D8 6592+DL2C48 DC    IL1'48'                   CYLINDER VALUE FOR 1 DISK
    17D9 0018             17DA 6593+DL2K18 DC    XL2'18'                   HEX SECTORS PER TRACK
    17DB 0001             17DC 6594+DL2C01 DC    IL2'1'                    CONSTANT FOR REGISTER MODE
    17DD 0005             17DE 6595+DL2C05 DC    IL2'5'                    DISP TO RIGHT END OF DPL
                               6596+**********************************************************************
                               6597+*      WORK AREA
                               6598+**********************************************************************
                          17DF 6599+DL2LST EQU   *                         LIST HIGH END
    17DF                  17E4 6600+DL2DPL DS    CL(@DPLNG)                WORKING DPL
                          17E1 6601+DL2PHY EQU   DL2LST+@DSAD              POINTER TO PHYSICAL DADDR
                          1766 6602+DL2SAD EQU   DL2001+@DOP2              SAVE SECTOR BYTE FROM DPI
                          176F 6603+DL2SEC EQU   DL2002+@DOP2              WORKING SECTOR ADDRESS FIELD
    17E5                  17E6 6604+DL2RAD DS    CL(@DADDR)                USER RELATIVE STARTING ADDR.
                          17E7 6605+DL2END EQU   *                         END OF DL2ICS
                               6606+***                       END OF DL2ICS                            ***
                               6607 *
                               6608 *      $DL4P
                               6610+***********************************************************************
                               6611+* 5703-XM1  COPYRIGHT IBM CORP. 1970                                  *
                               6612+*           REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083       *
                               6613+*                                                                     *
                               6614+***********************************************************************
                               6615+*STATUS                                                               *
                               6616+*   VERSION 1 MODIFICATION 0                                          *
                               6617+*                                                                     *
                               6618+*FUNCTION                                                             *
                               6619+*   * DL4ICS WILL CONVERT A RELATIVE DISK ADDRESS TO A PHYSICAL       *
                               6620+*     DISK ADDRESS AND CALL $DISKN TO PERFORM THE SPECIFIED FUNCTION  *
                               6621+*   * THE DISK ADDRESS IS A ONE BYTE CYLINDER ADDRESS AND A ONE BYTE  *
                               6622+*     SECTOR DISPLACEMENT RELATIVE TO SECTOR 0 ON A CYLINDER          *
                               6623+*     BOUNDARY                                                        *
                               6624+*   * WHEN MORE THAN 1 SECTOR IS PROCESSED, DL4ICS WILL MAKE MULTIPLE *
                               6625+*     CALLS TO $DISKN TO CROSS CYLINDER BOUNDARIES IF REQUIRED.       *
                               6626+*   * IF 1 SECTOR ONLY IS TO BE PROCESSED, THE USER MAY OVERLAY THE   *
                               6627+*     UNUSED CODE BY ORGING HIS NEXT MODULE AT DL4SPT                 *
                               6628+*                                                                     *
                               6629+*ENTRY POINTS                                                         *
                               6630+*     DL4ICS - ENTRY TO PROCESS A 4 SURFACE FILE. THE CALLING         *
                               6631+*              SEQUENCE IS AS FOLLOWS                                 *
                               6632+*                  DSKL4  DPL                                         *
                               6633+*              WHERE DPL IS THE LABEL OF A SIX BYTE DISK PARAMETER    *
                               6634+*              LIST AS DESCRIBED FOR $DISKN EXCEPT FOR THE SECTOR     *
                               6635+*              ADDRESS BYTE.                                          *
                               6636+*                                                                     *
                               6637+*INPUT                                                                *
                               6638+*   * INPUT TO DL4ICS IS THE ADDRESS OF THE DPL TO BE PROCESSED.      *
                               6639+*                                                                     *
                               6640+*OUTPUT                                                               *
                               6641+*   * N/A                                                             *
                               6642+*                                                                     *
                               6643+*EXTERNAL REFENECES                                                   *
                               6644+*     $DISKN - ENTRY TO SYSTEM DISK ROUTINE                           *
                               6645+*                                                                     *
                               6646+*EXITS, NORMAL                                                        *
                               6647+*   * NORMAL RETURN IS TO THE 1ST INSTRUCTION FOLLOWING THE TWO BYTE  *
                               6648+*     ADDRESS POINTING TO THE DPL.                                    *
                               6649+*                                                                     *
                               6650+*EXITS, ERROR                                                         *
                               6651+*   * N/A                                                             *
                               6652+*                                                                     *
                               6653+*TABLES/WORK AREAS                                                    *
                               6654+*   * N/A                                                             *
                               6655+*                                                                     *
                               6656+*ATTRIBUTES                                                           *
                               6657+*   * RELOCATABLE                                                     *
                               6658+*   * REUSABLE                                                        *
                               6659+*                                                                     *
                               6660+*CHARACTER CODE DEPENDENCY                                            *
                               6661+*   * THE OPERATION OF THIS MODULE DOES NOT DEPEND UPON A PARTICULAR  *
                               6662+*     INTERNAL REPRESENTATION OF THE EXTERNAL CHARACTER SET.          *
                               6663+*                                                                     *
                               6664+*NOTES                                                                *
                               6665+*   ERROR PROCEDURES                                                  *
                               6666+*      N/A                                                            *
                               6667+*                                                                     *
                               6668+*   REGISTER USAGE                                                    *
                               6669+*      @BR IS SAVED AND RESTORED ON EXIT, @XR IS NOT USED.  @ARR IS   *
                               6670+*      USED TO PROVIDE THE ADDRESS OF THE PARAMETER.  THE @ARR IS     *
                               6671+*      INCREMENTED BT TWO AND SAVED AS THE RETURN ADDRESS.            *
                               6672+*                                                                     *
                               6673+*   SAVED/RESTORED AREAS                                              *
                               6674+*      N/A                                                            *
                               6675+*                                                                     *
                               6676+*   MODIFICATION CONSIDERATIONS                                       *
                               6677+*      N/A                                                            *
                               6678+*                                                                     *
                               6679+*   REQUIRED MODULES                                                  *
                               6680+*      @SYSEQ - SYSTEM SOFTWARE EQUATES                               *
                               6681+*      @FXDEQ - SYSTEM NUCLEUS EQUATES                                *
                               6682+*                                                                     *
                               6683+*   OTHER                                                             *
                               6684+*      NONE                                                           *
                               6685+***********************************************************************
```

```
                          17E7 6687+DL4ICS EQU   *                         ENTRY TO DL4ICS
                          17EB 6688+       USING DL4010,@BR                ESTABLISH BASE REGISTER USAGE
    17E7 34 01 1857            6689+       ST    DL4900+@OP1,@BR           SAVE BASE REGISTER FOR EXIT
                          17EB 6690+DL4010 EQU   *                         BASE ADDRESSABILITY
    17EB C2 01 17EB            6691+       LA    DL4010,@BR                ESTABLISH BASE
    17EF 76 08 78              6692+       A     DL4C01(,@BR),@ARR         BUMP TO HIGH END OF ADDR
    17F2 74 08 14              6693+       ST    DL4020+@DOP2(,@BR),@ARR   SET UP MOVE INSTRUCTION
    17F5 76 08 78              6694+       A     DL4C01(,@BR),@ARR         BUMP TO RETURN ADDR
    17F8 74 08 70              6695+       ST    DL4920+@OP1(,@BR),@ARR    SAVE RETURN ADDR
                               6696+*
    17FB 4C 01 1D 0000         6697+DL4020 MVC   DL4030+@DOP2(@DADDR,@BR),*-* MOVE DPL ADDR INTO MOVE
    1800 5E 01 1D 7A           6698+       ALC   DL4030+@DOP2(@CADDR,@BR),DL4C05(,@BR) BUMP TO RIGHT END
    1804 4C 05 76 0000         6699+DL4030 MVC   DL4DPL(@DPLNG,@BR),*-*    MOVE USER DPL TO WORK AREA
                               6700+*
    1809 7C 00 5E              6701+DL4035 MVI   DL4100+@Q(,@BR),@ZERO     CLEAR TRACK, DISK SET INST
    180C 7C 80 67              6702+       MVI   DL4200+@Q(,@BR),@NOP      TURN OFF TWICE INDICATOR
                               6703+*
    180F 7D 60 73              6704+DL4040 CLI   DL4SCD(,@BR),DL4E96       TEST IF DISPLACEMENT OVER 95 ?
    1812 F2 82 0B              6705+       JL    DL4050                    JUMP IF NOT OVER 95
    1815 5E 00 72 78           6706+       ALC   DL4CYL(1,@BR),DL4C01(,@BR) INCREMENT CYLINDER COUNT
    1819 5F 00 73 25           6707+       SLC   DL4SCD(1,@BR),DL4C96(,@BR) DECREMENT DISP BY 96
    181D D0 87 24              6708+       B     DL4040(,@BR)              GO BACK CHECK FOR NEXT CYLINDER
                               6709+*
    1820 7D 30 73              6710+DL4050 CLI   DL4SCD(,@BR),DL4E48       TEST IF DISP ON NEXT DISK ?
    1823 F2 82 07              6711+       JL    DL4060                    JUMP IF NOT OVER 48
    1826 7A 01 5E              6712+       SBN   DL4100+@Q(,@BR),DL4EFD    TURN ON BIT FOR FIXED DISK
    1829 5F 00 73 36           6713+       SLC   DL4SCD(1,@BR),DL4C48(,@BR) DECREMENT DISP 1 DISK
    182D 7D 01 74              6714+DL4060 CLI   DL4SCT(,@BR),DL4E01       IS SECTOR COUNT GREATER THEN 1 ?
    1830 F2 84 33              6715+       JH    DL4SPT                    GO TO SPLIT CALL
    1833 7D 18 73              6716+DL4070 CLI   DL4SCD(,@BR),DL4E24       DISPLACEMENT OVER 23 ?
    1836 F2 82 07              6717+       JL    DL4080                    JUMP NOT OVER 24
    1839 7A 80 5E              6718+       SBN   DL4100+@Q(,@BR),DL4ETB    SET TRACK BIT ON
    183C 5F 00 73 49           6719+       SLC   DL4SCD(1,@BR),DL4C24(,@BR) DECR DISP TO NEXT TRACK
    1840 5E 00 73 73           6720+DL4080 ALC   DL4SCD(1,@BR),DL4SCD(,@BR) SHIFT LEFT 1 PLACE
    1844 5E 00 73 73           6721+       ALC   DL4SCD(1,@BR),DL4SCD(,@BR) SHIFT LEFT 1 PLACE
    1848 7A 00 73              6722+DL4100 SBN   DL4SCD(,@BR),*-*          SET TRACK, DISK BIT
                               6723+*
    184B C0 87 0025            6724+       B     $DISKN                    GO PERFORM DISK I/O
    184F 185C             1850 6725+       DC    AL2(DL4LST)               ADDR OF DISK PARAM LIST
                               6726+*
    1851 F2 00 3C              6727+DL4200 JC    DL4600,*-*                BRANCH OR NOP IF TWICE SET
                               6728+*
    1854 C2 01 0000            6729+DL4900 LA    *-*,@BR                   RESTORE OLD BASE TO RETURN
    1858 C0 87 0000            6730+DL4920 B     *-*                       RETURN TO CALLER

                          185C 6732+DL4LST EQU   *                         LEFT END OF DPL
    185C                  1861 6733+DL4DPL DS    CL(@DPLNG)                DPL SAVE AREA
                          185D 6734+DL4CYL EQU   DL4LST+@DCYL              CYLINDER COUNT BYTE
                          185E 6735+DL4SCD EQU   DL4LST+@DSAD              DISPLACEMENT SECTOR COUNT
                          0060 6736+DL4E96 EQU   96                        TWO DISK SECTOR COUNT PER CYL
                          0030 6737+DL4E48 EQU   48                        ONE DISK SECTOR COUNT PER CYL
                          0018 6738+DL4E24 EQU   24                        TRACK SECTOR COUNT
                          0001 6739+DL4E01 EQU   01                        VALUE TO TEST SECTOR COUNT
                          0001 6740+DL4EFD EQU   01                        VALUE TO SET FIXED DISK BIT
                          0080 6741+DL4ETB EQU   X'80'                     VALUE TO SET TRACK BIT
    1862 0001             1863 6742+DL4C01 DC    IL2'1'                    VALUE TO INCR TO CYLINDER
    1864 0005             1865 6743+DL4C05 DC    IL2'5'                    DISP TO RIGHT END OF DPL
                          1810 6744+DL4C96 EQU   DL4040+@Q                 VALUE TO DECR DISPLACEMENT
                          1834 6745+DL4C24 EQU   DL4070+@Q                 VALUE OF 1 TRACK
                          185F 6746+DL4SCT EQU   DL4LST+@DCNT              POINTER TO DPL SECTOR COUNT
                          1821 6747+DL4C48 EQU   DL4050+@Q                 VALUE TO DECR DISP BY 1 DISK

    1866 5C 00 14 74           6749+DL4500 MVC   DL4WRK(1,@BR),DL4SCT(,@BR) PICKUP SECTOR COUNT
                          1866 6750+DL4SPT EQU   DL4500                    POSSIBLE OVERLAY REFERENCE
    186A 5E 00 14 73           6751+       ALC   DL4WRK(1,@BR),DL4SCD(,@BR) BUMP BY DISPLACEMENT
    186E 7D 30 14              6752+       CLI   DL4WRK(,@BR),DL4E48       TEST FOR CYLINDER OVERLAP
    1871 D0 04 48              6753+       BNH   DL4070(,@BR)              BRANCH BACK IF NO OVERLAY
    1874 5F 00 14 36           6754+       SLC   DL4WRK(1,@BR),DL4C48(,@BR) DECREMENT WORK BY 48
    1878 5F 00 74 14           6755+       SLC   DL4SCT(1,@BR),DL4WRK(,@BR) SUBTRACT WORK FROM COUNT
    187C 7C 87 67              6756+       MVI   DL4200+@Q(,@BR),@UCB      SET TWICE SWITCH
    187F 5C 00 13 73           6757+       MVC   DL4SAV(1,@BR),DL4SCD(,@BR) SAVE SECTOR DISP IN WORK AREA
    1883 78 01 5E              6758+       TBN   DL4100+@Q(,@BR),DL4EFD    DISK BIT ON IN Q CODE ?
    1886 D0 90 48              6759+       BF    DL4070(,@BR)              BRANCH NOT ON
    1889 5E 00 13 36           6760+       ALC   DL4SAV(1,@BR),DL4C48(,@BR) BUMP TO NEXT DISK
    188D D0 87 48              6761+       B     DL4070(,@BR)              RETURN TO CALL I/O
                               6762+*
    1890 5C 00 73 13           6763+DL4600 MVC   DL4SCD(1,@BR),DL4SAV(,@BR) PICKUP NEXT HALF OF I/O
    1894 5E 00 75 74           6764+       ALC   DL4LST+@DBFR1(1,@BR),DL4SCT(,@BR) BUMP CORE ADDRESS
    1898 5E 00 73 74           6765+       ALC   DL4SCD(1,@BR),DL4SCT(,@BR)
    189C 5C 00 74 14           6766+       MVC   DL4SCT(1,@BR),DL4WRK(,@BR) MOVE IN NEW SECTOR COUNT
    18A0 D0 87 1E              6767+       B     DL4035(,@BR)              RETURN FOR SECOND PASS
                               6768+*
                          17FF 6769+DL4WRK EQU   DL4020+@DOP2              1 BYTE WORK AREA FOR SPLIT CALL
                          17FE 6770+DL4SAV EQU   DL4020+@DOP2-1            1 BYTE WORK AREA FOR SPLIT CALL
                          18A3 6771+DL4END EQU   *                         DEFINE END OF CODE
                               6772+***                       END OF DL4ICS                            ***
                               6773 *
                          FFFF 6774        END

TOTAL STATEMENTS IN ERROR IN THIS ASSEMBLY =     0
```

```
                                          CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE  79
$$$$$$  001  0600 3362
$$$NLN  001  00A0 3225   3691  3987
$$ZERO  001  0000 0687   0688  0690  0691  0692  0696
$ABORT  001  0010 0800
$BASIC  001  0080 0858
$BIGCD  001  0080 0934
$BLDPL  001  0579 1067   1069
$BLNOE  001  0569 1057
$BLOAD  001  0522 1048   1050  1053  1066  1067
$BLRTN  001  0550 1056   1057
$BRSAV  001  03C5 0745   0746
$BSADR  001  0587 1072   1074  3421
$BUFPT  001  03E3 0953   0954
$CABLD  001  04B4 1026   1027
$CAERK  001  0469 1003   1006  3692  3988  5568
$CAERR  001  03CD 0751   0753  3690* 3986* 5554* 5556* 5558* 5560* 5563* 6349*
$CAIPL  001  049D 1022   1024
$CALLI  001  0008 0943
$CARDI  001  0001 0714
$CARPL  001  04A1 1024   1026
$CIENT  001  0483 1013   1014
$CIEXT  001  0480 1012   1013
$CIMSK  001  0476 1009   1012
$CISUS  001  0496 1017   1022
$CLBFR  001  0010 0901
$CMDKY  001  0008 0813
$CMODE  001  0002 0863
$CONFG  001  03DD 0926   0936
$CRPOS  001  03E2 0952   0953
$CRTAD  001  044D 0991   0992
$CRTAV  001  0002 0807
$CRTDN  001  0002 0831
$CRTIN  001  03D3 0828   0835
$CRTNO  001  0004 0810
$CRTPU  001  0004 0832
$CRTSP  001  0008 0833
$CRTUP  001  0001 0830
$CRUSH  001  0080 0939
$CSDPL  001  050E 1038   1039
$C0001  001  0464 0995   1001
$DATE   001  043A 0976   0977
$DBGUF  001  03E0 0938   0947
$DBLOK  001  0001 0888
$DFDET  001  03E8 0959   0960
$DISKN  001  0025 0690   3436  3681  3861  3863  3868  3888  4079  4081  5406  5480  5508  5527
                         5896  5936  5979  6013  6046  6055  6065  6078  6583  6724
$DKERR  001  0008 0869
$DKSIZ  001  03D7 0913   0921  0962
$DK100  001  0001 0915
$DK200  001  0002 0916
$DK400  001  0004 0917
$DK600  001  0008 0918
$DK800  001  0010 0919
$DPLSV  001  0449 0987   0989
$DTNMB  001  0040 0734
$DTRDR  001  0040 0822
```

```
$ENDNU  001  0600 1081   1092
$ERDPL  001  046F 1006   1008
$ERFIL  001  0040 0761
$ERHRD  001  0004 0893
$ERKEY  001  0080 0765   5567
$ERLOG  001  0345 0695
$ERMAD  001  0472 1008   1009
$ERPND  001  0004 0866
$ERRCT  001  03CF 0767
$ERRPG  001  03CE 0755   3691* 3987* 5567*
$ERSFL  001  0035 0760
$ERSTK  001  0030 0758
$ER050  001  0363 0696
$ER1N2  001  0050 0763
$EXADR  001  0517 1041   1043
$EXCMD  001  0001 0795
$EXFTR  001  043B 0977   0982
$FCIND  001  0010 0873
$FDIND  001  0040 0880
$FEARR  001  0004 0688
$FEMAP  001  0588 1074   1075
$FILIB  001  03DA 0924   0925
$FITIN  001  0010 0849
$FUIND  001  0020 0878
$GUFIO  001  0583 1071   1072
$GUFIR  001  0008 0723
$HISTE  001  042E 0974   0975
$HIST1  001  0435 0975   0976
$HRDER  001  0020 0819
$INDR1  001  03D4 0835   0861
$INDR2  001  03D5 0861   0886
$INDR3  001  03D6 0886   0913
$INLNO  001  03CF 0753   0755  0767  0774
$INRPT  001  0020 0731
$IOIND  001  03D2 0802   0828
$IOPGS  001  0010 0942
$IOYES  001  0002 0717
$IPLDV  001  05FF 1078   1081
$IRKEY  001  0020 0941
$KEYBD  001  03E1 0947   0952
$KEYCD  001  03C3 0711   0745
$KEYDT  001  0040 0855
$KE090  001  00DE 0691
$KE130  001  01D5 0692
$KYBSY  001  0010 0728
$LDRTN  001  0571 1066
$LEVEL  001  03DF 0936   0938
$LIST   001  0002 0890
$LMRGN  001  03C1 0706   0708
$LNPTR  001  0080 0825
$LOADB  001  054A 1050
$LOADR  001  051A 1043   1046
$LPRIO  001  03EA 0960
$LPROS  001  03E5 0955   0957
$LPRP3  001  03E4 0954   0955
$MOUNT  001  0020 0904
```

```
$MPDWN  001  0001 0804
$NEXTB  001  03E6 0957   0958
$NEXTL  001  03E7 0958   0959
$NOENB  001  0008 0896
$NOLST  001  0004 0720
$NUCBS  001  03C0 0703   0704
$NWRKF  001  0080 0909
$NWRKR  001  0040 0906
$PASWD  001  042D 0973   0974
$PAUSD  001  04BA 1027   1029
$PAUSE  001  0002 0797
$PGMDT  001  0020 0852
$PGMST  001  0010 0816
$PKERT  001  0419 0971   0973
$PLST1  001  0454 0992   0993
$PLST2  001  045B 0993   0994
$PLST3  001  0462 0994   0995
$PRDEV  001  044B 0989   0991
$PRESN  001  0002 0840
$PROCI  001  0001 0837
$PRPOS  001  03C2 0708   0711
$PSDBR  001  04FA 1032
$PSDXR  001  04F2 1031   1032
$PSTEP  001  0004 0798
$PSTMT  001  0008 0799
$PTCH1  001  03F5 0962   0966
$READY  001  0080 0882
$REORD  001  0040 0940
$RLOAD  001  051E 1046   1048  3442
$RMRGN  001  03C0 0704   0706
$RSTR   001  04D6 1029   1031  1033  1038
$RUNIT  001  0001 0776
$SFAID  001  050D 1034
$SPRNT  001  0465 1001   1003
$SRTRN  001  04FE 1033   1034
$STEPT  001  0002 0777
$SWPCR  001  0511 1039   1041
$TABLN  001  03CB 0748   0751
$TFLOW  001  0008 0783
$TRACE  001  0004 0778   4072
$TRALL  001  0010 0784   4374  4379
$TROVR  001  054E 1053   1056
$TRUNK  001  0080 0736
$TRVAR  001  0020 0785   4377
$UNMSK  001  048D 1014   1017  5433  5898
$USRDR  001  03DC 0925   0926
$VMDEF  001  0080 0789   3397
$VOLF1  001  03FE 0968   0969
$VOLF2  001  040E 0970
$VOLID  001  03F6 0966   0967  0971
$VOLR1  001  03F6 0967   0968
$VOLR2  001  0406 0969   0970
$WAITF  001  057F 1069   1071  3437  3682  3864  3889  4082  5407  5481  5509  5528  5897  5937
                         5980  6014  6047  6056  6066  6079
$WFDEF  001  0040 0983
$WFLOK  001  0008 0846
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE  82
$WFNME  001  0443 0982   0987  3879
$WSIND  001  0004 0843
$XIND1  001  03D0 0774   0793  3401  3404  3407  3521  4072  4374  4377  4379  4425
$XIND2  001  03D1 0793   0802
$XIND3  001  03D8 0921   0924  3397  3399
$XPREC  001  0040 0786   3458  3521  4425
$XRSAV  001  03C7 0746   0748
$ZTRAD  001  05A2 1075
$12K    001  0004 0930
$16CKY  001  0008 0932
$16K    001  0002 0929
$22IMP  001  0001 0927
#$$#BL  001  0000 0539
#$$#CK  001  0000 0667
#$$#CN  001  0000 0635
#$$#CO  001  0000 0427
#$$#CS  001  0000 0487
#$$#DR  001  0000 0231
#$$#ER  001  0000 0431
#$$#FS  001  0000 0527
#$$#IN  001  0000 0671
#$$#PW  001  0000 0675
#$$#RS  001  0000 0507
#$$#SA  001  0000 0495
#$$#SS  001  0000 0491
#$$#VU  001  0600 0451
#$$#0T  001  0700 0223
#$$#1T  001  0000 0227
#$$BCO  001  0600 0239
#$$BOV  001  0800 0511
#$$DPR  001  0700 0247
#$$DRE  001  0889 0263
#$$DSP  001  2800 0283
#$$ECM  001  0C00 0543
#$$EFK  001  0C00 0563
#$$ERR  001  0C00 0535
#$$EXM  001  0C00 0423
#$$FIL  001  0E00 0503
#$$FIS  001  0E00 0499
#$$FML  001  0200 0631
#$$FMS  001  0200 0471
#$$GRA  001  0889 0395
#$$GUF  001  0C00 0531
#$$INL  001  0600 0611
#$$INS  001  0600 0235   3500
#$$KAL  001  0C00 0399
#$$KCA  001  0C00 0615
#$$KCH  001  0C00 0367
#$$KCN  001  0C00 0483
#$$KCT  001  0C00 0335
#$$KDE  001  0C00 0331
#$$KDI  001  0D00 0411
#$$KDN  001  0C00 0319
#$$KDO  001  0E00 0415
#$$KED  001  0C00 0255
#$$KEN  001  0C00 0259
```

```
#$$KEX  001  0C00 0279
#$$KGO  001  0C00 0251
#$$KHE  001  0C00 0435
#$$KKE  001  0C00 0663
#$$KLI  001  0C00 0339
#$$KLL  001  0920 0639
#$$KLO  001  0C00 0343
#$$KME  001  0D00 0323
#$$KMO  001  0C00 0267
#$$KNA  001  0C00 0379
#$$KOV  001  0E00 0299
#$$KPA  001  0C00 0275
#$$KPO  001  0C00 0363
#$$KPR  001  0C00 0387
#$$KRE  001  0C00 0307
#$$KRL  001  0700 0403
#$$KRM  001  0C00 0271
#$$KRN  001  0700 0291
#$$KRO  001  0D00 0295
#$$KRS  001  0C00 0619
#$$KRU  001  0C00 0315
#$$KRV  001  0800 0407
#$$KSA  001  0C00 0351
#$$KSE  001  0E00 0391
#$$KSO  001  0C20 0443
#$$KSS  001  0C00 0375
#$$KSV  001  0980 0371
#$$KSY  001  0C00 0383
#$$KWI  001  0C00 0311
#$$KWR  001  0C00 0303
#$$LOA  001  0600 0243   3361
#$$MIP  001  0C00 0439
#$$SDS  001  0C00 0551
#$$SFF  001  0E00 0555
#$$SFL  001  0F00 0547
#$$SFO  001  1500 0519
#$$SFS  001  0C00 0515
#$$SPA  001  0C00 0355
#$$SPO  001  0806 0359
#$$SPS  001  0C00 0347
#$$STR  001  1600 0523
#$$TDC  001  1000 0327
#$$TSY  001  1000 0287
#$$TVK  001  0FC0 0463
#$$UAL  001  0C00 0479
#$$UAT  001  0900 0575
#$$UCD  001  0900 0583
#$$UCN  001  0C00 0567
#$$UCP  001  0700 0571
#$$UDE  001  0C00 0587
#$$UDI  001  0C00 0591
#$$UEX  001  0C00 0475
#$$UIN  001  0C00 0579
#$$UPA  001  0C00 0559
#$$UPO  001  0C00 0627
#$$UPT  001  0C00 0623
```

```
                                         CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                          VER 15, MOD 00  05/08/20  PAGE  84
#$$VCR  001  2000 0419
#$$VLO  001  0600 0455
#$$VOD  001  0600 0459
#$$VVM  001  0000 0467
#$$VXI  001  0600 0447
#$$ZDU  001  1100 0599
#$$ZLB  001  1100 0643
#$$ZLO  001  1100 0603
#$$ZLV  001  0F00 0659
#$$ZL1  001  0F00 0647
#$$ZL2  001  0F00 0651
#$$ZL3  001  0C00 0655
#$$ZTR  001  1000 0595
#$$ZUT  001  0C00 0607
#$#BLN  001  18D4 0538
#$#CKT  001  2118 0666
#$#CNF  001  2000 0634
#$#COR  001  0800 0426
#$#CSA  001  1000 0486
#$#DRT  001  0000 0230
#$#ERM  001  0928 0430
#$#FSP  001  1880 0526
#$#INV  001  212C 0670
#$#PWR  001  2300 0674
#$#RSP  001  1780 0506
#$#SAV  001  1180 0494
#$#SSA  001  1128 0490
#$#VUF  001  0B08 0450
#$#0TR  001  0000 0222
#$#1TR  001  0080 0226
#$@#BL  001  0001 0540
#$@#CK  001  0004 0668
#$@#CN  001  0001 0636
#$@#CO  001  003A 0428
#$@#CS  001  003A 0488
#$@#DR  001  0008 0232
#$@#ER  001  0032 0432
#$@#FS  001  0030 0528
#$@#IN  001  003A 0672
#$@#PW  001  00C0 0676
#$@#RS  001  0030 0508
#$@#SA  001  0108 0496
#$@#SS  001  0001 0492
#$@#VU  001  0002 0452
#$@#0T  001  0018 0224
#$@#1T  001  0018 0228
#$@BCO  001  0018 0240
#$@BOV  001  0018 0512
#$@DPR  001  0005 0248
#$@DRE  001  0001 0264
#$@DSP  001  0004 0284
#$@ECM  001  0006 0544
#$@EFK  001  0002 0564
#$@ERR  001  0003 0536
#$@EXM  001  0003 0424
#$@FIL  001  0009 0504
```

```
#$@FIS  001  0009 0500
#$@FML  001  0052 0632
#$@FMS  001  0052 0472
#$@GRA  001  0003 0396
#$@GUF  001  0010 0532
#$@INL  001  0010 0612
#$@INS  001  0010 0236   3499
#$@KAL  001  000F 0400
#$@KCA  001  000C 0616
#$@KCH  001  000C 0368
#$@KCN  001  0010 0484
#$@KCT  001  0009 0336
#$@KDE  001  0010 0332
#$@KDI  001  0005 0412
#$@KDN  001  0010 0320
#$@KDO  001  000C 0416
#$@KED  001  000E 0256
#$@KEN  001  0006 0260
#$@KEX  001  0003 0280
#$@KGO  001  0002 0252
#$@KHE  001  000C 0436
#$@KKE  001  0006 0664
#$@KLI  001  0011 0340
#$@KLL  001  0001 0640
#$@KLO  001  0008 0344
#$@KME  001  0003 0324
#$@KMO  001  0004 0268
#$@KNA  001  0008 0380
#$@KOV  001  0009 0300
#$@KPA  001  0005 0276
#$@KPO  001  000D 0364
#$@KPR  001  0009 0388
#$@KRE  001  0002 0308
#$@KRL  001  0004 0404
#$@KRM  001  0003 0272
#$@KRN  001  0003 0292
#$@KRO  001  000A 0296
#$@KRS  001  000A 0620
#$@KRU  001  0003 0316
#$@KRV  001  000D 0408
#$@KSA  001  0011 0352
#$@KSE  001  0004 0392
#$@KSO  001  000D 0444
#$@KSS  001  000B 0376
#$@KSV  001  0002 0372
#$@KSY  001  000F 0384
#$@KWI  001  0002 0312
#$@KWR  001  0002 0304
#$@LOA  001  0013 0244
#$@MIP  001  000D 0440
#$@SDS  001  0004 0552
#$@SFF  001  0008 0556
#$@SFL  001  0005 0548
#$@SFO  001  0003 0520
#$@SFS  001  0011 0516
#$@SPA  001  0004 0356
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE  86
#$@SPO  001  0003 0360
#$@SPS  001  0001 0348
#$@STR  001  0002 0524
#$@TDC  001  0003 0328
#$@TSY  001  0003 0288
#$@TVK  001  0001 0464
#$@UAL  001  0011 0480
#$@UAT  001  000C 0576
#$@UCD  001  000B 0584
#$@UCN  001  0009 0568
#$@UCP  001  000F 0572
#$@UDE  001  000E 0588
#$@UDI  001  0008 0592
#$@UEX  001  000E 0476
#$@UIN  001  000F 0580
#$@UPA  001  0004 0560
#$@UPO  001  0005 0628
#$@UPT  001  0012 0624
#$@VCR  001  0008 0420
#$@VLO  001  0002 0456
#$@VOD  001  0016 0460
#$@VVM  001  0030 0468
#$@VXI  001  0002 0448
#$@ZDU  001  0008 0600
#$@ZLB  001  0002 0644
#$@ZLO  001  000C 0604
#$@ZLV  001  0006 0660
#$@ZL1  001  0007 0648
#$@ZL2  001  000D 0652
#$@ZL3  001  000A 0656
#$@ZTR  001  0001 0596
#$@ZUT  001  0014 0608
#$BCOM  001  0080 0238
#$BOLV  001  1780 0510
#$DPRI  001  014C 0246
#$DREA  001  0200 0262
#$DSPL  001  0240 0282
#$ECMA  001  1900 0542
#$EFKE  001  1990 0562
#$ERRP  001  18C0 0534
#$EXMS  001  07D4 0422
#$FILN  001  1724 0502
#$FIST  001  1700 0498
#$FMLN  001  1E00 0630   3466
#$FMST  001  0D00 0470   3465
#$GRAP  001  0690 0394
#$GUFU  001  1880 0530
#$INLN  001  1C84 0610   3724
#$INST  001  0020 0234   3498
#$KALL  001  06A4 0398
#$KCAL  001  1CC4 0614
#$KCHA  001  053C 0366
#$KCND  001  0F80 0482
#$KCTL  001  03BC 0334
#$KDEL  001  035C 0330
#$KDIS  001  0744 0410
```

```
#$KDNT  001  0300 0318
#$KDOV  001  0780 0414
#$KEDI  001  0188 0254
#$KENA  001  01C4 0258
#$KEXT  001  0234 0278
#$KGOS  001  0180 0250
#$KHEL  001  0A30 0434
#$KKEY  001  2100 0662
#$KLIS  001  0400 0338
#$KLLA  001  2004 0638
#$KLOG  001  0444 0342
#$KMER  001  030C 0322
#$KMOU  001  0204 0266
#$KNAM  001  05C0 0378
#$KOVM  001  0290 0298
#$KPAS  001  0220 0274
#$KPOO  001  0508 0362
#$KPRT  001  063C 0386
#$KREA  001  02BC 0306
#$KRLA  001  0700 0402
#$KRMO  001  0214 0270
#$KRNU  001  0280 0290
#$KROV  001  028C 0294
#$KRSU  001  1D24 0618
#$KRUN  001  02CC 0314
#$KRVL  001  0710 0406
#$KSAV  001  0488 0350
#$KSET  001  0680 0390
#$KSOV  001  0AC8 0442
#$KSSP  001  0594 0374
#$KSVL  001  058C 0370
#$KSYM  001  0600 0382
#$KWID  001  02C4 0310
#$KWRI  001  02B4 0302
#$LOAD  001  0100 0242
#$MIPP  001  0A80 0438
#$SDSY  001  192C 0550
#$SFFI  001  193C 0554
#$SFLO  001  1918 0546
#$SFOV  001  1844 0518
#$SFSY  001  1800 0514
#$SPAC  001  04CC 0354
#$SPOV  001  04DC 0358
#$SPSY  001  0484 0346
#$STRO  001  1850 0522
#$TDCK  001  0350 0326
#$TSYK  001  0250 0286
#$TVKB  001  0BAC 0462
#$UALL  001  0F00 0478
#$UATR  001  1A38 0574
#$UCDI  001  1AD8 0582
#$UCNF  001  19B8 0566
#$UCPL  001  19DC 0570
#$UDEL  001  1B24 0586
#$UDIS  001  1B5C 0590
#$UEXL  001  0EA8 0474
```

```
                                CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE  88
#$UINI  001  1A88 0578
#$UPAC  001  1980 0558
#$UPOV  001  1D24 0626
#$UPTF  001  1D5C 0622
#$VCRT  001  07B4 0418
#$VLOA  001  0B80 0454
#$VODK  001  0B88 0458
#$VVMR  001  0C00 0466
#$VXIT  001  0B00 0446
#$ZDUM  001  1BA4 0598
#$ZLBM  001  2008 0642
#$ZLOA  001  1BC4 0602
#$ZLVR  001  20B0 0658
#$ZL1M  001  2010 0646
#$ZL2M  001  2030 0650
#$ZL3M  001  2088 0654
#$ZTRA  001  1B9C 0594
#$ZUTM  001  1C14 0606
#@#BAD  001  0455 3178
#@#IO1  001  0459 3186   4139
#@#IO2  001  045D 3187
#@#TAT  001  0941 3214
#@#TBA  001  09A1 3218
#@#TFS  001  0941 3212
#@#TSY  001  0941 3216
#@#VFP  001  0700 3204
#@#VLP  001  093D 3207
#@#WDB  001  050C 3199
#@#WFT  001  0500 3197
#@@#BA  001  0001 3179
#@@#IO  001  0001 3191
#@@#SC  001  0002 3188   4140  4173  4178
#@@#TA  001  0010 3215
#@@#TB  001  0010 3219
#@@#TS  001  0005 3217
#@@#TW  001  0020 3213
#@@#VM  001  0100 3208
#@@#WD  001  00BD 3200
#@@#WF  001  0003 3198
#@@#04  001  0004 3190
#@@#08  001  0008 3189
#@@BOV  001  0018 3167
#@@ECM  001  0006 3181
#@@ERR  001  0003 3175
#@@GUF  001  0010 3171
#@@LDS  001  0002 3177
#@@SDS  001  0004 3173
#@@SFF  001  0008 3185
#@@SFL  001  0005 3183
#@@SFO  001  0005 3193
#@@SFS  001  0011 3169
#@@VSF  001  0010 3221
#@@VSL  001  000F 3222
#@@VTR  001  0001 3206
#@BOVL  001  0400 3166
#@ECMA  001  0481 3180
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE  89
#@ERRP  001  0441 3174
#@GUFU  001  0401 3170
#@LDSV  001  044D 3176   4172  4177
#@SDSY  001  04AD 3172
#@SFFI  001  04BD 3184
#@SFLO  001  0499 3182
#@SFOV  001  04C4 3192
#@SFSY  001  0480 3168
#@VSFI  001  09A1 3220
#@VTRL  001  0708 3205
#@WAF1  001  0401 3165
#@WAR1  001  0400 3164
#LOAD   001  0607 3365
#LOADR  001  0000 0001
@$D1BF  001  0008 2639
@$D1DC  001  0000 2638   3920  4032
@$D1DF  001  001E 2643
@$D1DP  001  0016 2642
@$D1DV  001  000E 2641
@$D1E1  001  0000 2632
@$D1FS  001  000A 2640
@$D1SW  001  001F 2645   3897  3909* 3933  4065
@$D2AS  001  0002 2650
@$D2BS  001  0003 2657   4003* 4038* 4045*
@$D2CB  001  0005 2660
@$D2CF  001  0001 2649
@$D2CP  001  0005 2658
@$D2CS  001  0004 2659
@$D2CY  001  0006 2661
@$D2DA  001  0007 2662
@$D2DC  001  0000 2654
@$D2DD  001  0009 2663
@$D2EE  001  000F 2666
@$D2E1  001  0040 2653   4188
@$D2FS  001  000B 2664
@$D2IO  001  0001 2655
@$D2LC  001  000D 2665
@$D2PN  001  000A 2651   3879*
@$D2SF  001  000B 2652
@$D2VB  001  0002 2656   4000* 4037* 4044*
@$L1BF  001  0008 2672
@$L1DC  001  0001 2671
@$L1DF  001  0008 2674
@$L1DP  001  0008 2675
@$L1DV  001  0006 2676
@$L1E   001  0020 2670   4116
@$L1FS  001  0002 2673
@$L2AS  001  0001 2682
@$L2BS  001  0001 2689
@$L2CB  001  0001 2692
@$L2CF  001  0002 2681
@$L2CP  001  0002 2690
@$L2CS  001  0001 2691
@$L2DA  001  0002 2693
@$L2DC  001  0001 2686
@$L2DD  001  0002 2694
```

```
@$L2E   001  0010 2685   4117
@$L2FS  001  0002 2695
@$L2HD  001  0040 2680
@$L2IO  001  0001 2687
@$L2LC  001  0002 2696
@$L2PN  001  0008 2684   3879
@$L2SF  001  0002 2683
@$L2VB  001  0001 2688
@$MBCD  001  0020 2710
@$MBCR  001  0008 2712
@$MBEN  001  000C 2700
@$MBND  001  0000 2707
@$MBPD  001  0080 2708   3920  4032
@$MBPT  001  0010 2711
@$MBPU  001  0001 2703
@$MBSD  001  0040 2709   3920  4032
@$M2CI  001  0008 2727
@$M2CO  001  0004 2728
@$M2EF  001  0002 2702
@$M2FI  001  0080 2716
@$M2FO  001  0040 2717
@$M2FP  001  0020 2718
@$M2FT  001  0010 2721
@$M2NS  001  00FF 2701
@@E001  001  0000 2520   2522
@@E003  001  0001 2522   2524
@@E004  001  0002 2524   2526
@@E005  001  0003 2526   2528
@@E006  001  0004 2528   2530
@@E007  001  0005 2530   2532
@@E008  001  0006 2532   2534
@@E009  001  0007 2534   2536
@@E010  001  0008 2536   2538
@@E011  001  0009 2538   2540
@@E012  001  000A 2540   2542
@@E013  001  000B 2542   2544
@@E014  001  000C 2544   2546
@@E015  001  000D 2546   2548
@@E016  001  000E 2548   2550
@@E017  001  000F 2550   2552
@@E018  001  0010 2552   2554
@@E019  001  0011 2554   2556
@@E020  001  0012 2556   2558
@@E021  001  0013 2558   2560
@@E023  001  0014 2560   2562
@@E024  001  0015 2562   2564
@@E025  001  0016 2564   2566
@@E026  001  0017 2566   2568
@@E027  001  0018 2568   2570
@@E028  001  0019 2570   2572
@@E029  001  001A 2572   2574
@@E030  001  001B 2574   2576
@@E031  001  001C 2576   2578
@@E032  001  001D 2578   2580
@@E035  001  001E 2580   2582
@@E036  001  001F 2582   2584
```

```
@@E037  001  0020 2584   2586
@@E038  001  0021 2586   2588
@@E039  001  0022 2588   2590
@@E040  001  0023 2590   2592
@@E041  001  0024 2592   2594
@@E042  001  0025 2594   2596
@@E043  001  0026 2596   2598
@@E044  001  0027 2598   2600
@@E045  001  0028 2600   2602
@@E046  001  0029 2602   2604
@@E060  001  002A 2604   2606
@@E080  001  002B 2606
@@E100  001  0000 1992   1994
@@E101  001  0001 1994   1996
@@E102  001  0002 1996   1998
@@E103  001  0003 1998   2000
@@E110  001  0004 2000   2002
@@E112  001  0005 2002   2004
@@E113  001  0006 2004   2006
@@E114  001  0007 2006   2008
@@E115  001  0008 2008   2010
@@E116  001  0009 2010   2012
@@E117  001  000A 2012   2014
@@E120  001  000B 2014   2016
@@E122  001  000C 2016   2018  6349
@@E123  001  000D 2018   2020
@@E124  001  000E 2020   2022
@@E129  001  000F 2022   2024
@@E130  001  0010 2024   2026
@@E131  001  0011 2026   2028
@@E133  001  0012 2028   2030
@@E134  001  0013 2030   2032
@@E135  001  0014 2032   2034
@@E136  001  0015 2034   2036
@@E137  001  0016 2036   2038
@@E138  001  0017 2038   2040
@@E139  001  0018 2040   2042
@@E142  001  0019 2042   2044
@@E143  001  001A 2044   2046
@@E150  001  001B 2046   2048
@@E151  001  001C 2048   2050
@@E160  001  001D 2050   2052
@@E162  001  001E 2052   2054
@@E163  001  001F 2054   2056
@@E164  001  0020 2056   2058
@@E200  001  0021 2058   2060
@@E205  001  0022 2060   2062
@@E210  001  0023 2062   2064
@@E211  001  0024 2064   2066
@@E212  001  0025 2066   2068
@@E213  001  0026 2068   2070
@@E215  001  0027 2070   2072
@@E216  001  0028 2072   2074
@@E217  001  0029 2074   2076
@@E220  001  002A 2076   2078
@@E221  001  002B 2078   2080
```

```
                                        CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE  92
@@E222  001  002C 2080   2082
@@E223  001  002D 2082   2084
@@E225  001  002E 2084   2086
@@E226  001  002F 2086   2088
@@E227  001  0030 2088   2090
@@E228  001  0031 2090   2092
@@E229  001  0032 2092   2094
@@E230  001  0033 2094   2096
@@E232  001  0034 2096   2098
@@E234  001  0035 2098   2100
@@E237  001  0036 2100   2102
@@E240  001  0037 2102   2104
@@E241  001  0038 2104   2106  3117
@@E242  001  0039 2106   2108
@@E248  001  003A 2108   2110
@@E249  001  003B 2110   2112
@@E250  001  003C 2112   2114  5554
@@E251  001  003D 2114   2116
@@E252  001  003E 2116   2118  5563
@@E253  001  003F 2118   2120  5558
@@E254  001  0040 2120   2122  5556
@@E255  001  0041 2122   2124
@@E256  001  0042 2124   2126  5560
@@E300  001  0043 2126   2128
@@E301  001  0044 2128   2130
@@E302  001  0045 2130   2132
@@E303  001  0046 2132   2134
@@E304  001  0047 2134   2136
@@E305  001  0048 2136   2138
@@E308  001  0049 2138   2140
@@E310  001  004A 2140   2142
@@E315  001  004B 2142   2144
@@E316  001  004C 2144   2146
@@E320  001  004D 2146   2148
@@E325  001  004E 2148   2150
@@E330  001  004F 2150   2152
@@E335  001  0050 2152   2154
@@E338  001  0051 2154   2156
@@E340  001  0052 2156   2158
@@E350  001  0053 2158   2160
@@E351  001  0054 2160   2162
@@E352  001  0055 2162   2164
@@E360  001  0056 2164   2166
@@E361  001  0057 2166   2168
@@E362  001  0058 2168   2170
@@E371  001  0059 2170   2172
@@E380  001  005A 2172   2174
@@E390  001  005B 2174   2176
@@E400  001  005C 2176   2178
@@E410  001  005D 2178   2180
@@E415  001  005E 2180   2182
@@E417  001  005F 2182   2184
@@E420  001  0060 2184   2186
@@E430  001  0061 2186   2188
@@E432  001  0062 2188   2190
@@E433  001  0063 2190   2192
```

```
@@E450  001  0064 2192   2194
@@E451  001  0065 2194   2196
@@E460  001  0066 2196   2198
@@E461  001  0067 2198   2200
@@E464  001  0068 2200   2202
@@E465  001  0069 2202   2204
@@E466  001  006A 2204   2206
@@E467  001  006B 2206   2208
@@E469  001  006C 2208   2210
@@E470  001  006D 2210   2212
@@E471  001  006E 2212   2214
@@E473  001  006F 2214   2216
@@E474  001  0070 2216   2218
@@E475  001  0071 2218   2220
@@E476  001  0072 2220   2222
@@E477  001  0073 2222   2224
@@E478  001  0074 2224   2226
@@E479  001  0075 2226   2228
@@E480  001  0076 2228   2230
@@E481  001  0077 2230   2232
@@E482  001  0078 2232   2234
@@E483  001  0079 2234   2236
@@E484  001  007A 2236   2238
@@E485  001  007B 2238   2240
@@E486  001  007C 2240   2242
@@E487  001  007D 2242   2244
@@E488  001  007E 2244   2246
@@E489  001  007F 2246   2248
@@E490  001  0080 2248   2250
@@E491  001  0081 2250   2252
@@E492  001  0082 2252   2254
@@E493  001  0083 2254   2256
@@E494  001  0084 2256   2258
@@E495  001  0085 2258   2260
@@E496  001  0086 2260   2262
@@E497  001  0087 2262   2264
@@E498  001  0088 2264   2266
@@E500  001  0089 2266   2268
@@E501  001  008A 2268   2270
@@E530  001  008B 2270   2272
@@E531  001  008C 2272   2274
@@E535  001  008D 2274   2276
@@E540  001  008E 2276   2278
@@E541  001  008F 2278   2280
@@E542  001  0090 2280   2282
@@E543  001  0091 2282   2284
@@E544  001  0092 2284   2286
@@E545  001  0093 2286   2288
@@E546  001  0094 2288   2290
@@E547  001  0095 2290   2292
@@E548  001  FFFF 2496
@@E549  001  0096 2292   2294
@@E550  001  0097 2294   2296
@@E551  001  0098 2296   2298
@@E552  001  0099 2298   2300
@@E553  001  009A 2300   2302
```

```
@@E554  001  009B 2302   2304
@@E555  001  009C 2304   2306
@@E556  001  009D 2306   2308
@@E558  001  009E 2308   2310
@@E570  001  009F 2310   2312
@@E571  001  00A0 2312   2314
@@E572  001  00A1 2314   2316
@@E573  001  00A2 2316   2318
@@E574  001  00A3 2318   2320
@@E575  001  FFFF 2498
@@E578  001  00A4 2320   2322
@@E579  001  FFFF 2500
@@E580  001  FFFF 2502
@@E585  001  00A5 2322   2324
@@E595  001  FFFF 2504
@@E597  001  FFFF 2506
@@E598  001  FFFF 2508
@@E600  001  00A6 2324   2326
@@E601  001  00A7 2326   2328
@@E602  001  00A8 2328   2330
@@E603  001  00A9 2330   2332
@@E604  001  00AA 2332   2334
@@E606  001  00AB 2334   2336
@@E607  001  00AC 2336   2338
@@E608  001  00AD 2338   2340
@@E609  001  00AE 2340   2342
@@E610  001  00AF 2342   2344
@@E611  001  00B0 2344   2346  3690
@@E612  001  00B1 2346   2348
@@E613  001  00B2 2348   2350  3986
@@E614  001  00B3 2350   2352
@@E700  001  00B4 2352   2354
@@E701  001  00B5 2354   2356
@@E710  001  00B6 2356   2358
@@E712  001  00B7 2358   2360
@@E713  001  00B8 2360   2362
@@E714  001  00B9 2362   2364
@@E715  001  00BA 2364   2366
@@E716  001  00BB 2366   2368
@@E717  001  00BC 2368   2370
@@E718  001  00BD 2370   2372
@@E720  001  00BE 2372   2374
@@E721  001  00BF 2374   2376
@@E723  001  00C0 2376   2378
@@E724  001  00C1 2378   2380
@@E725  001  00C2 2380   2382
@@E726  001  00C3 2382   2384
@@E727  001  00C4 2384   2386
@@E728  001  00C5 2386   2388
@@E729  001  00C6 2388   2390
@@E730  001  00C7 2390   2392
@@E732  001  00C8 2392   2394
@@E752  001  00C9 2394   2396
@@E753  001  00CA 2396   2398
@@E754  001  00CB 2398   2400
@@E755  001  00CC 2400   2402
```

```
SYMBOL  LEN VALUE DEFN   REFERENCES
@@E756  001  00CD 2402   2404
@@E757  001  00CE 2404   2406
@@E758  001  00CF 2406   2408
@@E759  001  00D0 2408   2410
@@E760  001  00D1 2410   2412
@@E761  001  00D2 2412   2414
@@E762  001  00D3 2414   2416
@@E763  001  00D4 2416   2418
@@E764  001  00D5 2418   2420
@@E765  001  00D6 2420   2422
@@E766  001  00D7 2422   2424
@@E767  001  00D8 2424   2426
@@E768  001  00D9 2426   2428
@@E769  001  00DA 2428   2430
@@E770  001  00DB 2430   2432
@@E771  001  00DC 2432   2434
@@E772  001  00DD 2434   2436
@@E773  001  00DE 2436   2438
@@E774  001  00DF 2438   2440
@@E775  001  00E0 2440   2442
@@E776  001  00E1 2442   2444
@@E777  001  00E2 2444   2446
@@E778  001  00E3 2446   2448
@@E779  001  00E4 2448   2450
@@E780  001  00E5 2450   2452
@@E781  001  00E6 2452   2454
@@E782  001  00E7 2454   2456
@@E783  001  00E8 2456   2458
@@E784  001  00E9 2458   2460
@@E785  001  00EA 2460   2462
@@E786  001  00EB 2462   2464
@@E790  001  00EC 2464   2466
@@E791  001  00ED 2466   2468
@@E792  001  00EE 2468   2470
@@E793  001  00EF 2470   2472
@@E794  001  00F0 2472   2474
@@E795  001  00F1 2474   2476
@@E796  001  00F2 2476   2478
@@E797  001  00F3 2478   2480
@@E798  001  00F4 2480   2482
@@E800  001  FFFF 2510
@@E801  001  FFFF 2512
@@E802  001  FFFF 2514
@@E803  001  FFFF 2516
@@E804  001  FFFF 2518
@@E900  001  00F5 2482   2484  3113
@@E901  001  00F6 2484   2486  3115
@@E902  001  00F7 2486   2488  3114
@@E903  001  00F8 2488   2490  3116
@@E905  001  00F9 2490   2492
@@E906  001  00FA 2492   2494
@@E910  001  00FB 2494   3112
@ARR    001  0008 0016   4547  5418  5477  5493  6253  6346  6534* 6535  6536* 6537  6692* 6693
                         6694* 6695
@ASIGN  001  007C 0071
@ASTER  001  005C 0069
```

```
@BCRDL  001  0050 0088
@BE     001  0081 0043
@BF     001  0090 0052
@BH     001  0084 0041
@BL     001  0082 0042
@BLANK  001  0040 0065   4703  6382
@BM     001  0082 0054
@BNE    001  0001 0046
@BNH    001  0004 0044
@BNL    001  0002 0045
@BNM    001  0002 0057
@BNOL   001  0020 0050
@BNOZ   001  0008 0049
@BNP    001  0004 0056
@BNZ    001  0001 0058
@BOL    001  00A0 0048
@BOZ    001  0088 0047
@BP     001  0084 0053
@BR     001  0001 0013   3392  3393* 3412  3417  3426  3426  3427  3427  3428  3428  3516  3517*
                         3526  3527  3528  3532  3533  3533  3534  3534  3535  3535  3536  3536
                         3537  3537  3546  3547  3551  3552  3557  3560  3565  3565  3566  3568
                         3569  3569  3574  3578  3578  3579  3580  3581  3581  3582  3586  3586
                         3588  3588  3594  3595  3595  3596  3596  3602  3603  3603  3604  3604
                         3608  3617  3617  3618  3618  3619  3619  3620  3620  3629  3630  3634
                         3636  3638  3642  3643  3643  3644  3645  3646  3646  3647  3651  3651
                         3653  3653  3659  3660  3660  3661  3661  3667  3668  3668  3669  3669
                         3673  3673  3674  3678  3678  3856  3857* 3883  3884  3901  3902  3906
                         3907  3908  3908  3909  3910  3922  3922  3924  3924  3928  3936  3937
                         3947  3948  3949  3949  3950  3950  3951  3957  3963  3965  3966  3968
                         3968  3973  3975  3980  3980  3992* 3993  4014* 4025* 4027  4032  4051*
                         4368  4369* 4370  4376  4400  4410  4444  4445  4446  4446  4457  4468
                         4469  4476  4477  4485  4495  4500  4500  4501  4506  4507  4507  4513
                         4514  4514  4515  4516  4516  4517  4517  4518  4519  4520  4521  4522
                         4523  4527  4528  4530  4540  4552  4562  4568  4573  4574  4581  4582
                         4593  4603  4609  4614  4615  4622  4623  4635  4649  4659  4665  4670
                         4671  4678  4679  4701  4733  4742  4744  4751  4760  4762  4764  4781
                         4790  4792  4799  4808  4810  4812  4825  4830  4831  4834  4841  4842
                         4863  4865  4867  4882  4883  4888  4892  4904  4905  4910  4918  4927
                         4936  4938  4946  4948  4949  4958  4963  4971  4980  4982  4991  4996
                         5002  5003  5017  5018  5023  5028  5032  5044  5045  5050  5056  5065
                         5074  5076  5084  5089  5091  5092  5101  5106  5112  5121  5123  5132
                         5137  5157  5162  5165  5174  5191  5192  5193  5209  5217  5222  5224
                         5225  5229  5230  5232  5233  5235  5242  5243  5245  5245  5248  5248
                         5251  5252  5253  5253  5254  5258  5263  5264  5264  5266  5266  5267
                         5267  5268  5268  5273  5274  5275  5275  5282  5282  5283  5283  5288
                         5299  5301  5307  5316  5333  5334  5335  5352  5362  5367  5369  5371
                         5372  5379  5380  5381  5385  5386  5387  5387  5388  5388  5390  5391
                         5391  5398  5403  5403  5419  5423  5434  5436  5437  5442  5444  5446
                         5451  5453  5453  5454  5460  5460  5462  5463  5470  5472  5492  5493
                         5497  5497  5502  5502  5505  5505  5521  5521  5524  5524  5532  5533
                         5534  5535  5535  5537* 5538  5539* 5566  5931* 5942* 5956  5959  5968
                         5968* 6002* 6008* 6010  6024  6025  6035  6035* 6036  6038  6257* 6262*
                         6271  6276  6277  6277  6278  6283* 6289* 6296  6297  6297  6298  6341
                         6343  6344* 6346  6348  6350  6350  6360  6360  6365  6365  6366  6366
                         6367  6367  6368  6368  6369  6369  6373  6374  6374  6377  6383  6384
                         6389  6390  6390  6392* 6522  6531  6533* 6534  6535  6536  6537  6539
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                            VER 15, MOD 00  05/08/20  PAGE  97
                         6540  6540  6541  6542  6542  6544  6544  6545  6546  6546  6550  6550
                         6551  6555  6555  6556  6558  6558  6559  6559  6560  6560  6561  6561
                         6562  6562  6568  6569  6570  6570  6571  6576  6576  6577  6577  6579
                         6579  6585* 6688  6689  6691* 6692  6693  6694  6695  6697  6698  6698
                         6699  6701  6702  6704  6706  6706  6707  6707  6708  6710  6712  6713
                         6713  6714  6716  6718  6719  6719  6720  6720  6721  6721  6722  6729*
                         6749  6749  6751  6751  6752  6753  6754  6754  6755  6755  6756  6757
                         6757  6758  6759  6760  6760  6761  6763  6763  6764  6764  6765  6765
                         6766  6766  6767
@BT     001  0010 0051
@BZ     001  0081 0055
@B1     001  0001 0063   6376  6381
@CADDR  001  0002 0142   1841  1842  1843  3066  3093  3423  3425  3433  3435  3437  3680  3682
                         3740  3862  3864  3889  4058  4060  4068  4075  4080  4082  4148  4155
                         4162  4168  4174  4179  4432  4435  4446  4733  4742  4781  4790  4825
                         4883  4911  4946  4964  5003  5023  5051  5089  5107  5307  5386  5405
                         5407  5422  5504  5507  5509  5523  5526  5528  5615  5616  5619  5623
                         5714  5715  5716  5895  5897  5909  5937  5977  5980  6000  6012  6014
                         6045  6047  6054  6056  6064  6066  6077  6079  6164  6540  6698
@CARDL  001  0060 0087
@CHARA  001  00C1 0072
@CHARF  001  00C6 0073
@CHARR  001  00D9 0074
@CHARZ  001  00E9 0075
@CLOFF  001  0010 0094
@CLON   001  0011 0093
@COMMA  001  006B 0066
@CPLUS  001  004E 0079
@DADDR  001  0002 0140   3528  3724  6539  6604  6697
@DBFR1  001  0004 0129   6764*
@DBFR2  001  0005 0130
@DCALK  001  0001 0081
@DCBCY  001  0009 0115   1670  6133  6140  6161
@DCBT1  001  0050 0117   1673  5893
@DCNT   001  0003 0128   3430* 3431* 6746
@DCST1  001  0040 0116   1671  5907  6162
@DCTRL  001  0000 0125
@DCYL   001  0001 0126   6544* 6734
@DD2    001  0003 0030   5534*
@DGET   001  0001 0134   3477  3495  4138  4144  4176  5730  6132  6139  6146
@DOLAR  001  005B 0068
@DOP2   001  0004 0028   4432* 4435* 4733* 4781* 4883* 4946* 5023* 5089* 5307* 6535* 6539* 6540*
                         6602  6603  6693* 6697* 6698* 6769  6770
@DPLNG  001  0006 0132   6541  6600  6699  6733
@DPOS   001  0000 0133
@DPUT   001  0002 0135   3486  3736  4151  4158  4164  4171  5723  5740  6153  6160
@DSAD   001  0002 0127   3908* 6542* 6546* 6550  6551* 6555* 6558* 6562  6568* 6576* 6579* 6601
                         6735
@DSBCY  001  0004 0106   1608
@DSCS1  001  0000 0107   1609
@DSIVF  001  0003 0138
@DSPIN  001  0002 0131
@DTRSZ  001  0018 0085
@DVBCY  001  0007 0108   1667  3487  3737  4145  4152  4159  4165  5724  5731  5741  6147  6154
@DVRFY  001  0031 0136
@DWAIT  001  00FF 0137
```

```
                                 CROSS REFERENCE
SYMBOL   LEN VALUE DEFN   REFERENCES                                  VER 15, MOD 00  05/08/20  PAGE  98
@DWBCY   001  0005 0103   1664
@DWSIZ   001  00C0 0105
@DWTB1   001  0003 0104   1665
@DZERO   001  00F0 0064
@D1      001  0002 0026   3543  3608* 3626  3673* 3912  3928* 3935* 4431* 4434* 4479* 4513* 4587*
                          4629* 4685* 4707  4718* 4731* 4739  4744* 4750* 4752  4752* 4777* 4787
                          4792* 4798* 4800  4800* 4822  4831* 4833* 4834* 4840* 4844  4850  4853
                          4855* 4857* 4901  4919  4919* 4936* 4955  4972  4972* 4991* 5004* 5041
                          5057  5057* 5074* 5098  5113  5113* 5132* 5164* 5171  5180  5180* 5190*
                          5191* 5192* 5193  5274* 5288* 5302* 5322  5322* 5332* 5333* 5334* 5335
                          5390* 5398* 5430  5437* 5444* 5463* 5516  5516* 5532  5533  5533* 5534
                          5773  5774  5775  5776  5913* 5917* 5926  5926* 6024* 6360
@EOF     001  001C 0077
@EOFTC   001  0075 0162
@EOS     001  001E 0076   1680  4398
@FDDBC   001  0000 0195
@FDE1    001  000C 0200
@FDFNA   001  000B 0198
@FDHLN   001  0002 0208
@FDLNC   001  0002 0193
@FDNSC   001  0003 0210
@FDSD    001  0000 0206
@FLACE   001  0009 0197
@FLDBC   001  0001 0196
@FLENT   001  0004 0201
@FLFNA   001  0002 0199
@FLHLN   001  0002 0209
@FLLNC   001  0002 0194
@FLNSC   001  0001 0211
@FLSD    001  0001 0207
@HDRLN   001  0007 0092
@IAR     001  0010 0017
@INDEX   001  0001 0156   0157
@INST3   001  0003 0032
@INST4   001  0004 0033
@INST5   001  0005 0034
@INST6   001  0006 0035
@I1IAR   001  00C0 0020
@LINSZ   001  00F4 0084
@MAPEN   001  0005 0089
@MINCR   001  2000 0083
@MINUS   001  0060 0080
@NOP     001  0080 0040   5998  6355  6429  6581  6702
@NUMBR   001  007B 0070
@OPD2    001  0004 0029
@OP1     001  0003 0027   4393* 4394* 4456* 4457* 4494* 4495* 4547* 4561* 4562* 4602* 4603* 4642*
                          4658* 4659* 4830* 5003* 5179* 5224* 5232* 5236* 5321* 5371* 5373* 5418*
                          5419* 5471* 5472* 5477* 5492* 5493* 5535* 5975* 6253* 6271* 6272* 6343*
                          6346* 6531* 6537* 6689* 6695*
@OP2     001  0005 0031
@PCTRL   001  0000 0149
@PDATA   001  0003 0151
@PGCSZ   001  0020 0082   0083
@PPLNG   001  0004 0148
@PRCNT   001  0001 0150
@PRETR   001  00C0 0154
```

```
SYMBOL  LEN VALUE DEFN   REFERENCES

@PRINT  001  0040 0152   0154
@PSR    001  0004 0015
@PWAIT  001  00FF 0158
@P1IAR  001  0020 0018
@P2IAR  001  0040 0019
@Q      001  0001 0024   3526* 3575  4430* 4433* 4436* 4437* 4438* 4439* 4440* 4441* 4442* 4443*
                         4447* 4529* 4631* 4632* 4637* 4687* 4688* 4690* 4705  4720  4732* 4773*
                         4774  4778  4847* 4884* 4885  4889  4893  4913* 4932* 4947* 4966* 4987*
                         5024* 5025  5029  5033  5070* 5090* 5128* 5197* 5259  5306* 5313  5339*
                         5532* 5997  6092  6423  6427  6582  6701* 6702* 6712* 6718* 6744  6745
                         6747  6756* 6758
@REGL   001  0002 0012
@RETRN  001  0080 0153   0154
@RLDWN  001  004F 0159
@RTRNC  001  0080 0161
@SBLN   001  0005 0170
@SBLNL  001  0002 0184
@SCTSZ  001  0100 0100
@SDFLN  001  0007 0090
@SDF0   001  0000 0166
@SDF1   001  0001 0167
@SDF2   001  0002 0168
@SDF3   001  0003 0169
@SECCY  001  0030 0086
@SIST   001  0001 0181
@SLASH  001  0061 0067
@SLAST  001  0002 0183
@SMIDL  001  0003 0182
@SNULL  001  0080 0173
@SONLY  001  0000 0180
@STEXT  001  0007 0172
@STYPE  001  0006 0171
@SYLVL  001  0005 3148
@TBCNT  001  0000 0160
@TBLEF  001  0010 0155   0157
@TBLIX  001  0011 0157
@UCB    001  0087 0039   6425  6756
@UPARW  001  005A 0078   3131
@VADDR  001  0002 0141   1401  1837  1849  1850  1851  1851  1865  1868  1870  1894  1895  1896
                         1934  1937  1940  1943  1946  1949  1952  1961  1964  1967  1970  1973
                         3067  3093  3594  3602  3659  3667  3708  4476  4522  4581  4622  4678
                         5174  5273  5316
@VENTA  001  0056 0113   1668  1923  5951  6036
@VMDDV  001  00FE 0114   3738
@VMFD1  001  0000 0109   4153
@VMFD2  001  0001 0110   4160  4166
@VMRS3  001  0002 0112
@VMTRL  001  0001 0111
@VOLID  001  0006 0091
@VQ     001  0001 0025   4704  4719  5429  5515  5538
@WSFIT  001  0500 0101
@WSTBL  001  0503 0102
@XR     001  0002 0014   3541* 3542  3542* 3546* 3551* 3552  3557  3558  3560  3564  3566  3567
                         3567  3579  3594  3602  3624* 3625  3625* 3629* 3634* 3635  3636  3638
                         3644  3659  3667  3878* 3879  3893* 3897  3909  3911  3911* 3915  3920
                         3933  3939* 3941* 3981* 4000  4003  4013* 4026* 4037  4038  4044  4045
                         4050* 4064* 4065  4391* 4392  4392* 4393  4394  4398  4400  4401  4403
                         4405  4408  4410  4456  4457  4462* 4463  4468* 4474* 4475  4475* 4476
                         4478* 4480  4484* 4485* 4494  4495  4499* 4501  4506* 4512* 4515* 4518*
                         4520* 4521* 4522  4527* 4528* 4529  4532* 4534* 4535  4539* 4540* 4548
                         4550  4552* 4561  4562  4566* 4568  4573* 4579* 4580  4580* 4581  4586*
                         4588  4592* 4593* 4602  4603  4604* 4609  4614* 4620* 4621  4621* 4622
                         4627* 4628  4633  4635* 4642  4643* 4644  4648* 4649* 4658  4659  4660*
                         4665  4670* 4676* 4677  4677* 4678  4683* 4684  4702* 4703  4704  4719
                         4737* 4738  4738* 4742  4750  4751  4753* 4754  4754* 4755  4785* 4786
                         4786* 4790  4798  4799  4801* 4802  4802* 4803  4821* 4825  4840  4841
                         4842* 4843  4843* 4847  4848* 4849  4849* 4858  4899* 4900  4900* 4905*
                         4910* 4911  4918  4920* 4921  4921* 4922  4953* 4954  4954* 4958* 4963*
                         4964  4971  4973* 4974  4974* 4975  5039* 5040  5040* 5045* 5050* 5051
                         5056  5058* 5059  5059* 5060  5096* 5097  5097* 5101* 5106* 5107  5112
                         5114* 5115  5115* 5116  5169* 5170  5170* 5174  5179  5181* 5182  5182*
                         5183  5185  5195* 5196  5196* 5197  5201* 5202  5204  5209* 5210  5212
                         5217* 5222* 5224  5226  5228  5230* 5232  5234  5236  5240* 5241  5241*
                         5242* 5243  5251  5263  5273  5277* 5311* 5312  5312* 5316  5321  5323*
                         5324  5324* 5325  5327  5337* 5338  5338* 5339  5343* 5345  5347  5352*
                         5353  5355  5357  5362* 5367* 5370  5371  5373  5377* 5378  5378* 5379*
                         5380  5386  5393* 5420* 5429  5454* 5464* 5465  5465* 5471  5492  5513*
                         5514  5514* 5515  5536* 5538  5543* 5565* 5566* 5914* 5915  5920* 5921
                         5927  5927* 5928* 5929  5930* 5941* 5946  5951  5956  5959  5960  5963
                         5963* 5975  5978* 5985  5985  6001* 6015* 6025  6255  6256* 6261* 6263
                         6266  6266  6272  6273  6282* 6284  6284  6288* 6294  6294  6304  6304
                         6348  6357  6373  6376  6376* 6381  6381* 6382  6389
@ZERO   001  0000 0062   3873  3897  3899  3904  3910  3933  3937  4065  4703* 6551  6701
B$ADMK  001  0001 1305
B$ADSW  001  159D 1304
B$ARMK  001  0001 1290
B$ARSW  001  0A45 1289
B$BABF  001  1D00 1095
B$BCKT  001  1590 1217
B$BDPL  001  19E8 1169
B$BDSA  001  19EA 1170
B$BINO  001  1A6A 1233
B$BRLN  001  19F1 1168
B$BROP  001  1AF7 1274
B$BRVA  001  19EF 1167
B$BRVP  001  19EE 1166
B$BTAB  001  1996 1165
B$CADR  001  1AF9 1275
B$CASA  001  0000 1110
B$CASC  001  0671 1114
B$CASM  001  0608 1112
B$CBAS  001  14BB 1240
B$CBFA  001  0CBC 1195
B$CCGT  001  0600 1120
B$CCLS  001  0695 1126
B$CCON  001  001F 1193
B$CDAT  001  0600 1106
B$CDEF  001  0600 1107
B$CDIM  001  0673 1108
B$CDUM  001  0000 1144
B$CEND  001  0600 1142   1143
B$CEOF  001  0600 1143
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                          VER 15, MOD 00  05/08/20  PAGE 101
B$CFOR  001  0600 1115
B$CGET  001  06A3 1123
B$CGSB  001  0690 1121
B$CGTO  001  06B3 1119
B$CIFA  001  0600 1117
B$CIFC  001  0600 1118
B$CIMG  001  0600 1132
B$CINP  001  0600 1127
B$CLTA  001  0000 1109
B$CLTC  001  0669 1113
B$CLTM  001  0600 1111
B$CMAT  001  0600 1133
B$CMGT  001  0665 1134
B$CMIN  001  06D3 1135
B$CMPR  001  069B 1138
B$CMPT  001  069B 1137
B$CMPU  001  0600 1139
B$CMRD  001  06D0 1136
B$CNXT  001  0600 1116
B$CPCT  001  0CA8 1198
B$CPRT  001  0600 1130
B$CPRU  001  0600 1131
B$CPSE  001  06E7 1140
B$CPUT  001  0600 1124
B$CPWA  001  0CA6 1269
B$CRAD  001  150D 1239
B$CRBS  001  1509 1241
B$CREA  001  06CF 1128
B$CREM  001  0000 1105
B$CRMK  001  0001 1317
B$CRSR  001  06E3 1129
B$CRST  001  06A6 1125
B$CRSW  001  0E42 1316
B$CRTN  001  06CF 1122
B$CSBF  001  0600 1092   1106  1107  1108  1111  1112  1113  1114  1115  1116  1117  1118  1119
                         1120  1121  1122  1123  1124  1125  1126  1127  1128  1129  1130  1131
                         1132  1133  1134  1135  1136  1137  1138  1139  1140  1141  1142  1145
                         1146  1147  1148  1149
B$CSCN  001  14B0 1214
B$CSMK  001  0007 1320
B$CSSW  001  14BC 1319
B$CSTP  001  06D6 1141
B$CSTR  001  14CC 1238
B$CSXA  001  2000 1098
B$CTYP  001  0A5F 1192
B$CVPD  001  0C5D 1197
B$CVPG  001  0CA5 1196
B$CWRK  001  F500 1266
B$DIST  001  0700 1158
B$DLNK  001  1B37 1264
B$DL4T  001  1A6B 1235
B$DPWA  001  0E46 1270
B$DST2  001  073A 1159
B$ERMK  001  0007 1293
B$ERSW  001  0993 1292
B$FACA  001  0E53 1201
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE 102
B$FAIS  001  15AC 1218
B$FAIW  001  15A0 1219
B$FCON  001  0A46 1191
B$FORT  001  1B0E 1260
B$FPWA  001  15AC 1271
B$FRMK  001  0007 1311
B$FRSW  001  16CC 1310
B$FSC1  001  0E4C 1202
B$FSC2  001  0E4D 1203
B$FSMK  001  0007 1302
B$FSSW  001  0E5C 1301
B$FSVA  001  0E4F 1204
B$FTND  001  1B0B 1262
B$FTPT  001  1B0D 1261
B$FVME  001  15A2 1223
B$FVMP  001  15A4 1224
B$FVMS  001  15A6 1225
B$FVPE  001  15A8 1220
B$FVPP  001  15AA 1221
B$FVPS  001  15AC 1222
B$GBSW  001  08AF 1295
B$GBWK  001  0001 1296
B$GETC  001  0867 1172
B$GPTR  001  0878 1174
B$GTBF  001  1E00 1096
B$IFMK  001  0007 1314
B$IFSW  001  16E5 1313
B$INVT  001  1B38 1254
B$KWMK  001  0001 1308
B$KWSW  001  159E 1307
B$LBAS  001  185E 1245
B$LBSV  001  18E7 1243
B$LDRP  001  1A00 1093   3700  3701  3702  3703  3704  3705  3725  3740  4109  4110  5616  5762
                         5763  5764  5765  5766  5767  5768  5769  5770  5771  6164
B$LINE  001  07D0 1160
B$LIST  001  1853 1227
B$LRTN  001  18EB 1244
B$LSTR  001  1862 1242
B$LTYP  001  18F2 1228
B$MATR  001  18F3 1230
B$MBMK  001  0007 1329
B$MBSW  001  1903 1328
B$MFBK  001  1B8F 1256
B$MGMK  001  0007 1326
B$MGSW  001  18FF 1325
B$MPMK  001  0007 1332
B$MPSW  001  1981 1331
B$MRMK  001  0007 1323
B$MRSW  001  0DDE 1322
B$NUMC  001  0873 1173
B$NXMK  001  0007 1299
B$NXSW  001  071D 1298
B$PARP  001  0A41 1181
B$PBNL  001  0A01 1187
B$PCAD  001  0A40 1182
B$PCDL  001  09D3 1186
```

```
B$PCPG  001  0A35 1185
B$PECT  001  0A44 1189
B$PERC  001  0A39 1188
B$PFAE  001  0033 1179
B$PFCL  001  009D 1180
B$PFNC  001  094E 1177
B$PFWP  001  0015 1178
B$PNBY  001  0A41 1183
B$PPWA  001  0A35 1268
B$PRM1  001  1AF3 1272
B$PTBF  001  1F00 1097
B$PUTC  001  093A 1176
B$PVAD  001  0A43 1184
B$RMRK  001  1AE6 1237
B$RTRN  001  1AF5 1273
B$SABF  001  1C00 1094
B$SCAN  001  1514 1216
B$SCAT  001  13C8 1211
B$SCON  001  001B 1194
B$SCVT  001  12E0 1209
B$SDPL  001  07DA 1162
B$SFAB  001  0E48 1206
B$SFNT  001  143C 1212
B$SLDT  001  109C 1208
B$SLVT  001  1062 1207
B$SNAT  001  131A 1210
B$SPAT  001  07E0 1163
B$SSTA  001  1BAC 1258
B$STAS  001  061B 1147
B$STIF  001  0606 1149
B$STMA  001  061B 1148
B$STML  001  0600 1146
B$STRL  001  0600 1145
B$SVRB  001  0E46 1205
B$SYMB  001  0DBC 1200
B$TCD2  001  0001 1278
B$TLTH  001  0002 1279   1280
B$TOD1  001  0000 1277
B$TOTB  001  1AF8 1280
B$TTAB  001  1AFA 1276   1280
B$TYPE  001  0739 1161
B$WORK  001  15A0 1265
B$ZDBN  001  19F2 1232
B@ABAS  001  0007 1865   3594* 3602* 4911  5051  5273  5772
B@ACD1  001  0001 1862   1863  3560* 3566  5243  5251
B@ACD2  001  0003 1863   1864  3552  3557* 3567  5253  5263
B@AFLG  001  0000 1857   3558  3564* 3635*
B@ALLA  001  005C 1682
B@AMAX  001  0005 1864   1865  3567* 3579  4918  5056
B@BLNK  001  0040 1691   4548
B@BLSZ  001  0100 1816   1955  1958  1961  1976  1979
B@BREQ  001  0084 1471
B@BRHI  001  0088 1472
B@BRLO  001  0082 1470
B@BRNE  001  0094 1474
B@BRNH  001  0098 1475
```

```
B@BRNL  001  0092 1473
B@CADD  001  0006 1340
B@CADF  001  0058 1381
B@CBAS  001  0003 1868   3659* 3667* 4964  5107  5386
B@CBNX  001  004A 1374
B@CBRA  001  0046 1372
B@CBRC  001  0044 1371
B@CBRD  001  0048 1373
B@CBRS  001  004C 1375
B@CCLS  001  005E 1384
B@CCMC  001  0042 1370
B@CCMF  001  0040 1369
B@CCNT  001  001F 1794
B@CCSA  001  003E 1368
B@CDCA  001  006A 1390
B@CDDL  001  006C 1391
B@CDIV  001  000C 1343
B@CDMN  001  0001 1867   1868  3636  3638* 3644  4971  5112  5380
B@CDWA  001  006E 1392
B@CEOF  001  0070 1393
B@CEOP  001  0068 1389
B@CFCI  001  0016 1348
B@CFN0  001  0012 1346
B@CFN1  001  0014 1347
B@CFOR  001  004E 1376
B@CGET  001  0052 1378
B@CHAR  001  0000 1807
B@CHLT  001  0004 1339
B@CIEX  001  00C5 1767
B@CIMH  001  0066 1388
B@CINI  001  0056 1380
B@CIPI  001  00D7 1770
B@CIS2  001  00E2 1773
B@CMF1  001  0018 1349
B@CMF2  001  001A 1350
B@CMF3  001  001C 1351
B@CMMA  001  006B 1702   4550  5228
B@CMPY  001  000A 1342
B@CMSM  001  001E 1352
B@CNEG  001  0010 1345
B@CNXT  001  0050 1377
B@COLN  001  007A 1704
B@CPMK  001  00FF 1612   1616  1620  1621  1655
B@CPRS  001  0060 1385
B@CPRU  001  0062 1386
B@CPUT  001  0054 1379
B@CPWR  001  000E 1344
B@CRSR  001  005A 1382
B@CRST  001  005C 1383
B@CSA1  001  0036 1364
B@CSA2  001  0038 1365
B@CSB1  001  003A 1366
B@CSC1  001  002A 1358
B@CSD0  001  002E 1360
B@CSD1  001  0030 1361
B@CSD2  001  0032 1362
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE 105
B@CSF1  001  0022 1354
B@CSF2  001  0024 1355
B@CSTA  001  0034 1363
B@CSTC  001  0028 1357
B@CSTF  001  0020 1353
B@CSTH  001  0064 1387
B@CSTX  001  003C 1367
B@CSUB  001  0008 1341
B@CSVC  001  0002 1338
B@CTYP  001  0020 1792
B@CUSC  001  002C 1359
B@CUSF  001  0026 1356
B@CVAR  001  005B 1681   4403  5353
B@DAMK  001  0080 1860   3635
B@DASA  001  00FF 1621
B@DASC  001  0040 1625
B@DASM  001  0038 1623
B@DCGT  001  0050 1631
B@DCLS  001  0054 1637
B@DDAT  001  0024 1617
B@DDEF  001  0034 1618
B@DDIM  001  0004 1619
B@DDUM  001  00FF 1655
B@DEC0  001  00F0 1750   4408  4628  4684  5212  5357
B@DEC1  001  00F1 1751
B@DEC2  001  00F2 1752
B@DEC3  001  00F3 1753
B@DEC4  001  00F4 1754
B@DEC5  001  00F5 1755
B@DEC6  001  00F6 1756
B@DEC7  001  00F7 1757
B@DEC8  001  00F8 1758
B@DEC9  001  00F9 1759
B@DEND  001  0058 1653   1654
B@DEOF  001  0058 1654
B@DFOR  001  0028 1626
B@DGET  001  0040 1634
B@DGSB  001  0020 1632
B@DGTO  001  0044 1630
B@DIFA  001  0048 1628
B@DIFC  001  004C 1629
B@DIGS  001  007B 1684
B@DIMG  001  003C 1643
B@DINP  001  0000 1638
B@DIVD  001  0061 1701
B@DLTA  001  00FF 1620
B@DLTC  001  0040 1624
B@DLTM  001  0038 1622
B@DL01  001  0001 1935   1938
B@DL02  001  0003 1938   1941  5763
B@DL03  001  0005 1941   1944
B@DL04  001  0007 1944   1947  3703
B@DL05  001  0009 1947   1950  5764
B@DL06  001  000B 1950   1953  5765  5767
B@DL07  001  0045 1953   1956  5768
B@DL08  001  0145 1956   1959
```

```
SYMBOL  LEN VALUE DEFN   REFERENCES
B@DL09  001  0245 1959   1962
B@DL10  001  0289 1962   1965  5769
B@DL11  001  02C3 1965   1968  3704  5770
B@DL12  001  02FD 1968   1971  3705  5771
B@DL13  001  0337 1971   1974
B@DL14  001  0371 1974   1977
B@DL15  001  0471 1977   1980
B@DL16  001  0507 1980   3725  3740  5616
B@DMAT  001  0008 1644
B@DMGT  001  0044 1645
B@DMIN  001  0038 1646
B@DMPR  001  0048 1649
B@DMPT  001  004C 1648
B@DMPU  001  0054 1650
B@DMRD  001  003C 1647
B@DNXT  001  0044 1627
B@DPNT  001  004B 1692
B@DPRT  001  002C 1641
B@DPRU  001  0030 1642
B@DPSE  001  0050 1651
B@DPUT  001  0040 1635
B@DREA  001  000C 1639
B@DREM  001  00FF 1616
B@DRSR  001  005C 1640
B@DRST  001  0050 1636
B@DRTN  001  005C 1633
B@DSCY  001  0004 1608
B@DSIF  001  001C 1657
B@DSLT  001  0010 1656
B@DSML  001  0010 1658
B@DSNS  001  0018 1610
B@DSS1  001  0000 1609
B@DSTP  001  0054 1652
B@DTBN  001  0010 1674
B@DTB1  001  0050 1673
B@DTCY  001  0009 1670
B@DTSN  001  0010 1672
B@DTS1  001  0040 1671
B@DTYP  001  0040 1786   5689  5692
B@DURE  001  0020 1504
B@DVCY  001  0007 1667
B@DVC1  001  0056 1668
B@DWCY  001  0005 1664
B@DWT1  001  0003 1665
B@D1MK  001  0080 1858
B@D2MK  001  00C0 1859   3558  3564
B@EOST  001  001E 1680   5202  5345
B@EQUL  001  007E 1706
B@EXPC  001  00C5 1683
B@FOFL  001  005C 1685
B@FVAD  001  0001 1870
B@GETC  001  0001 1809
B@GETE  001  00FF 1810
B@GETS  001  0000 1808
B@GRTR  001  006E 1703
B@ICON  001  0050 1765
```

```
B@LADD  001  0001 1409
B@LADF  001  0002 1450
B@LADV  001  0008 1894   1915
B@LBIN  001  0002 1819   1820  1826  5243  5245  5248
B@LBNX  001  0003 1443
B@LBRA  001  0003 1441
B@LBRC  001  0004 1440
B@LBRD  001  0003 1442
B@LBRS  001  0001 1444
B@LCCA  001  0004 1850
B@LCCC  001  0001 1402   1440
B@LCDV  001  0004 1895   1916
B@LCER  001  0001 1400   1464
B@LCFN  001  0004 1851
B@LCLN  001  0002 1405   1456  1457  1464
B@LCLS  001  0001 1453
B@LCMC  001  0001 1439
B@LCMF  001  0001 1438
B@LCNA  001  0006 1849
B@LCNN  001  0001 1403   1428  1437  1449  1461
B@LCOP  001  0001 1399   1407  1408  1409  1410  1411  1412  1413  1414  1415  1416  1417  1418
                         1419  1420  1421  1422  1423  1424  1425  1426  1427  1428  1429  1430
                         1431  1432  1433  1434  1435  1436  1437  1438  1439  1440  1441  1442
                         1443  1444  1445  1446  1447  1448  1449  1450  1451  1452  1453  1454
                         1455  1456  1457  1458  1459  1460  1461  1462
B@LCRV  001  0013 1893   1913  3642  3727  4431  4708  4731  4732  4947  4948  4949  5090  5091
                         5092  5302  5306  5385
B@LCSA  001  0002 1437
B@LCVA  001  0002 1401   1415  1416  1417  1418  1419  1420  1421  1422  1423  1424  1426  1427
                         1429  1430  1431  1432  1433  1434  1435  1440  1441  1442  1443  1445
                         1446  1447  1459  1460
B@LCXX  001  0001 1404   1436  1448  1450  1454  1455
B@LDAT  001  0004 1563
B@LDCA  001  0003 1459
B@LDDL  001  0003 1460
B@LDDM  001  0004 1823
B@LDEF  001  0003 1564
B@LDIM  001  0003 1565
B@LDIN  001  0004 1822   1823  1824
B@LDIV  001  0001 1412
B@LDMN  001  0002 1820   1849  1850  1862  1863  1864  1867  1894  1895  3552  3557  3560  3566
                         3567  3636  3638  4918
B@LDSN  001  0004 1824
B@LDWA  001  0002 1461
B@LELP  001  0010 1892
B@LEND  001  0003 1592
B@LEOF  001  0001 1462
B@LEOP  001  0001 1458
B@LERC  001  0003 1464
B@LESP  001  0008 1891
B@LESS  001  004C 1693
B@LET$  001  005B 1713
B@LET#  001  007B 1714
B@LET@  001  007C 1715
B@LETA  001  00C1 1717
B@LETB  001  00C2 1719
```

```
B@LETC  001  00C3 1720
B@LETD  001  00C4 1721
B@LETE  001  00C5 1722
B@LETF  001  00C6 1723
B@LETG  001  00C7 1724
B@LETH  001  00C8 1725
B@LETI  001  00C9 1726
B@LETJ  001  00D1 1727
B@LETK  001  00D2 1728
B@LETL  001  00D3 1729
B@LETM  001  00D4 1730
B@LETN  001  00D5 1731
B@LETO  001  00D6 1732
B@LETP  001  00D7 1733
B@LETQ  001  00D8 1734
B@LETR  001  00D9 1735
B@LETS  001  00E2 1736
B@LETT  001  00E3 1737
B@LETU  001  00E4 1738
B@LETV  001  00E5 1739
B@LETW  001  00E6 1740
B@LETX  001  00E7 1741
B@LETY  001  00E8 1742
B@LETZ  001  00E9 1743
B@LEXP  001  0008 1782
B@LFCI  001  0003 1417
B@LFNA  001  0002 1896   1917
B@LFN0  001  0003 1415
B@LFN1  001  0003 1416
B@LFOR  001  0003 1445
B@LFRT  001  0004 1837   1838
B@LGET  001  0003 1447
B@LGSB  001  0005 1571
B@LGTO  001  0004 1570
B@LHLT  001  0001 1408
B@LIEX  001  0002 1768
B@LIFN  001  0003 1831
B@LILP  001  0009 1890   1908  1909  1910  3526  3527  4430  4431  4433  4434  4437  4439  4440
                         4442  4443  4447  5594  5641  5643  5645  5647  5649  5651
B@LIMG  001  0001 1582
B@LIMH  001  0003 1457
B@LINI  001  0002 1449
B@LINP  001  0005 1577
B@LIPI  001  0003 1771
B@LISP  001  0005 1889   1897  1903  1904  1905  3576  3719  4706  4708  4721  4722  4775  4779
                         4886  4890  4894  5026  5030  5034  5260  5627  5629  5631  5633  5635
                         5637
B@LIS2  001  0005 1774
B@LIVT  001  0001 1847
B@LKCL  001  0005 1576
B@LKFR  001  0003 1567
B@LKGT  001  0003 1573
B@LKIF  001  0002 1569
B@LKON  001  0002 1602
B@LKPT  001  0003 1574
B@LKPU  001  000A 1581
```

```
B@LKRR  001  0007 1579
B@LKRT  001  0005 1575
B@LKTO  001  0002 1596
B@LLET  001  0003 1566
B@LL01  001  0002 1934   1935
B@LL02  001  0002 1937   1938
B@LL03  001  0002 1940   1941
B@LL04  001  0002 1943   1944
B@LL05  001  0002 1946   1947
B@LL06  001  0002 1949   1950
B@LL07  001  003A 1952   1953
B@LL08  001  0100 1955   1956
B@LL09  001  0100 1958   1959
B@LL10  001  0044 1961   1962
B@LL11  001  003A 1964   1965
B@LL12  001  003A 1967   1968  3544  4902  5042  5172
B@LL13  001  003A 1970   1971  3627  4956  5099  5314
B@LL14  001  003A 1973   1974
B@LL15  001  0100 1976   1977
B@LL16  001  0096 1979   1980
B@LMAT  001  0003 1583
B@LMF1  001  0003 1418
B@LMF2  001  0003 1419
B@LMF3  001  0003 1420
B@LMGT  001  0006 1584
B@LMIN  001  0008 1585
B@LMPR  001  0008 1588
B@LMPT  001  0006 1587
B@LMPU  001  000D 1589
B@LMPY  001  0001 1411
B@LMRD  001  0007 1586
B@LMSM  001  0003 1421
B@LNEG  001  0001 1414
B@LNEX  001  0004 1568
B@LNXT  001  0003 1446
B@LPAR  001  004D 1694   4401  4405  5210  5355
B@LPRS  001  0002 1454
B@LPRT  001  0005 1580
B@LPRU  001  0002 1455
B@LPSE  001  0005 1590
B@LPUT  001  0002 1448
B@LPWR  001  0001 1413
B@LREA  001  0004 1578
B@LREM  001  0003 1562
B@LRSR  001  0001 1451
B@LRST  001  0001 1452
B@LRTN  001  0006 1572
B@LSA1  001  0003 1433
B@LSA2  001  0003 1434
B@LSB1  001  0003 1435
B@LSC1  001  0003 1427
B@LSDF  001  0004 1817
B@LSD0  001  0003 1429
B@LSD1  001  0003 1430
B@LSD2  001  0003 1431
B@LSF1  001  0003 1423
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                          VER 15, MOD 00  05/08/20  PAGE 110
B@LSF2  001  0003 1424
B@LSKW  001  0002 1833
B@LSNO  001  0002 1826
B@LSPT  001  0003 1841   1844
B@LSTA  001  0003 1432
B@LSTC  001  0003 1426
B@LSTE  001  0004 1597
B@LSTF  001  0003 1422
B@LSTH  001  0003 1456
B@LSTP  001  0004 1591
B@LSTX  001  0002 1436
B@LSUB  001  0001 1410
B@LSVC  001  0001 1407
B@LTHN  001  0004 1598
B@LTYP  001  0001 1827
B@LUFN  001  0002 1834
B@LUSC  001  0002 1428
B@LUSF  001  0001 1425
B@LVPG  001  0100 1921   1924
B@MINS  001  0060 1700
B@MULT  001  005C 1697
B@NAAR  001  001D 1885   1915  1967
B@NCAR  001  001D 1886   1916  1970
B@NCRV  001  001D 1884   1913  1964
B@NDGT  001  000A 1877   1883
B@NEQL  001  007F 1707
B@NFRT  001  000A 1836   1838
B@NICN  001  0006 1879   1881  4430  4431  4706  4708
B@NIEL  001  0007 1881   1897  1903  1908
B@NIFN  001  0018 1830
B@NIVR  001  0001 1880   1881  4433  4434  4713  4721  4722
B@NIVT  001  0057 1846
B@NLDV  001  0122 1883   1905  1910  1961
B@NLRV  001  001D 1882   1904  1909  1952
B@NLTR  001  001D 1876   1882  1883  1884  1885  1886  1887
B@NSKW  001  0004 1832
B@NSPT  001  0028 1840
B@NUFN  001  001D 1887   1917  1973
B@NVPG  001  0100 1920   1924
B@NXHI  001  00E3 1801
B@NXLO  001  001E 1800
B@NXZR  001  0080 1799   1800  1801
B@PLUS  001  004E 1695
B@POWR  001  005A 1696
B@PREC  001  0020 1788   5688  5691
B@PROD  001  0023 1897
B@PRPL  001  0002 1484
B@PRPN  001  0001 1483
B@PRPR  001  0004 1486
B@PRPS  001  0003 1485
B@PRRC  001  0007 1489
B@PRRL  001  0008 1490
B@PRSL  001  0005 1487
B@PRSS  001  0006 1488
B@PTAB  001  0000 1842
B@PTAD  001  0001 1843
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE 111
B@PTSA  001  0002 1844
B@PUD1  001  0006 1500
B@PUD2  001  0007 1501
B@PUI0  001  0001 1494
B@PUI1  001  0004 1495
B@PUI2  001  0005 1496
B@PUNL  001  0002 1498
B@PUNS  001  0003 1499
B@PUTM  001  0010 1503
B@RPAR  001  005D 1698   4633  5226  5234  5370
B@SADV  001  00E8 1915   1918
B@SAVL  001  0B76 1911   1928
B@SAVS  001  065E 1906   1927
B@SCDV  001  0074 1916   1918
B@SCLN  001  005E 1699
B@SCRV  001  0227 1913   1927  1928
B@SDMK  001  0080 1828
B@SEXP  001  0004 1781
B@SFAT  001  0196 1918   1927  1928  1979
B@SFNA  001  003A 1917   1918
B@SFRT  001  0028 1838
B@SIEL  001  003F 1908   1911
B@SIES  001  0023 1903   1906
B@SIGN  001  0010 1790
B@SLDL  001  0A32 1910   1911
B@SLDS  001  05AA 1905   1906
B@SLVL  001  0105 1909   1911
B@SLVS  001  0091 1904   1906
B@SQUO  001  007D 1705
B@STAT  001  0000 1780
B@TASA  001  0012 1515
B@TASC  001  001E 1521
B@TASM  001  0018 1517
B@TASS  001  007B 1522
B@TCGT  001  0030 1530
B@TCLS  001  0042 1536
B@TDAT  001  0006 1511
B@TDEF  001  0009 1512
B@TDIM  001  000C 1513
B@TDUM  001  0078 1554
B@TEND  001  0072 1552
B@TEOF  001  0075 1553
B@TFOR  001  0021 1524
B@TGET  001  0039 1533
B@TGSB  001  0033 1531
B@TGTO  001  002D 1529
B@TIFA  001  0027 1526
B@TIFC  001  002A 1527
B@TIFS  001  007D 1528
B@TIMG  001  0054 1542
B@TINP  001  0045 1537
B@TLTA  001  000F 1514
B@TLTC  001  001B 1518
B@TLTM  001  0015 1516
B@TLTS  001  0079 1519
B@TMAS  001  007C 1523
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE 112
B@TMAT  001  0057 1543
B@TMGT  001  005A 1544
B@TMIN  001  005D 1545
B@TMLS  001  007A 1520
B@TMPR  001  0066 1548
B@TMPT  001  0063 1547
B@TMPU  001  0069 1549
B@TMRD  001  0060 1546
B@TNXT  001  0024 1525
B@TPRT  001  004E 1540
B@TPRU  001  0051 1541
B@TPSE  001  006C 1550
B@TPUT  001  003C 1534
B@TRAC  001  0080 1784   5165  5301  5690  5691  5692
B@TREA  001  0048 1538
B@TREM  001  0003 1510
B@TRSR  001  004B 1539
B@TRST  001  003F 1535
B@TRTN  001  0036 1532
B@TSTP  001  006F 1551
B@VMC1  001  0056 1923
B@VMLB  001  F0CD 1928
B@VMSB  001  F5E5 1927
B@VMSZ  001  0000 1924   1926  1927  1928
B@VMTB  001  0000 1926
B@ZNEG  001  00D0 1797
B@ZPOS  001  00F0 1796
C4BCHC  001  0004 6417
C4BCHR  001  174A 6405   6373* 6374
C4BINI  001  1749 6403   6350
C4BIN2  001  16DE 6340   5223  5231  5368  6341  6344
C4BLEN  002  1746 6415   6389* 6390*
C4BLNK  003  16F9 6423
C4BLOW  001  00F0 6419   6357
C4BLVL  002  0002 6421   6350  6365  6366  6367  6368  6369  6374
C4BNMC  004  16F5 6427
C4BNOP  001  0080 6429
C4BSAV  002  174C 6409   6348* 6390
C4BSPC  001  0087 6425
C4BVAL  002  1748 6401   5225  5233  5369  6350* 6365  6365* 6366  6367  6367* 6368  6368* 6369*
                         6374* 6421
C4BWRK  002  1746 6398   6366* 6369  6415  6421
C4BYT1  001  1747 6400
C4B100  004  16F4 6351   6427
C4B200  003  16F8 6355   6377  6423
C4B300  003  16FB 6357   6383
C4B590  003  172A 6381   6360  6384
C4B600  003  172D 6382   6355
C4B700  003  1736 6389   6358
C4B800  004  173D 6392   6343* 6361
C4B850  004  1741 6394   6346*
C4B900  001  174D 6411   6351* 6360*
C4END   001  174E 6430
DL2C01  002  17DC 6594   6534  6536  6544
DL2C05  002  17DE 6595   6540
DL2C48  001  17D8 6592   6542  6546
```

```
DL2DPL  006  17E4 6600   6541*
DL2END  001  17E7 6605
DL2E01  001  0001 6524   6542  6544  6546  6550  6562  6570
DL2E02  001  0002 6525   6555  6558  6576
DL2E18  001  0018 6526   6556
DL2E60  001  0060 6527   6571
DL2E7C  001  007C 6529   6568
DL2ICS  001  174E 6530   3422  3432
DL2K18  002  17DA 6593   6559
DL2K60  002  17D5 6590   6577
DL2K80  002  17D7 6591   6558  6576
DL2LST  001  17DF 6599   6542* 6544* 6546* 6550  6551* 6555* 6558* 6562  6568* 6576* 6579* 6584
                         6601
DL2PHY  001  17E1 6601
DL2RAD  002  17E6 6604   3412* 3417* 3421* 6555
DL2SAD  005  1766 6602   6562* 6569* 6570* 6571  6577* 6579
DL2SEC  005  176F 6603   6550* 6556  6559* 6560  6560* 6561  6561* 6570
DL2SWH  003  17C4 6582
DL2TSD  001  0083 6528   6569
DL2000  001  1752 6532   6522  6533
DL2001  005  1762 6539   6535* 6602
DL2002  005  176B 6541   6539* 6540* 6603
DL2005  004  1770 6542   6545
DL2006  004  177E 6546   6543
DL2008  004  179B 6560   6557
DL2010  003  17B1 6571
DL2100  004  17BF 6579   6572
DL2110  003  17C3 6581   6582
DL2900  004  17CC 6585   6531* 6581
DL2910  004  17D0 6586   6537*
DL4CYL  001  185D 6734   6706*
DL4C01  002  1863 6742   6692  6694  6706
DL4C05  002  1865 6743   6698
DL4C24  003  1834 6745   6719
DL4C48  003  1821 6747   6713  6754  6760
DL4C96  003  1810 6744   6707
DL4DPL  006  1861 6733   6699*
DL4EFD  001  0001 6740   6712  6758
DL4END  001  18A3 6771
DL4ETB  001  0080 6741   6718
DL4E01  001  0001 6739   6714
DL4E24  001  0018 6738   6716
DL4E48  001  0030 6737   6710  6752
DL4E96  001  0060 6736   6704
DL4ICS  001  17E7 6687   3424  3434  3679  4057  4059  4067  4074  5404  5478  5503  5506  5522
                         5525  5894  5908  5976  6011  6044  6053  6063  6076
DL4LST  001  185C 6732   6725  6734  6735  6746  6764*
DL4SAV  005  17FE 6770   6757* 6760* 6763
DL4SCD  001  185E 6735   6704  6707* 6710  6713* 6716  6719* 6720  6720* 6721  6721* 6722* 6751
                         6757  6763* 6765*
DL4SCT  001  185F 6746   6714  6749  6755* 6764  6765  6766*
DL4SPT  004  1866 6750   6715
DL4WRK  005  17FF 6769   6749* 6751* 6752  6754* 6755  6766
DL4010  001  17EB 6690   6688  6691
DL4020  005  17FB 6697   6693* 6769  6770
DL4030  005  1804 6699   6697* 6698*
```

```
SYMBOL  LEN VALUE DEFN   REFERENCES

DL4035  003  1809 6701   6767
DL4040  003  180F 6704   6708  6744
DL4050  003  1820 6710   6705  6747
DL4060  003  182D 6714   6711
DL4070  003  1833 6716   6745  6753  6759  6761
DL4080  004  1840 6720   6717
DL4100  003  1848 6722   6701* 6712* 6718* 6758
DL4200  003  1851 6727   6702* 6756*
DL4500  004  1866 6749   6750
DL4600  004  1890 6763   6727
DL4900  004  1854 6729   6689*
DL4920  004  1858 6730   6695*
LALAAC  002  07FA 3725   3551  3634
LALAEL  002  07EE 3717   3527* 3536  3537  3617  3618
LALAP1  008  07F0 3744   3534  3536* 3594  3595* 3617* 3619* 3659  3660* 3678*
LALAP2  008  07F2 3745   3533* 3534* 3588  3596* 3653  3661*
LALAP3  008  07F4 3746   3535  3537* 3602  3603* 3618* 3620* 3667  3668*
LALAP4  008  07F6 3747   3535* 3586  3604* 3651  3669*
LALASC  002  069B 3496   3528*
LALASM  001  1CC4 3704   3541
LALB00  001  0000 3453
LALB01  001  0001 3454   3428
LALCEL  002  07FC 3727   3619  3620  3678
LALCSM  001  1CFE 3705   3624
LALCTR  001  068C 3470   3428* 3471
LALCYL  001  068E 3478   3480
LALEBC  001  0002 3709   3565  3569  3578  3579  3643  3644
LALECT  002  07E8 3714   3565  3565* 3574* 3581* 3642* 3646*
LALFDA  001  068F 3479   3426*
LALH00  002  07FE 3728   3546  3552  3629  3636
LALH01  002  0800 3729   3533  3569  3581  3646
LALH02  002  0802 3730   3608  3673
LALH10  002  0804 3731   3557  3560  3638
LALLCT  002  07EA 3715   3566* 3569*
LALLFA  002  068B 3466   3417
LALLOC  001  0607 3376
LALLPI  002  07F8 3724   3528
LALOVR  001  0700 3460   3483  3492
LALPLI  001  0699 3494   3443
LALPMK  001  0040 3458   3399  3401  3404  3407
LALPUT  001  0805 3735   3680
LALRFL  001  068D 3476   3423  3430* 3433
LALSBC  001  0001 3706   3581  3646
LALSCT  001  0690 3482   3426  3427
LALSC5  001  0005 3451   3430  3431
LALSDP  001  0002 3457   3490
LALSDS  001  0695 3488   3427* 3489
LALSFA  002  0689 3465   3412
LALSIZ  002  07EC 3716   3578  3578* 3579* 3586  3588  3595  3596  3603  3604  3643  3643* 3644*
                         3651  3653  3660  3661  3668  3669
LALSRF  001  0000 3459
LALSTD  001  0001 3707   3608  3673
LALVAP  008  07F6 3720   3532* 3744  3745  3746  3747  4183  4184  4185  4186
LALVA1  001  1A00 3700
LALVA2  001  1A02 3701
LALVA3  001  1A04 3702
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE 115
LALVA4  001  1A07 3703   3532
LALWFL  001  0693 3485   3425  3431* 3435
LALX01  001  0687 3464   3428
LALX02  001  0002 3708   3534  3535  3536  3537  3586  3588  3595  3596  3603  3604  3617  3618
                         3619  3620  3651  3653  3660  3661  3668  3669  3678
LALX03  001  0003 3455
LALX08  001  0008 3710   3532
LAL000  001  060B 3391   3392  3393  6080
LAL010  004  0627 3404   3400
LAL020  004  0631 3407   3398
LAL030  005  0638 3412   3402
LAL040  005  0640 3417   3405  3408
LAL050  006  0645 3421   3413
LAL055  004  064B 3422   3429
LAL060  004  0681 3442   3403  3406
LAL100  004  069F 3517   3377
LAL110  005  06B5 3532   3522
LAL120  004  06CE 3541   3609
LAL125  003  06D2 3542   3543  3545  3608*
LAL130  003  06DB 3551
LAL150  003  06F3 3564   3553  3559
LAL160  004  06FE 3567   3570
LAL170  003  070D 3574   3516  3517  3526* 3575  3577
LAL180  004  0714 3579   3582
LAL190  004  0722 3586
LAL2BY  001  0002 3456   3412  3417  3421
LAL200  004  0730 3594
LAL210  004  073F 3602   3587
LAL220  005  074B 3608   3547  3597
LAL400  004  0754 3617
LAL410  004  0764 3624   3674
LAL420  003  0768 3625   3626  3628  3673*
LAL430  003  0771 3634
LAL434  003  0774 3635
LAL436  004  0777 3636
LAL440  003  0782 3642   3637
LAL450  004  0789 3644   3647
LAL460  004  0797 3651
LAL470  004  07A5 3659
LAL480  004  07B4 3667   3652
LAL490  004  07C0 3673   3630  3662
LAL495  004  07CB 3679
LAL500  004  07D7 3686
LAL900  004  07DB 3690   3568  3580  3589  3645  3654
LDFADB  001  0002 4096   4019  4020
LDFAP1  008  07EF 4183   3883* 3899  3901  3902* 4002  4006  4037  4039* 4044  4046*
LDFAP3  008  07F3 4185   3884* 3904  3906  3907* 4118
LDFBF1  001  1A00 4106   3893  3941  3992  4025  4064  4141  4155  4174  4179  4187
LDFBF2  001  1900 4107   3878  4162  4188  4189
LDFBF3  001  1B00 4108   3939  4168
LDFBY0  001  0000 4093   3915  3993  4027
LDFCNT  001  09F8 4130   3968* 3973  3975* 4045  4046  4131
LDFCTB  001  0001 4095   3922  3924  3928  3947  3948  3949  3950  3957  3965  3966  3968  3980
                         3998  4000  4003  4004  4008  4037  4038  4039  4044  4045  4046
LDFDKD  001  09F6 4126   3924* 3949  3963  3966  4127
LDFDMK  001  0080 4100
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE 116
LDFD1R  002  09EF 4116   3928  4014  4051
LDFD2R  002  09F1 4117   4013  4050
LDFEB1  001  1AFF 4187
LDFEB2  001  19FF 4189   3873* 3874  4192
LDFFDR  001  09FA 4137   3869
LDFFDW  001  0A06 4150   4058
LDFFE2  001  1940 4188   3981  4026
LDFFN1  001  0003 4097
LDFFN2  001  0000 4091
LDFF2W  001  0A0C 4157   4060
LDFF3W  001  0A12 4163   3908* 4068
LDFH01  001  09ED 4115   3883  3884  3902  3907  3922  3924  3968  3998  4003  4004  4008  4038
                         4039  4052
LDFILE  001  080B 3855   3686
LDFLE2  001  0070 4101
LDFLFE  001  00E0 4102   3913
LDFLN2  001  0010 4099
LDFLTH  001  00FF 4104   3874
LDFL2E  001  0060 4103   3935  3936
LDFNDD  001  09F5 4123   3922* 3950  3965  4124
LDFNUL  001  0000 4092   3915  3951  3963  3993  4027
LDFOP1  006  0943 4191   4020*
LDFOP2  005  093A 4190   4019*
LDFPGL  001  0001 4094   3883  3884  3901  3902  3906  3907
LDFPR2  002  09F3 4118   4019  4020
LDFRBF  001  0A1E 4175   4080
LDFRIP  008  07F1 4184   3947  3957  3965* 3966* 3998*
LDFR2P  008  07F5 4186   3948
LDFSAV  001  09F9 4133   3901* 3906* 3908  3909  3910* 3936* 3937
LDFSBF  001  0A18 4170   3862
LDFTAP  001  09F4 4122   3947* 3948* 3980
LDFTDT  001  09F7 4129   3949* 3950* 3951  3957  3980  4008* 4052*
LDFTLA  001  0955 4111
LDFTLB  001  1900 4105   4148
LDFTRL  001  0A00 4143   4075
LDFVA2  001  1A02 4109
LDFVA4  001  1A06 4110
LDFX08  001  0008 4098   3973  3975
LDF100  004  081B 3868
LDF110  004  0821 3873
LDF120  004  082B 3878
LDF130  005  0834 3883
LDF145  004  083E 3888
LDF150  004  0844 3893
LDF155  004  0863 3904   3900
LDF157  004  0874 3908   3903
LDF160  003  087F 3911   3898  3912  3914  3928* 3935* 3940  3942
LDF170  003  0888 3920
LDF180  004  0895 3924   3921
LDF190  005  0899 3928   3916  3923
LDF192  003  08AF 3937   3929
LDF194  004  08BD 3941   3938
LDF2BP  001  19FE 4192   3874*
LDF200  005  08C5 3947   3934
LDF202  005  08DD 3957
LDF203  003  08E5 3963
```

```
LDF204  005  08F0 3966   3969
LDF206  003  0900 3973   3967
LDF220  004  090C 3980   3958
LDF225  004  0910 3981   3964
LDF230  004  0917 3986   3856  3857  3905
LDF240  004  0923 3992   3982
LDF244  003  0927 3993   4015  4021
LDF246  005  0936 4000   4190
LDF247  006  0940 4004   4005  4191
LDF248  004  094F 4013
LDF249  006  095B 4019   3999
LDF250  004  096B 4025   3974  3976
LDF260  003  0973 4027   4053
LDF270  005  097F 4037
LDF280  005  0992 4044   4033
LDF290  004  09A2 4050   4040
LDF310  004  09B4 4057   3952  3994  4009  4028
LDF315  004  09D0 4072   4066
LDF317  004  09DD 4079   4073
LDF320  004  09E9 4083
LRAACT  001  0002 6103   5946  5956  5959  5961  5985  6025
LRABB1  001  06A0 6111   5914  5920  5930  5931  5941  6001  6002  6008  6114  6136
LRABCT  001  0002 6104   5902
LRABEQ  001  06A1 6114   5902
LRABMK  001  0007 6109   5919  6007
LRABMT  001  0000 6097
LRABSW  003  154C 6092   5919* 6007*
LRACIN  001  1602 6118   6019
LRACTR  001  160B 6126   6009  6009* 6019* 6029
LRACT0  001  0000 6098   6029
LRADDR  004  1474 5893   5408
LRADIS  001  0001 6100   6024
LRADPG  001  0002 6105   5951
LRAH00  002  1604 6119   5902
LRAINC  001  0004 6108   5963  5968  6035
LRALST  001  00FC 6110   5913
LRAN02  001  0002 6102   6001
LRAN04  002  160A 6122   5928
LRAPCT  001  1611 6135   5974  6071
LRAPDP  001  0000 6099   6010  6036  6038
LRAPGD  001  0003 6107   5946  5956  5956  5959* 5985  5985*
LRAPGV  001  1622 6155   6043* 6052* 6062*
LRAPLB  001  160E 6131   5895
LRAPLS  001  1614 6138   5909  5977
LRAPLV  001  161A 6145   6012
LRAPNO  001  1610 6134   5893* 6071*
LRAPUT  001  1626 6159   6077
LRAP04  002  1608 6121   5917
LRASAV  002  160D 6127   5960* 5961
LRASBE  001  19FF 6113   5946
LRASB1  001  1900 6112   5942  6015  6143  6150  6157
LRASPG  001  1616 6141   5907* 5974*
LRASVA  001  0001 6101   5959
LRAVMD  001  0003 6106   6025
LRAVPG  001  161C 6148   6010* 6038  6043  6052  6062
LRAVPL  001  1620 6152   6045  6054  6064
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE 118
LRAX00  001  0000 6096   5915
LRAX02  002  1606 6120   6000
LRA010  004  1478 5894   6072
LRA012  006  1488 5902
LRA015  004  1492 5907
LRA020  003  14A4 5915   5913* 5917* 5918  5926
LRA030  006  14C3 5926   5916
LRA035  003  14C9 5927   5926*
LRA037  004  14E0 5936   5922
LRA040  004  14E6 5941
LRA045  004  14EA 5942   5981
LRA050  005  14EE 5946   5964
LRA060  003  14F6 5951   5969
LRA070  004  14FC 5956
LRA080  004  1506 5959
LRA090  004  150A 5960   5986
LRA100  003  151E 5968   5958
LRA110  006  1525 5974   5947
LRA120  004  1535 5978   5975*
LRA130  004  1543 5985   5957
LRA200  003  154B 5996   5952  5962  5997  5999  6092
LRA210  004  1560 6007   5996
LRA220  005  156E 6010   6048
LRA230  006  1583 6019   6039
LRA240  004  158E 6025   6024*
LRA250  006  15C1 6052   6037
LRA260  006  15D6 6062   6030
LRA270  006  15E8 6071
LRA280  004  15F2 6076   5903  6057
LSOBCT  001  0002 6316   6264  6266  6274  6284  6294  6304
LSOBEY  002  16D7 6331   6263* 6264
LSOBOT  002  16D3 6329   5921* 5929* 5961  6000* 6264
LSOBSW  001  16D1 6328   6254* 6286  6303*
LSOB00  001  0000 6314   6254
LSOB01  001  0001 6315   6286  6303
LSODEC  002  16D0 6324   6256  6257  6282  6283
LSOECT  001  0004 6318   6276  6277  6278  6296  6297  6298
LSOHLD  004  16DD 6333   6276* 6278  6296* 6298
LSOINC  002  16CE 6323   6261  6262
LSORTA  004  162C 6253   5932  6003
LSOTEY  002  16D9 6332   6273* 6274
LSOTOP  002  16D5 6330   6255* 6274
LSO1ST  001  0003 6317   6266  6276  6277* 6284  6294  6296  6297* 6304
LSO100  004  1640 6261   6267  6290
LSO2ND  001  0007 6319   6266  6277  6278* 6284  6294  6297  6298* 6304
LSO200  004  1665 6273   6285  6305
LSO210  005  1672 6276
LSO250  004  1680 6282
LSO300  004  1690 6286   6295  6299
LSO400  004  1697 6288   6272*
LSO500  004  169B 6289   6271*
LSO600  004  16A3 6294   6275
LSO650  005  16AB 6296
LSO800  004  16BD 6303   6265
LSO900  004  16C9 6306   6253* 6287
LVIAAA  001  0008 5681   4637  4922  5060  5185
```

```
LVIAAC  002  1371 5616   4910  4963  5050  5106  5242  5379
LVIAAP  001  0004 5682   4631  5183
LVIAIV  002  1405 5714   4446* 4781  4883  5023
LVIALC  002  1375 5621   4432
LVIASC  005  1397 5778   4709
LVIATL  001  141D 5750   4462  4499  4566  4604  4660  5195  5337
LVIBDC  001  0008 5592   4530  4853
LVIBF1  001  1900 5605   5779
LVIBF2  001  0700 5606   4702  5513  5536  5537  5727  5736
LVIBIO  001  0711 5781   5461*
LVIBOA  001  0B11 5780   5461
LVIBOF  001  0004 5591   5423  5446  5451
LVIBRS  002  136F 5615   5003
LVIBYC  001  0001 5586   4400  4463  4469  4479  4501  4513  4529  4568  4574  4587  4609  4615
                         4629  4665  4671  4685  4744  4750  4751  4752  4792  4798  4799  4800
                         4841  5002  5005  5436  5453  5470  5497  5502  5505  5521  5524
LVIBY2  001  0002 5589   4500  4507  4514  4516  4517  4830  4971  5056  5112  5225  5233  5251
                         5253  5263  5264  5266  5267  5282  5283  5369  5380  5387  5419  5434
                         5442  5472
LVICAA  001  0002 5683   4690  4975  5116  5327
LVICAP  001  0001 5684   4687  5325
LVICAT  001  1CFE 5771   4676  4953  5096  5311
LVICES  005  1388 5631
LVICFL  009  13B2 5645
LVICHV  002  1379 5623   4733  4946  5089  5307
LVICMB  001  13E9 5663   5623
LVICNT  002  13F9 5700   4892* 4949* 5032* 5092* 5419  5463  5472
LVICSB  001  13D7 5662   4760* 4762* 4980* 4982* 5121* 5123* 5301*
LVICTF  001  0040 5689   4762  4980  5121
LVICTN  001  00C0 5692   4760  4982  5123
LVICTR  002  1401 5710
LVICVM  001  0010 5595   4588  4755
LVICVT  001  1C8A 5769   4579  4737  5777
LVIDNM  001  00F0 5601   4519
LVIDPT  001  0009 5593   4833
LVIDSA  002  13FF 5707   4410* 4519* 4520  4521  4528  4530
LVIDVP  001  0006 5772
LVID01  001  0001 5585   5462
LVIECC  002  1373 5619   5566
LVIECT  002  13FB 5703   5258* 5266  5268* 5385* 5388*
LVIELC  002  1403 5713   4918* 4971* 5056* 5112*
LVIGET  001  1410 5729   5507  5526
LVIHLD  001  13F6 5698   4751* 4799* 4841* 5275* 5391* 5497  5505
LVIH00  002  1365 5610   4476  4522  4581  4622  4678  4742  4790  4825  4905  4958  5045  5101
                         5174  5193  5243  5245  5248  5316  5335
LVIH01  002  1367 5611   4468  4485  4506  4552  4573  4614  4635  4649  4670  4831  4996  5017
                         5084  5137  5162  5192  5209  5217  5222  5230  5264  5268  5299  5334
                         5352  5362  5388  5453
LVIH02  002  1369 5612   4469  4507  4540  4574  4593  4615  4671  4744  4792  4830  4834  4936
                         4991  5074  5132  5191  5288  5333  5367  5398
LVIH03  002  136B 5613
LVIH64  002  136D 5614   5454
LVIIB1  001  0700 5604   5422  5464  5615  5746  5780  5781
LVIICP  001  1A08 5764   4701
LVIINN  001  1412 5732   4701* 5403  5497  5502  5505* 5521  5524* 5733
LVIIVD  001  1A0B 5765   4718
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE 120
LVILAV  002  1377 5622   4435  4446
LVILDP  002  13FD 5706   4500  4500* 4507* 4513  4514  4514* 4515  4516  4516* 4517  4517* 4518
LVILDT  001  1A46 5768   4512
LVILET  001  1C88 5777   4823
LVILPE  009  13D6 5657   5622
LVILSA  001  13F7 5699   4400* 4463  4501  4568  4609  4665
LVILTB  001  1473 5782   4385* 4386  4386* 4420* 4421  4421*
LVILTF  001  0020 5688   4444
LVILTN  001  00A0 5691
LVILUP  001  0008 5594   4436  4438  4441
LVILVM  001  0020 5597   4480  4803
LVILVT  001  1A0C 5767   4474  4785
LVIMKT  001  0030 5598
LVINAT  001  1CC4 5770   4620  4899  5039  5169
LVINEL  009  13CD 5651   5621
LVINES  005  1397 5637   5778
LVINIL  009  13C4 5649
LVINIT  001  0A24 4367   4083
LVINUL  001  0000 5581   4376  4385  4420  4445  4461  4567  4605  4661  4713  5004  5157  5190
                         5332
LVIN1S  005  1392 5635
LVIN2L  009  13BB 5647
LVIN2S  005  138D 5633
LVIOBC  001  0012 5596   5461
LVIOUT  001  140C 5725   5403* 5502* 5521*
LVIPCT  001  1419 5742   4882* 5018* 5460*
LVIPIB  001  06FF 5603   5564*
LVIPIN  001  141A 5743   5002* 5451  5453* 5460  5462* 5744
LVIPLN  001  13F5 5697   4888* 4948* 5028* 5091* 5437  5444
LVIPTL  001  143A 5755   4478  4532  4534  4586  4643  4753  4801  4848  4920  4973  5058  5114
                         5181  5323  5757  5782
LVIPUT  001  140A 5722   5405  5504  5523
LVIP1L  009  13A9 5643
LVIP1S  005  1383 5629
LVIRG1  001  1A03 5763   4911  4964  5051  5107
LVIRLL  001  00FF 5602   5564
LVISIN  001  1413 5735   5524
LVISPM  001  13D2 5783   4445* 4723  5716
LVISPS  001  13CE 5656   4444* 4808* 4810* 4863* 4865* 4904* 4927* 4938* 5044* 5065* 5076* 5165*
                         5658  5783
LVISS1  002  1407 5717   5225* 5245  5248  5251  5264* 5266* 5267* 5273* 5274  5275  5282  5283
                         5283* 5386* 5387* 5390  5391
LVISS2  002  1409 5718   5233* 5253  5263* 5267  5282* 5369* 5380  5387
LVISTN  001  0080 5690
LVISWC  001  1416 5738   5005* 5423  5434  5436* 5442* 5446* 5470*
LVISWO  001  0001 5584   4370  4632  4688  4764  4812  4867  4913  4932  4966  4987  5070  5128
LVIS2L  009  13A0 5641
LVIS2S  005  137E 5627
LVITD0  001  0000 5580   4398  4400  4548  4550  4633  5202  5204  5210  5226  5228  5234  5345
                         5347  5353  5370
LVITD1  001  0001 5587   4392  4401  4403  4408  4410  5174  5212  5241  5316  5355  5378
LVITD2  001  0002 5588   4405  4628  5357
LVITD3  001  0003 5590   4684
LVITLL  001  0039 5599   4386  4421
LVITMK  001  0080 5600   4808  4810  4863  4865  4904  4927  4938  5044  5065  5076
LVITM0  001  13EA 5668   4527  4842
```

```
LVITM1  001  13EB 5669
LVITM2  001  13EC 5670
LVITM3  001  13ED 5671
LVITM4  001  13EE 5672
LVITM5  001  13EF 5673
LVITM6  001  13F0 5674
LVITM7  001  13F1 5675
LVITM8  001  13F2 5676
LVITM9  001  13F3 5677
LVITRL  001  1900 5779   4391  5564
LVITSW  001  13F4 5696   4370* 4376* 4764* 4812* 4867* 5157
LVITT1  003  0B23 5773   4461* 4469* 4479
LVITT2  003  0BF9 5774   4567* 4574* 4587
LVITT3  003  0C41 5775   4605* 4615* 4629
LVITT4  003  0CB4 5776   4661* 4671* 4685
LVIVA1  001  1A00 5762   4882
LVIVA2  001  1A04 5766   5018
LVIVMI  001  1417 5739   5479
LVI0TD  001  0000 5582   4463  4501  4529  4568  4609  4665  4751  4799  4841  4847  4855  4922
                         4975  5060  5116  5164  5197  5339
LVI010  004  0A2B 4374
LVI012  004  0A46 4385   4375
LVI014  004  0A35 4377   4387
LVI015  004  0A54 4391   4378
LVI020  003  0A63 4398   4416
LVI030  003  0A85 4408   4404
LVI040  004  0A92 4415   4486  4541  4594  4650
LVI045  004  0A9A 4420   4380
LVI050  004  0AA4 4425   4381  4399
LVI060  005  0CDF 4701   5138  5143
LVI065  005  0CEB 4704   4430* 4431* 4432* 4705  4707
LVI070  004  0CF0 4713
LVI075  006  0CF8 4718
LVI078  005  0CFE 4719   4433* 4434* 4435* 4718* 4720
LVI080  004  0AFA 4456   4409
LVI090  005  0B09 4463   4470
LVI092  004  0B1D 4474   4464
LVI094  003  0B21 4475   5773
LVI095  004  0B24 4476
LVI096  003  0B35 4480   4479*
LVI098  004  0B38 4484   4456*
LVI1TD  001  0001 5583   4476  4522  4581  4622  4678  4742  4750  4755  4790  4798  4803  4825
                         4840  4857  5183  5185  5325  5327
LVI100  004  0B43 4494   4411
LVI103  004  0B52 4501   4508
LVI105  004  0B64 4512   4502
LVI106  004  0BA7 4534   4531
LVI107  003  0BAB 4535   4513* 4529* 4533
LVI109  004  0BAE 4539   4494*
LVI110  004  0BB9 4547   4415
LVI112  003  0BBD 4548   4553
LVI114  004  0BC6 4551   4547*
LVI116  003  0BCA 4552   4549
LVI120  004  0BD1 4561   4407
LVI125  004  0BE0 4568   4575
LVI130  004  0BF3 4579   4569
```

```
LVI132  003  0BF7 4580   5774
LVI134  003  0C0B 4588   4587*
LVI135  004  0C0E 4592   4561*
LVI140  004  0C19 4602   4402
LVI145  004  0C28 4609   4616
LVI150  004  0C3B 4620   4610
LVI153  003  0C3F 4621   5775
LVI155  004  0C49 4627   4602*
LVI157  003  0C61 4633   4636  4638  4691
LVI159  004  0C6E 4637   4630
LVI160  004  0C76 4642   4634
LVI165  003  0C7E 4644   4629* 4631* 4637* 4685* 4687* 4690*
LVI167  004  0C81 4648   4642*
LVI170  004  0C8C 4658   4406
LVI175  004  0C9B 4665   4672
LVI180  004  0CAE 4676   4666
LVI182  003  0CB2 4677   5776
LVI185  004  0CBC 4683   4658*
LVI187  004  0CD7 4690   4686
LVI188  004  0CDB 4691   4689
LVI200  004  0D03 4731   4714
LVI205  004  0D10 4737   4745
LVI210  003  0D14 4738   4739  4744* 4752
LVI215  005  0D1E 4744   4765
LVI220  005  0D2A 4750   4743
LVI225  003  0D3D 4754   4752*
LVI230  003  0D4C 4762   4756
LVI235  004  0D4F 4763   4761
LVI240  004  0D5A 4773   4436* 4746  4774
LVI242  004  0D5E 4777   4437* 4778
LVI245  004  0D67 4785   4793
LVI250  003  0D6B 4786   4787  4792* 4800
LVI255  005  0D75 4792   4813
LVI260  005  0D81 4798   4791
LVI265  003  0D94 4802   4800*
LVI269  003  0DA3 4810   4804
LVI270  004  0DA6 4811   4809
LVI280  004  0DB1 4821   4794  4822  4830* 4832  4836
LVI285  005  0DBC 4830   4868
LVI290  005  0DDB 4840   4826
LVI292  003  0DE7 4843   4831* 4833* 4844  4853
LVI294  003  0DF3 4849   4834* 4850
LVI296  004  0E04 4857   4854
LVI298  003  0E08 4858   4847* 4855* 4856  4857*
LVI300  003  0E0E 4863
LVI305  003  0E14 4865   4859
LVI310  004  0E17 4866   4864
LVI320  005  0E22 4882   4426  4448
LVI322  004  0E2C 4884   4438* 4885
LVI326  003  0E30 4888   4439* 4889
LVI328  003  0E33 4892   4440* 4893
LVI330  004  0E36 4899   4937
LVI335  003  0E3A 4900   4901  4919  4936*
LVI336  003  0E46 4910
LVI340  004  0E59 4918   4912
LVI345  003  0E67 4921   4919*
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE 123
LVI355  003  0E70 4927
LVI360  004  0E73 4931   4923
LVI370  005  0E7B 4936   4906  4914
LVI380  005  0E87 4946
LVI390  004  0E96 4953   4992
LVI395  003  0E9A 4954   4955  4972  4991*
LVI400  003  0EA3 4963
LVI410  004  0EB5 4971   4965
LVI415  003  0EC3 4974   4972*
LVI420  003  0ECC 4980
LVI425  003  0ED2 4982   4976
LVI430  004  0ED5 4986   4981
LVI440  005  0EDD 4991   4959  4967
LVI450  003  0EE6 4996   4932* 4987*
LVI460  004  0EEC 5001
LVI470  003  0F00 5017   4913* 4997
LVI480  005  0F0B 5023
LVI482  004  0F10 5024   4441* 5025
LVI484  003  0F14 5028   4442* 5029
LVI486  003  0F17 5032   4443* 5033
LVI490  004  0F1A 5039   5075
LVI495  003  0F1E 5040   5041  5057  5074*
LVI500  003  0F2A 5050
LVI510  004  0F35 5056
LVI515  003  0F43 5059   5057*
LVI525  003  0F4C 5065
LVI530  004  0F4F 5069   5061
LVI540  005  0F57 5074   5046  5052
LVI550  003  0F63 5084   4966* 5019
LVI560  005  0F69 5089
LVI570  004  0F78 5096   5133
LVI575  003  0F7C 5097   5098  5113  5132*
LVI580  003  0F85 5106
LVI590  004  0F90 5112
LVI595  003  0F9E 5115   5113*
LVI600  003  0FA7 5121
LVI605  003  0FAD 5123   5117
LVI610  004  0FB0 5127   5122
LVI620  005  0FB8 5132   5102  5108
LVI630  003  0FC1 5137   5070* 5085  5128*
LVI640  004  0FC8 5142
LVI670  003  0FD0 5157   4835
LVI675  003  0FD7 5162   4632*
LVI680  004  0FE5 5169   5289
LVI685  003  0FE9 5170   5171  5180  5288*
LVI690  004  0FF4 5179
LVI695  003  1002 5182   5180* 5191* 5193
LVI700  005  1015 5191   5194
LVI705  003  102C 5196   5190* 5192*
LVI710  004  1034 5201   4394*
LVI713  003  1038 5202   5218
LVI715  003  103E 5204   5197*
LVI720  003  1044 5209
LVI725  003  1053 5217   5205  5211  5278
LVI730  003  105A 5222   5213
LVI733  004  108A 5236   5284
```

```
LVI734  004  108E 5240   5179*
LVI735  004  1098 5243
LVI736  004  10AA 5248   5244
LVI737  004  10B2 5251   5246
LVI738  003  10C0 5258   4447* 5259
LVI740  004  10C6 5263   5265
LVI742  004  10CA 5264   5262
LVI746  004  10D6 5267   5269
LVI748  004  10E2 5273
LVI749  004  10F3 5277   5236*
LVI750  004  10FB 5282   5227
LVI755  005  1107 5288   5175  5184  5186  5203
LVI760  003  1110 5299   4688* 5163
LVI762  004  111D 5306
LVI775  004  1126 5311   5399
LVI777  003  112A 5312   5313  5322  5398*
LVI778  004  1134 5321
LVI779  003  1142 5324   5322* 5333* 5335
LVI780  005  1155 5333   5336
LVI782  003  116C 5338   5332* 5334*
LVI784  004  1174 5343   4393*
LVI785  003  1178 5345   5363  5394
LVI786  003  117E 5347   5339*
LVI788  003  1184 5352
LVI790  003  1199 5362   5348  5354  5356
LVI791  003  11A0 5367   5358
LVI792  004  11B9 5377   5321*
LVI794  003  11CA 5385
LVI795  004  11D1 5387   5389
LVI796  004  11EA 5393   5373*
LVI797  005  11F2 5398   5317  5326  5328  5346
LVI798  004  11FB 5403   5300
LVI800  004  120F 5418   4931  4986  5069  5127
LVI805  004  1218 5420   5003* 5419* 5471* 5472*
LVI810  005  1227 5429   4883* 4884* 4946* 4947* 5004* 5023* 5024* 5089* 5090* 5430  5437* 5443
                         5444*
LVI812  001  1242 5439   5424  5469
LVI813  004  1248 5443   5441
LVI814  005  124C 5444   5435
LVI815  003  125A 5451   5425  5438
LVI820  004  126A 5459   5452
LVI825  004  1280 5464
LVI828  003  1284 5465   5463*
LVI830  004  1287 5469   5455
LVI833  003  128B 5470   5445
LVI835  004  128E 5471   5447
LVI840  004  1297 5473   5418*
LVI850  004  129B 5477   5001  5142  5459
LVI860  004  12AB 5482   5477*
LVI900  003  12AF 5492   4763  4811  4866  5276  5392
LVI910  004  12B5 5497
LVI920  004  12BC 5502
LVI930  004  12D6 5513   5498
LVI933  003  12DA 5514   4731* 4777* 5164* 5302* 5516
LVI935  005  12DD 5515   4732* 4733* 4750* 4773* 4781* 4798* 4840* 5274* 5306* 5307* 5390* 5516*
                         5532  5533  5534
```

```
LVI940  004  12EB 5521
LVI945  004  131F 5538   4368  4369  5532* 5533* 5534*
LVI948  004  1323 5539   5535*
LVI950  004  1327 5543   5492* 5517
LVI955  004  132B 5544   5493*
LVI990  004  132F 5554   4477  4523  4582
LVI991  004  1336 5556   5158
LVI992  004  133D 5558   4623  4679
LVI993  004  1344 5560   5247  5250
LVI994  004  134B 5563   5229  5235  5252  5254  5372  5381
LVI995  006  134F 5564   5555  5559  5561
LVI996  004  1355 5565   4457* 4495* 4562* 4603* 4659* 5224* 5232* 5371*
LVI997  004  135C 5567   5557
LVI998  004  1360 5568
V$APWR  001  0800 2775   2920
V$BFR1  001  5400 2838   3028
V$BFR2  001  5500 2839   3029
V$CBNZ  001  0CB2 2847   2927
V$CCON  001  5120 2854   3025
V$CDCV  001  3100 2851   2980
V$CDSY  001  2E00 2850   2977
V$CFPZ  001  0C70 2845   2926
V$CNXZ  001  0470 2848   2915
V$CSSR  001  5100 2853   3024
V$CZFP  001  04AD 2846   2916
V$DTLN  001  4600 2860   3012
V$DTVR  001  4700 2861   3013
V$FABS  001  1761 2746   2944
V$FACS  001  1400 2762   2936
V$FASN  001  1413 2761   2937
V$FATN  001  1100 2760   2933
V$FCOS  001  0A00 2757   2922
V$FCOT  001  0D00 2755   2928
V$FCSC  001  1725 2759   2943
V$FDEG  001  17DA 2766   2948
V$FDET  001  4540 2769   3011
V$FEXP  001  0500 2753   2917
V$FHCS  001  1500 2765   2938
V$FHSN  001  1557 2764   2939
V$FHTN  001  1593 2763   2940
V$FINT  001  176C 2747   2945
V$FLGT  001  0200 2751   2910
V$FLOG  001  0219 2750   2912
V$FLTW  001  020B 2752   2911
V$FRAD  001  17CB 2767   2947
V$FRND  001  1800 2768   2949
V$FSEC  001  1700 2758   2942
V$FSGN  001  17A7 2748   2946
V$FSIN  001  0A1A 2756   2923
V$FSQR  001  0900 2749   2921
V$FTAN  001  0D28 2754   2929
V$IFCI  001  1B00 2738   2953
V$IFIO  001  1A00 2740   2952
V$ISDN  001  1900 2739   2950
V$KBTL  001  1EAC 2882
V$KBTS  001  0DAC 2881
```

```
V$LPRB  001  4F00 2836   3022
V$LPRT  001  4D00 2834   3020
V$LPR2  001  4E00 2835   3021
V$MADD  001  4007 2783   3000
V$MASN  001  43A0 2781   3007
V$MCON  001  4324 2788   3005
V$MIDN  001  4300 2789   3004
V$MINV  001  4500 2793   3010
V$MMPY  001  4100 2785   3001
V$MSMY  001  4264 2786   3003
V$MSUB  001  4000 2784   2999
V$MTRN  001  4400 2792   3009
V$MZER  001  432B 2790   3006
V$PCH1  001  5200 2874   3026
V$PCH2  001  5300 2875   3027
V$SCDI  001  2A00 2831   2971
V$SCDO  001  2A96 2832   2972
V$SFA2  001  5000 2816   3023
V$SFD1  001  0000 2826   2908
V$SFD2  001  0100 2827   2909
V$SKEY  001  2500 2830   2966
V$SPRT  001  2800 2829   2969
V$VMPL  001  4C06 2868   3019
V$VMPS  001  4C00 2867   3018
V$XKAF  001  1C00 2815   2954
V$XKCA  001  2400 2819   2962
V$XKCL  001  240A 2818   2963
V$XKIN  001  2B00 2814   2973
V$XKLP  001  24AD 2820
V$XKRS  001  240D 2817   2964
V$XMGT  001  3E06 2808   2994
V$XMIN  001  3D00 2807   2992
V$XMPL  001  3F06 2811   2997
V$XMPS  001  3F00 2810   2996
V$XMPT  001  3E0C 2809   2995
V$XMPU  001  3F13 2812   2998
V$XMRD  001  3E00 2806   2993
V$XSGT  001  2100 2801   2959
V$XSIN  001  2B6E 2800   2974
V$XSPR  001  3400 2803   2983
V$XSPT  001  1D00 2802   2955
V$XSPU  001  3800 2804   2987
V$XSRD  001  3300 2799   2982
V$00E1  001  0000 2908
V$01E1  001  0100 2909
V$02E1  001  0200 2910
V$02E2  001  020B 2911
V$02F3  001  0219 2912
V$03CC  001  0300 2913
V$04CC  001  0400 2914
V$04E1  001  0470 2915
V$04E2  001  04AD 2916
V$05E1  001  0500 2917
V$06CC  001  0600 2918
V$07CC  001  0700 2919
V$08E1  001  0800 2920
```

```
V$09E1  001  0900 2921
V$10E1  001  0A00 2922
V$10E2  001  0A1A 2923
V$11CC  001  0B00 2924
V$12CC  001  0C00 2925
V$12E1  001  0C70 2926
V$12E2  001  0CB2 2927
V$13E1  001  0D00 2928
V$13E2  001  0D28 2929
V$14CC  001  0E00 2930
V$15CC  001  0F00 2931
V$16CC  001  1000 2932
V$17E1  001  1100 2933
V$18CC  001  1200 2934
V$19CC  001  1300 2935
V$20E1  001  1400 2936
V$20E2  001  1413 2937
V$21E1  001  1500 2938
V$21E2  001  1557 2939
V$21E3  001  1593 2940
V$22CC  001  1600 2941
V$23E1  001  1700 2942
V$23E2  001  1725 2943
V$23E3  001  1761 2944
V$23E4  001  176C 2945
V$23E5  001  17A7 2946
V$23E6  001  17CB 2947
V$23E7  001  17DA 2948
V$24E1  001  1800 2949
V$25E1  001  1900 2950
V$26E1  001  1A00 2952
V$27E1  001  1B00 2953
V$28E1  001  1C00 2954
V$29E1  001  1D00 2955
V$30CC  001  1E00 2956
V$31CC  001  1F00 2957
V$32CC  001  2000 2958
V$33E1  001  2100 2959
V$34CC  001  2200 2960
V$35CC  001  2300 2961
V$36CC  001  2400 2965
V$36E1  001  2400 2962
V$36E2  001  240A 2963
V$36E3  001  240D 2964
V$37E1  001  2500 2966
V$38CC  001  2600 2967
V$39CC  001  2700 2968
V$40E1  001  2800 2969
V$41CC  001  2900 2970
V$42E1  001  2A00 2971
V$42E2  001  2A96 2972
V$43E1  001  2B00 2973
V$43E2  001  2B6E 2974
V$44CC  001  2C00 2975
V$45CC  001  2D00 2976
V$46E1  001  2E00 2977
```

```
V$47CC  001  2F00 2978
V$48CC  001  3000 2979
V$49E1  001  3100 2980
V$50CC  001  3200 2981
V$51E1  001  3300 2982
V$52E1  001  3400 2983
V$53CC  001  3500 2984
V$54CC  001  3600 2985
V$55CC  001  3700 2986
V$56E1  001  3800 2987
V$57CC  001  3900 2988
V$58CC  001  3A00 2989
V$59CC  001  3B00 2990
V$60CC  001  3C00 2991
V$61E1  001  3D00 2992
V$62E1  001  3E00 2993
V$62E2  001  3E06 2994
V$62E3  001  3E0C 2995
V$63E1  001  3F00 2996
V$63E2  001  3F06 2997
V$63E3  001  3F13 2998
V$64E1  001  4000 2999
V$64E2  001  4007 3000
V$65E1  001  4100 3001
V$66CC  001  4200 3002
V$66E1  001  4264 3003
V$67E1  001  4300 3004
V$67E2  001  4324 3005
V$67E3  001  432B 3006
V$67E4  001  43A0 3007
V$68E1  001  4400 3009
V$69E1  001  4500 3010
V$69E2  001  4540 3011
V$70E1  001  4600 3012
V$71E1  001  4700 3013
V$72CC  001  4800 3014
V$73CC  001  4900 3015
V$74CC  001  4A00 3016
V$75CC  001  4B00 3017
V$76E1  001  4C00 3018
V$76E2  001  4C06 3019
V$77CC  001  4D00 3020
V$78CC  001  4E00 3021
V$79CC  001  4F00 3022
V$80E1  001  5000 3023
V$81E2  001  5100 3024
V$81E3  001  5120 3025
V$82E1  001  5200 3026
V$83E2  001  5300 3027
V$84E1  001  5400 3028
V$85E2  001  5500 3029
V@CDPT  001  0007 3040
V@CHGH  001  0008 3145
V@CMIC  001  0002 3041
V@CMNI  001  00FF 3038
V@CMUL  001  0007 3146
```

```
SYMBOL  LEN VALUE DEFN   REFERENCES
V@CNIX  001  0080 3039
V@COEX  001  001E 3036
V@CPLS  001  00F0 3043
V@CPRC  001  000A 3045
V@CSQR  001  0003 3143
V@CSTR  001  0002 3144
V@CTTA  001  0027 3046
V@DCAD  001  0002 3066   3067
V@DEXP  001  0000 3071
V@DMAN  001  000D 3073   3074
V@DMN1  001  0001 3072
V@DPDF  001  0002 3061
V@DSAD  001  0001 3062
V@DSGN  001  000D 3074
V@DVAD  001  0004 3067
V@EART  001  0001 3044
V@ECRT  001  0038 3117
V@EFUL  001  00F8 3116
V@EINV  001  00FB 3112
V@EIPR  001  00F5 3113
V@ENSV  001  00F7 3114
V@ENUL  001  0000 3111
V@ERPC  001  0020 3042
V@ESAV  001  00F6 3115
V@FEHN  001  0002 3141
V@FEPL  001  0091 3137
V@FERS  001  0003 3140
V@FPGS  001  0081 3136
V@FRET  001  0015 3139
V@FSPC  001  0040 3138
V@FTAB  001  0000 3142
V@KADD  001  004E 3127
V@KCLE  001  006E 3124
V@KDIV  001  0061 3130
V@KEMN  001  006C 3122
V@KEPL  001  006B 3121
V@KMUL  001  005C 3129
V@KPER  001  004B 3132
V@KPST  001  007B 3126
V@KPWR  001  005A 3131
V@KSQR  001  006F 3123
V@KSTO  001  006D 3125
V@KSUB  001  0060 3128
V@LAIP  001  0003 3092   3093
V@LDEX  001  0002 3095
V@LETE  001  0003 3099
V@LEXP  001  0001 3089   3091
V@LFKO  001  0006 3094
V@LINI  001  0200 3098
V@LLKS  001  0010 3091
V@LMAN  001  000F 3090   3091
V@LNOP  001  0015 3096
V@LTBE  001  0007 3093
V@LVPG  001  0100 3097   3098
V@MCHS  001  00C0 3078
V@MCRD  001  0010 3054
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  05/08/20  PAGE 130
V@MDEF  001  0008 3055
V@MEXC  001  0080 3052
V@MEXT  001  0004 3081
V@MICC  001  0010 3037
V@MIPC  001  0080 3079
V@MIPL  001  0020 3085
V@MLST  001  0040 3053
V@MPND  001  0000 3084
V@MPOF  001  0080 3082
V@MPRC  001  0020 3051
V@MSFU  001  0002 3056
V@MSTN  001  0004 3050
V@OALL  001  00F4 3107
V@ONUL  001  00F0 3103   3104
V@OPM1  001  00F2 3105   3106
V@ORTN  001  00F1 3104   3105
V@OSTK  001  00F3 3106   3107
V@PEOF  001  0002 3080
V@PSQ2  001  0014 3083

TOTAL STATEMENTS IN ERROR IN THIS ASSEMBLY =     0

OL105 I  THE CODE LENGTH OF #LOADR IS  6307 DECIMAL.
OL103 I  TOTAL NUMBER OF LIBRARY SECTORS REQUIRED IS   39
         NAME-#LOADR,PACK-R1R1R1,UNIT-R1,RETAIN-P,LIBRARY-R,CATEGORY-000
```

```
        OVERLAY LINKAGE EDITOR CORE USAGE MAP    05/08/20


 START    CATEGORY  NAME AND       CODE LENGTH
ADDRESS               ENTRY    HEXADECIMAL DECIMAL

 0000         0     #LOADR          18A3    6307


OL100 I  THE TOTAL CORE USED BY #LOADR IS  6307 DECIMAL.
OL101 I  THE START CONTROL ADDRESS OF THIS MODULE IS 0000.
OL104 I  TOTAL NUMBER OF LIBRARY SECTORS REQUIRED IS   25
         NAME-#LOADR,PACK-R1R1R1,UNIT-R1,RETAIN-P,LIBRARY-O
```