```
 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  10/02/22  PAGE   1

#KSOVR    MODULE
```

```
ERR LOC   OBJECT CODE     ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  10/02/22  PAGE   2

    0000                          1 #KSOVR START 0
                                  2 *      @SYS  EXP-N
                                213+       PRINT ON
                                214 *      @FXD  EXP-N
                                619+       PRINT ON
                                620 *      @CAN  EXP-N
                                723+       PRINT ON
                                724 *      @ERM  EXP-N
                               1346+       PRINT ON
                               1347 *      @B@E  EXP-N
                               2247+       PRINT ON
                               2248 *      $I$E  EXP-N
                               2402+       PRINT ON
```

```
                      2404 ********************************************************************
                      2405 *   5703-XM1 COPYRIGHT IBM CORP. 1970                               *
                      2406 *            REFER TO INSTRUCTIONS ON COPY RIGHT NOTICE, 120-2083    *
                      2407 *                                                                  *
                      2408 ********************************************************************
                      2409 *STATUS -                                                          *
                      2410 *    VERSION 1 MODIFICATION 0                                       *
                      2411 *                                                                  *
                      2412 *FUNCTION -                                                         *
                      2413 *    * THE SPECIFIED VARIABLE OR ARRAY ELEMENT SYMBOL IS CONVERTED   *
                      2414 *      TO A VIRTUAL ADDRESS                                         *
                      2415 *    * THE SPECIFIED CONSTANT IS CONVERTED TO A FORM SUITABLE FOR     *
                      2416 *      STORAGE IN VIRTUAL MEMORY                                     *
                      2417 *    * THE INTERNAL FORM IS MOVED TO VIRTUAL MEMORY AT THE ADDRESS    *
                      2418 *      ASSOCIATED WITH THE SYMBOL                                    *
                      2419 *                                                                  *
                      2420 *ENTRY POINTS -                                                     *
                      2421 *    * KSOVRL HAS ONLY ONE ENTRY POINT, KSOVRL, THE FIRST EXECUTABLE  *
                      2422 *      INSTRUCTION                                                  *
                      2423 *    * THE CALLING SEQUENCE IS                                        *
                      2424 *         B     $RLOAD                                              *
                      2425 *         DC    AL2'DPL'                                            *
                      2426 *      WHERE DPL IS THE PARAMETER LIST THAT CONTAINS THE PARAMETERS   *
                      2427 *      NECESSARY FOR $RLOAD TO GET KSOVRL FROM DISK                   *
                      2428 *                                                                  *
                      2429 *INPUT -                                                            *
                      2430 *    * PRIMARY INPUT BUFFER - 256 BYTES, CONTAINS THE SET COMMAND     *
                      2431 *      LINE AS ENTERED                                              *
                      2432 *    * SYMBOL AND ARRAY TABLES                                        *
                      2433 *       * LETTER VARIABLE TABLE (LVT)    - 58 BYTES, 29 2-BYTE ENTRIES *
                      2434 *       * LETTER DIGIT TABLE (LOT)       - 580 BYTES, 290 2-BYTE ENTRIES
                      2435 *       * CHARACTER VARIABLE TABLE (CVT) - 58 BYTES, 29 2-BYTE ENTRIES *
                      2436 *       * ARITHMETIC ARRAY TABLE (NAT)   - 58 BYTES, 29 2-BYTE ENTRIES *
                      2437 *       * CHARACTER ARRAY TABLE (CAT)    - 58 BYTES, 29 2-BYTE ENTRIES *
                      2438 *    * FUNCTION AND ARRAY TABLE (FAT) - 406                           *
                      2439 *       * ARITHMETIC ARRAY DOPE VECTORS  - 29 8-BYTE ENTRIES           *
                      2440 *       * CHARACTER ARRAY DOPE VECTORS   - 29 4-BYTE ENTRIES           *
                      2441 *    * VIRTUAL MEMORY - CONTAINS THE PROGRAM VARIABLE TO BE MODIFIED   *
                      2442 *    * SET COMMON PARAMETER BLOCK - 256 BYTES                          *
                      2443 *                                                                  *
                      2444 *OUTPUT -                                                           *
                      2445 *    * VIRTUAL MEMORY - UPDATED WITH THE SET CONSTANT                 *
                      2446 *                                                                  *
                      2447 *EXTERNAL REFERENCES -                                              *
                      2448 *        $XIND1 - PRIMARY EXECUTION INDICATOR                        *
                      2449 *        SDISKN - SYSTEM DISK IOCR                                   *
                      2450 *        $CARPL - NORMAL SYSTEM ENTRY                                *
                      2451 *        SCAERK - SYSTEM ERROR MESSAGE ENTRY                         *
                      2452 *        $CAERR - $CAERK ERROR CODE PARAMETER                        *
                      2453 *        SCANIT - COMMAND LINE DELIMITER SCAN ROUTINE                *
                      2454 *        C4BIN2 - DECIMAL TO BINARY CONVERSION ROUTINE               *
                      2455 *        C4BVAL - BINARY OUTPUT PARAMETER FROM C4BIN2                 *
                      2456 *        I$MDFY - PAGING ROUTINE ENTRY                               *
                      2457 *        I$LDXR - PAGING ROUTINE ENTRY                               *
                      2458 *        I$VADR - PAGING ROUTINE VIRTUAL ADDRESS PARAMETER           *
                      2459 *        I$CADR - PAGING ROUTINE CORE ADDRESS PARAMETER              *
```

#KSOVR - SET KEYWORD COMMAND OVERLAY

```
                        2460 *                                                      *
                        2461 *EXITS, NORMAL -                                       *
                        2462 *    KSOVRL HAS ONLY 1 NORMAL EXIT                     *
                        2463 *        $CARPL - AFTER SET COMMAND EXECUTION          *
                        2464 *                                                      *
                        2465 *EXITS, ERROR -                                        *
                        2466 *        $CAERK - WITH ERROR CODES                     *
                        2467 *           @@E250 - VARIABLE NOT IN PROGRAM           *
                        2468 *           @@E251 - <ARITHMETIC CONSTANT> NOT IN RANGE 1E-99 < X < 1E99 *
                        2469 *           @@E252 - SUBSCRIPT EXCEEDS <ARRAY SIZE LIMIT> *
                        2470 *           @@E253 - ARRAY NOT IN PROGRAM              *
                        2471 *           @@E256 - INCONSISTENT NUMBER OF SUBSCRIPTS *
                        2472 *                                                      *
                        2473 *TABLES/WORK AREAS -                                   *
                        2474 *    * PRIMARY INPUT BUFFER                            *
                        2475 *    * SYMBOL AND ARRAY TABLES (SEE INPUT FOR FORMAT)  *
                        2476 *    * FUNCTION AND ARRAY TABLES (SEE INPUT FOR FORMAT) *
                        2477 *    * ALPHABETIC SYMBOL TABLE - 29 BYTES, CONTAINS EACH LETTER IN *
                        2478 *      THE ALPHABET PLUS $, #, @                       *
                        2479 *    * BUFFER 1 - 256 BYTES FOR VIRTUAL MEMORY BUFFER  *
                        2480 *    * BUFFER 2 - TO CONTAIN THE PAGING MODULE         *
                        2481 *                                                      *
                        2482 *ATTRIBUTES -                                          *
                        2483 *    NONE                                             *
                        2484 *                                                      *
                        2485 *CHARACTER CODE DEPENDENCY -                           *
                        2486 *    THE OPERATION OF THIS MODULE DEPENDS UPON THE FOLLOWING *
                        2487 *    PROPERTIES OF THE INTERNAL REPRESENTATION OF THE EXTERNAL *
                        2488 *    CHARACTER SET                                     *
                        2489 *        * MOST CODING HAS BEEN ARRANGED SO THAT REDEFINITION OF *
                        2490 *          CHARACTER CONSTANTS, BY REASSEBLY, WILL RESULT IN A CORRECT *
                        2491 *          MODULE FOR THE NEW DEFINITION              *
                        2492 *        * ALPHABETIC LETTERS A THROUGH Z ARE PRESUMED TO BE CODED IN *
                        2493 *          INCREASING COLLATING SEQUENCE, AND THE RANGE OF CHARACTER *
                        2494 *          CONSTANTS FOR THIS SERIES IS EXPECTED TO EXCLUDE ALL NUMERIC *
                        2495 *          CHARACTER CONSTANTS                        *
                        2496 *        * NUMERIC CHARACTERS 0 - 9 ARE PRESUMED TO BE CODED IN *
                        2497 *          INCREASING COLLATING SEQUENCE              *
                        2498 *        * EXTENDED ALPHABETIC LETTERS ($, #, @) ARE PRESUMEMED TO BE *
                        2499 *          IN INCREASING COLLATING SEQUENCE, AND ARE ALL EXPECTED TO *
                        2500 *          COLLATE LOWER THAN LETTER (A)              *
                        2501 *        * DECIMAL NUMBERS MUST BE CODED SO THAT THE LOW ORDER FOUR *
                        2502 *          BITS, WHEN CONSIDERED AS A BINARY INTEGER, IDENTIFY THE *
                        2503 *          VALUE OF THE DIGIT                         *
                        2504 *    THE SPECIFIC INSTRUCTIONS (INSTRUCTION SEQUENCES) WHICH REQUIRE *
                        2505 *    MODIFICATION IF THESE PROPERTIES OF THE CHARACTER SET ARE CHANGED *
                        2506 *    MAY BE IDENTIFIED BY -                            *
                        2507 *        * THE TABLE IDENTIFIED BY LABEL KSOATB        *
                        2508 *                                                      *
                        2509 *NOTES -                                               *
                        2510 *    ERROR PROCEDURES                                  *
                        2511 *        * ERROR CODE IS SET AT $CAERR                 *
                        2512 *        * @XR IS MODIFIED TO POINT TO CORE ADDRESS 0000 *
                        2513 *                                                      *
                        2514 *    REGISTER USAGE                                    *
                        2515 *        * BOTH REGISTERS ARE USED DURING PROGRAM EXECUTION *
```

```
                          2516 *      * THE REGISTERS ARE NOT SAVED OR RESTORED                    *
                          2517 *                                                                   *
                          2518 *    SAVED/RESTORED AREAS                                           *
                          2519 *      NONE                                                         *
                          2520 *                                                                   *
                          2521 *    MODIFICATION CONSIDERATIONS                                    *
                          2522 *      * KSOVRL OVERLAYS THE CORE I/O ROUTINES WITH THE SYMBOL TABLES *
                          2523 *        AND MUST BE RESTORED BY THE COMMAND ANALYSER ON EXIT       *
                          2524 *      * KSOVRL MUST NOT OVERLAY CORE FROM 1120 TO 1200.  THIS AREA *
                          2525 *        CONTAINS THE SET COMMON PARAMETER BLOCK PASSED FROM KSETIT *
                          2526 *      * THE PAGING ROUTINE MODULE IS LOADED FROM CORE PAGE 1200 UP. *
                          2527 *                                                                   *
                          2528 *    REQUIRED MODULES                                               *
                          2529 *      @SYSEQ - COMMON SYSTEM EQUATES                                *
                          2530 *      @FXDEQ - SYSTEM NUCLEUS ADDRESSES AND INDICATORS              *
                          2531 *      @CANEQ - SYSTEM LOCATION EQUATES                              *
                          2532 *      @ERMEQ - GENERAL ERROR MESSAGE EQUATES                        *
                          2533 *      $B$EQU - COMPILER FIXED EQUATES                               *
                          2534 *      $B@EQU - COMPILER SYSTEM EQUATES                              *
                          2535 *      $I$EQU - INTERPRETER FIXED EQUATES                            *
                          2536 *      SCANIT - COMMAND LINE DELIMITER SCAN ROUTINE                  *
                          2537 *      C4BIN2 - DECIMAL TO BINARY CONVERSION                         *
                          2538 *                                                                   *
                          2539 *    OTHER                                                          *
                          2540 *      NONE                                                         *
                          2541 ***********************************************************************
```

#KSOVR - SET KEYWORD COMMAND OVERLAY

```
                          2543 *         HDR   #KSOVR
                          2544 ********************************************************************
                          2545 *    PROGRAM HEADER FOR DISK LOAD                                  *
                          2546 ********************************************************************
              0AC8 2547 #$KSOV EQU   X'0AC8'                    DISK ADDR OF #KSOVR
              0C20 2548 #$$KSO EQU   X'0C20'                    CORE LOAD ADDRESS OF #KSOVR
              0005 2549 #$@KSO EQU   005                        SECTOR CNT OF #KSOVR
   0C20            2550        ORG   #$$KSO                     CORE LOAD ADDRESS
              0C20 2551 $$$$$$ EQU   *                          FIRST LOCATION IN PROGRAM
   0C20 7BD2E2D6E5D9  0C25 2552        DC    CL6'#KSOVR'        PROGRAM NAME
   0C26 35       0C26 2553        DC    IL1'053'                PROGRAM NUMBER OF #KSOVR
              0C27 2554 $KSOVR EQU   *                          ENTRY POINT TO PROGRAM
                          2555 *** END OF EXPANSION ***

   0C27            2557        ORG   X'0C27'
              0C27 2558 KSOVRL EQU   *                          CADDR KSOVRL
              1120 2559        USING KSOEQU,@BR                 SET BASE ADDR
   0C27 C2 01 1120  2560        LA    KSOEQU,@BR                LOAD KSOVRL BASE

                          2562 ********************************************************************
                          2563 *                                                                  *
                          2564 * SYMBOL TRANSLATION                                               *
                          2565 *                                                                  *
                          2566 ********************************************************************
                          2567 *
                          2568 * GET THE PAGING ROUTINE
                          2569 *
                          2570 *KSO190 DISK  KSODPL,WAIT
   0C2B C0 87 0025  2571 KSO190 B     $DISKN                    PERFORM PHYSICAL DISK OP
   0C2F 116C        0C30 2572        DC    AL2(KSODPL)          DPL ADDRESS
   0C31 C0 87 0025  2573        B     $DISKN                    WAIT AND CHECK DISK ERRORS
   0C35 057F        0C36 2574        DC    AL2($WAITF)          WAIT DPL ADDRESS
                          2575 *** END OF EXPANSION ***

   0C37 C0 87 1330  2577        B     I$LDXR                    GET THE FIRST PAGE OF THE
   0C3B FEFF        0C3C 2578        DC    AL(@VADDR)(KSO2LS)   * FUNCTION AND ARRAY THIS
   0C3D 2C 94 0B07 00  2579        MVC   KSO1PG(KSO1LN),0(,@XR) SHIFT TO TBL AREA
   0C42 C0 87 1330  2580        B     I$LDXR                    GET THE SECOND PAGE OF THE
   0C46 FFFF        0C47 2581        DC    AL(@VADDR)(KSOLST)   * FUNTION AND ARRAY TBL
   0C48 2C FF 0C07 00  2582        MVC   KSO2PG(B@LVPG),0(,@XR) SHIFT TO TBL AREA
                          2583 *
                          2584 * TEST PROGRAM PRECISION
                          2585 *
   0C4D 38 40 03D0  2586 KSO192 TBN   $XIND1,$XPREC             IS PREC LONG ?
   0C51 F2 90 0E    2587        JF    KSO195                    NO, PROCESS AS SHORT
                          2588 *
                          2589 * SET LONG PRECISION LENGTHS IN PRECISION SENSITIVE INSTRUCTIONS
                          2590 *
   0C54 3C F7 0FB8  2591 KSO194 MVI   KSO945+@Q,256-B@LILP      SET OVERFLOW FOR LONG PREC
   0C58 3C 08 0FF9  2592        MVI   KSO970+@DD2,B@LILP-1      SET FINAL MOVE INST TO
   0C5C 0C 01 0FF8 0FF9  2593        MVC   KSO970+@D1,KSO970+@DD2(2) * MOVE PACKED LONG VALUE
                          2594 *
                          2595 * RESTORE CORE ADDRESS OF FIRST CHARACTER
                          2596 *
   0C62 35 02 1185  2597 KSO195 L     KSOVAD,@XR                RESTORE PT
   0C66 6C 00 62 00  2598        MVC   KSOLSA(,@BR),KSOPD0(,@XR) SAVE LETTER
```

```
       0C6A C2 02 11A7        2599          LA    KSOATB,@XR              ADDR OF ALPHA TBL
                             2600 *
                             2601 * DETERMINE SYMBOL TABLE INDEX
                             2602 *
       0C6E 9D 00 00 62      2603 KSO200 CLC   KSOPD0(1,@XR),KSOLSA(,@BR)  DO LETTERS MATCH ?
       0C72 F2 81 0B         2604          JE    KSO210                  YES
       0C75 76 02 03         2605 KSO205 A     KSOI01(,@BR),@XR        INCR TO NEXT ALPHA TBL BYTE
       0C78 5E 00 67 05      2606          ALC   KSOATI(1,@BR),KSOI02(,@BR)  INCR INDEX BY 2.
       0C7C C0 87 0C6E       2607          B     KSO200                  LOOP UNTIL LETTERS MATCH
                             2608 *
                             2609 * TEST FOR LETTER-DIGIT REFERENCE
                             2610 *
       0C80 35 02 1185       2611 KSO210 L     KSOVAD,@XR              RESTORE PT
       0C84 76 02 03         2612          A     KSOI01(,@BR),@XR        INCR TO NEXT BYTE
       0C87 BD F0 00         2613 KSO215 CLI   KSOPD0(,@XR),B@DEC0     IS BYTE A DIGIT ?
       0C8A F2 02 5E         2614          JNL   KSO270                  YES, PROCESS LETTER-DIGIT VAR
                             2615 *
                             2616 * TEST FOR A CHARACTER VARIABLE OR ARRAY REFERENCE
                             2617 *
       0C8D BD 5B 00         2618 KSO220 CLI   KSOPD0(,@XR),B@CVAR     A CHAR DESIGNATOR ?
       0C90 F2 01 03         2619          JNE   KSO225                  NO, TEST FOR ARITH ARRAY
       0C93 76 02 03         2620          A     KSOI01(,@BR),@XR        INCR TO NEXT BYTE
                             2621 *
                             2622 * TEST FOR AN ARRAY REFERENCE
                             2623 *
       0C96 BD 4D 00         2624 KSO225 CLI   KSOPD0(,@XR),B@LPAR     A LEFT PAREN ?
       0C99 F2 81 28         2625          JE    KSO250                  YES, PROCESS ARRAY VAR
                             2626 *
                             2627 * TEST IF VARIABLE IS REFERENCED IN THE SYMBOL TABLE
                             2628 *
       0C9C 7D 01 5E         2629 KSO230 CLI   KSOCRS(,@BR),KSOONN     A CHAR REF ?
       0C9F C2 01 070C       2630          LA    KSOLVT,@BR              CADDR LETTER VAR TBL
       0CA3 F2 01 04         2631          JNE   KSO240                  NO, ADD DISP
       0CA6 C2 01 098A       2632          LA    KSOCVT,@BR              CADDR CHAR VAR TBL
       0CAA 36 01 1187       2633 KSO240 A     KSOATI,@BR              INCR TO PROPER ENTRY
       0CAE 1D 01 1121 01    2634          CLC   KSOI00,KSOPD1(@VADDR,@BR) IS ENTRY NULL ?
       0CB3 C0 81 100E       2635          BE    KSO991                  YES, EXIT TO ERROR RTN
       0CB7 1C 01 1185 01    2636          MVC   KSOVAD,KSOPD1(@VADDR,@BR) SET VADDR IN PUT PARAM
       0CBC C2 01 1120       2637          LA    KSOEQU,@BR              RESTORE BASE
       0CC0 C0 87 0DE7       2638          B     KSO360                  TO VM MODIFICATION
                             2639 *
                             2640 * TEST IF ARRAY VARIABTF IS REFERENCED IN THE SYMBOL TABLE
                             2641 *
       0CC4 7D 01 5E         2642 KSO250 CLI   KSOCRS(,@BR),KSOONN     A CHAR REF
       0CC7 C2 01 09C4       2643          LA    KSONAT,@BR              CADDR NAT
       0CCB F2 01 04         2644          JNE   KSO260                  NO, ADD DISP
       0CCE C2 01 09FE       2645          LA    KSOCAT,@BR              CADDR CAT
       0CD2 36 01 1187       2646 KSO260 A     KSOATI,@BR              ADD DISP TO INDEX PROPER ENTRY
       0CD6 1D 01 1121 01    2647          CLC   KSOI00,KSOPD1(@VADDR,@BR) IS ENTRY NULL ?
       0CDB C0 81 1023       2648          BE    KSO994                  YES, EXIT TO ERROR RTN
       0CDF 1C 01 1185 01    2649          MVC   KSOVAD,KSOPD1(@VADDR,@BR) SET VADDR IN PUT PARAM
       0CE4 C2 01 1120       2650          LA    KSOEQU,@BR              RESTORE BASE
       0CE8 F2 87 2B         2651          J     KSO280                  TO SUBSC SCAN
                             2652 *
                             2653 * DETERMINE THE LETTER-DIGIT TABLE INDEX
                             2654 *
```

```
     #KSOVR - SET KEYWORD COMMAND OVERLAY
```

```
    0CEB 68 03 69 00           2655 KSO270 MNN    KSODGT(,@BR),KSOPD0(,@XR) CONVERT DIGIT TO BINARY
    0CEF 5E 00 69 69           2656        ALC    KSODGT(,@BR),KSODGT(1,@BR)  DOUBLE THE BINARY DIGIT
    0CF3 5E 01 67 67           2657        ALC    KSOATI(,@BR),KSOATI(KSOLAC,@BR)  DOUBLE THE INDEX
    0CF7 C2 01 0746            2658        LA     KSOLDT,@BR                  CADDR LETTER-DIGIT TBL
    0CFB 36 01 1187            2659        A      KSOATI,@BR                  ADD 10 TIMES THE SYMBOL INDEX
    0CFF 0E 01 1187 1187       2660        ALC    KSOATI,KSOATI(KSOLAC)       * AND TWICE THE DIGIT VALUE
    0D05 0E 01 1187 1187       2661        ALC    KSOATI,KSOATI(KSOLAC)       * TO THE BASE ADDR TO INDEX
    0D0B 36 01 1189            2662        A      KSODGT,@BR                  * THE PROPER TABLE ENTRY
    0D0F E2 02 01              2663        LA     1(,@XR),@XR                 INCR PT
    0D12 C0 87 0CAA            2664        B      KSO240                      TEST FOR NULL ENTRY
```

```
                              2666 ***********************************************************************
                              2667 *                                                                     *
                              2668 * SUBSCRIPT PROCESSING                                                *
                              2669 *                                                                     *
                              2670 ***********************************************************************
                              2671 *
                              2672 * PREPARE RTN IF REF IS A CHARACTER ARRAY
                              2673 *
    0D16 7D 01 5E            2674 KSO280 CLI   KSOCRS(,@BR),KSOONN        A CHAR REF ?
    0D19 F2 81 06            2675        JE    KSO286                     YES, GO SET TERM SW ON
    0D1C 7C 00 5F            2676 KSO284 MVI   KSOSTS(,@BR),KSOOFF        SET SUBSC TERM SW OFF
    0D1F F2 87 03            2677        J     KSO290                     INCR TO NEXT BYTE
    0D22 7C 01 5F            2678 KSO286 MVI   KSOSTS(,@BR),KSOONN        SET SUBSC TERM SW ON
                              2679 *
                              2680 * SKIP LEADING ZEROS
                              2681 *
    0D25 76 02 03            2682 KSO290 A     KSOI01(,@BR),@XR           INCR TO NEXT BYTE
    0D28 BD F0 00            2683        CLI   KSOPD0(,@XR),B@DEC0        IS BYTE A ZERO
    0D2B C0 81 0D25          2684        BE    KSO290                     YES, TEST NEXT BYTE
                              2685 *
                              2686 * TEST FOR A VALID SUBSCRIPT
                              2687 *
    0D2F BD F1 00            2688 KSO295 CLI   KSOPD0(,@XR),B@DEC1        IS BYTE A DIGIT
    0D32 C0 82 101C          2689        BL    KSO993                     NO, EXTT TO ERROR RTN
                              2690 *
                              2691 * CONVERT SUBSCRIPT TO BINARY
                              2692 *
    0D36 C0 87 102F          2693 KSO300 B     C4BIN2                     CONVERT TO BINARY
    0D3A 5C 01 6B 6D         2694        MVC   KSOSC1(,@BR),KSOSC2(B@LDMN,@BR)  SHIFT SUBSC 2 TO SUBSC 1
    0D3E 4C 01 6D 1099       2695        MVC   KSOSC2(B@LDMN,@BR),C4BVAL SAVE 2ND SUBSCRIPT
                              2696 *
                              2697 * TEST FOR   END OF SUBSCRIPT
                              2698 *
    0D43 BD 5D 00            2699 KSO310 CLI   KSOPD0(,@XR),B@RPAR        AT END OF SUBSC ?
    0D46 F2 81 12            2700        JE    KSO320                     YES, TEST VALIDITY OF SUBSC
    0D49 7D 01 5F            2701        CLI   KSOSTS(,@BR),KSOONN        IS SUBSC TERM SW ON
    0D4C C0 81 101C          2702        BE    KSO993                     YES, EXIT TO ERROR RTN
    0D50 BD 6B 00            2703        CLI   KSOPD0(,@XR),B@CMMA        IS BYTE A COMMA
    0D53 C0 81 0D22          2704        BE    KSO286                     YES, RECYCLE SUBSC LOOP
    0D57 C0 87 101C          2705        B     KSO993                     NO, EXIT TO ERROR RTN
                              2706 *
                              2707 * TEST THE SUBSCRIPTS OF ARITHMETIC ARRAY FOR VALIDITY
                              2708 *
    0D5B 34 02 0DE3          2709 KSO320 ST    KSO350+@OP1,@XR            SAVE PT
    0D5F 5F 01 6F 6F         2710        SLC   KSOCNT(,@BR),KSOCNT(KSOLAC,@BR) CLEAR ACCUM
    0D63 75 02 65            2711        L     KSOVAD(,@BR),@XR           DOPE VECTOR VADDR
    0D66 76 02 0B            2712        A     KSOAAC(,@BR),@XR           CONVERT VADDR TO CADDR
                              2713 *
                              2714 * TEST FOR ARRAY SUBSCRIPT COMPATABILITY
                              2715 *
    0D69 7D 01 5E            2716        CLI   KSOCRS(,@BR),KSOONN        IS REF A CHAR REF ?
    0D6C F2 81 52            2717        JE    KSO340                     YES, TEST SUBSC FOR VALIDITY
    0D6F 5D 01 6B 01         2718        CLC   KSOSC1(B@LBIN,@BR),KSOI00(,@BR)  INPUT ARRAY A VECTOR ?
    0D73 F2 01 0A            2719        JNE   KSO321                     NO, TEST IF D/V IS A VECTOR
    0D76 9D 01 01 01         2720        CLC   B@ACD1(B@LBIN,@XR),KSOI00(,@BR)  IS D/V A VECTOR REF ?
    0D7A F2 81 12            2721        JE    KSO323                     YES, CHECK SUBSC VALIDITY
```

```
    0D7D F2 87 07              2722        J     KSO322                NO, SET ERROR CODE

    0D80 9D 01 01 01          2724 KSO321 CLC   B@ACD1(B@LBIN,@XR),KSOI00(,@BR) IS D/V A MATRIX
    0D84 F2 01 08             2725        JNE   KSO323                YES, CONTINUE PROC
    0D87 3C 42 03CD           2726 KSO322 MVI   $CAERR,@@E256         SET ERROR CODE
    0D8B C0 87 1027           2727        B     KSO995                TO ERROR RTN
    0D8F 6C 01 65 07          2728 KSO323 MVC   KSOVAD(,@BR),B@ABAS(B@LDMN,@XR)  SET ARRAY BASE VADDR
    0D93 6D 01 6B 01          2729 KSO324 CLC   KSOSC1(B@LDMN,@BR),B@ACD1(,@XR)  IS SUBSC 1 VALID
    0D97 C0 84 101C           2730        BH    KSO993                NO, EXIT TO ERROR RTN
    0D9B 6D 01 6D 03          2731 KSO326 CLC   KSOSC2(B@LDMN,@BR),B@ACD2(,@XR)  IS SUBSC 2 VALID
    0D9F C0 84 101C           2732        BH    KSO993                NO, EXIT TO ERROR RTN
                              2733 *
                              2734 * DETERMINE ELEMENT DISPLACEMENT FROM ARRAY BASE ADDR
                              2735 *
    0DA3 5F 01 6B 03          2736 KSO330 SLC   KSOSC1(,@BR),KSOI01(B@LDMN,@BR) DECR SUBSC 1 BY 1
    0DA7 F2 04 08             2737        JNH   KSO335                UNDERFLOW, CONTINUE PROC
    0DAA 6E 01 6D 03          2738        ALC   KSOSC2(,@BR),B@ACD2(B@LDMN,@XR)  MULTIPLY SUBSC 1 BY THE
    0DAE C0 87 0DA3           2739        B     KSO330                     * 2ND DIMENSION
    0DB2 5E 01 6F 6D          2740 KSO335 ALC   KSOCNT(,@BR),KSOSC2(B@LDMN,@BR)  MULTIPLY ELEMENT COUNT
    0DB6 5F 00 70 03          2741        SLC   KSOELL(1,@BR),KSOI01(,@BR)  * BY THE LENGTH IN BYTES
    0DBA C0 01 0DB2           2742        BNE   KSO335                     * OF THE ELEMENT
                              2743 *
                              2744 * INCREMENT ARRAY BASE ADDRESS, SET PUT RTN PARAMETER
                              2745 *
    0DBE F2 87 1B             2746 KSO337 J     KSO347                TO SET ELEMENT ADDR
                              2747 *
                              2748 * TEST FOR CHARACTER ARRAY SUBSCRIPT VALIDITY
                              2749 *
    0DC1 6D 01 6D 01          2750 KSO340 CLC   KSOSC2(B@LDMN,@BR),B@CDMN(,@XR)  IS SUBSC VALID ?
    0DC5 C0 84 101C           2751        BH    KSO993                NO, EXIT TO ERROR RTN
    0DC9 7C 13 70             2752        MVI   KSOELL(,@BR),B@LCRV      SET LNG OF ELEMENT
    0DCC 6C 01 65 03          2753        MVC   KSOVAD(,@BR),B@CBAS(B@LDMN,@XR) SET ARRAY BASE VADDR
                              2754 *
                              2755 * DETERMINE ELEMENT DISPLACEMENT      FROM ARRAY BASE ADDR
                              2756 *
    0DD0 5E 01 6F 6D          2757 KSO345 ALC   KSOCNT(,@BR),KSOSC2(B@LDMN,@BR)  MULTIPLY SUBSC 2 BY
    0DD4 5F 00 70 03          2758        SLC   KSOELL(1,@BR),KSOI01(,@BR) * ELEMENT LENGTH TO GET
    0DD8 C0 01 0DD0           2759        BNE   KSO345                     * THE DISP
                              2760 *
                              2761 * INCREMENT THE ARRAY BASE ADDRESS, SET PUT RTN PARAMETER
                              2762 *
    0DDC 5E 01 65 6F          2763 KSO347 ALC   KSOVAD(@VADDR,@BR),KSOCNT(,@BR)  SET ELEMENT ADDR
                              2764 *
                              2765 * INCREMENT LINE POINTER PAST RIGHT PAREN
                              2766 *
    0DE0 C2 02 0000           2767 KSO350 LA    *-*,@XR                RESTORE LINE PT
    0DE4 76 02 03             2768        A     KSOI01(,@BR),@XR       INCR LINE PT
                              2769 *
                              2770 *  ADVANCE LINE POINTER TO THE 1ST CHARACTER OF THE CONSTANT
                              2771 *
    0DE7 C0 87 109F           2772 KSO360 B     SCANIT                 TO THE ? SIGN
    0DEB 76 02 03             2773        A     KSOI01(,@BR),@XR       INCR PAST THE = SIGN
    0DEE C0 87 109F           2774        B     SCANIT                 TO 1ST CON CHAR
                              2775 *
                              2776 * BUILD THE NECESSARY CONSTANT IN THE CONSTANT BUCKET
                              2777 *
```

```
    0DF2 7D 01 5E               2778 KSO370 CLI    KSOCRS(,@BR),KSOONN      A CHAR REF ?
    0DF5 C0 81 0F4C             2779        BE     KSO600                   BUILD CHAR CON
    0DF9 7D 01 60               2780 KSO375 CLI    KSOICS(,@BR),KSOONN      IS SW ON ?
    0DFC C0 81 0F99             2781        BE     KSO900                   MOVE CON TO VM

                                2783 ***********************************************************************
                                2784 *                                                                     *
                                2785 * ARITHMETIC CONVERSION ROUTINE                                       *
                                2786 *                                                                     *
                                2787 ***********************************************************************
                                2788 *
                                2789 * INITIALIZE THE STATUS AND EXPONENT BYTES IN THE BUCKET
                                2790 *
    0E00 7C 00 74               2791 KSO400 MVI    KSOCB0(,@BR),@ZERO       SET STATUS TO POSITIVE
    0E03 7C 80 75               2792        MVI    KSOCB1(,@BR),B@NXZR      SET EXPONENT TO 0
    0E06 7C F0 86               2793        MVI    KSOCHR(,@BR),B@DEC0      SET MANTISSA TO DECIMAL
    0E09 5C 0F 85 86            2794        MVC    KSOCHR-1(B@LELP,@BR),KSOCHR(,@BR)  * ZEROS
                                2795 *
                                2796 * TEST SIGN OF VALUE AND SET CONSTANT BUCKET EQUAL TO IT
                                2797 *
    0E0D BD 4E 00               2798 KSO410 CLI    KSOPD0(,@XR),B@PLUS      IS SIGN POSITIVE ?
    0E10 F2 81 09               2799        JE     KSO420                   YES, GO ELIMINATE LEADING ZEROS
    0E13 BD 60 00               2800        CLI    KSOPD0(,@XR),B@MINS      IS SIGN NEGATIVE ?
    0E16 F2 01 06               2801        JNE    KSO430                   NO, GO ELIMINATE LEADING ZEROS
    0E19 7A 10 74               2802        SBN    KSOCB0(,@BR),B@SIGN      SET STATUS BYTE NEGATIVE
                                2803 *
                                2804 * ELIMINATE LEADING ZEROS
                                2805 *
    0E1C 76 02 03               2806 KSO420 A      KSOI01(,@BR),@XR         INCR PT TO NEXT BYTE
    0E1F BD F0 00               2807 KSO430 CLI    KSOPD0(,@XR),B@DEC0      IS BYTE A ZERO
    0E22 C0 81 0E1C             2808        BE     KSO420                   YES, INCR PAST IT
                                2809 *
                                2810 * TEST FOR DECIMAL POINT
                                2811 *
    0E26 D2 01 76               2812 KSO440 LA     KSOCB2(,@BR),@BR         PT TO 1ST MANTISSA BYTE
    0E29 BD 4B 00               2813        CLI    KSOPD0(,@XR),B@DPNT      IS BYTE A DECIMAL POINT ?
    0E2C F2 01 1B               2814        JNE    KSO470                   NO, TEST IF A DIGIT
                                2815 *
                                2816 * DETERMINE NEGATIVE EXPONENT
                                2817 *
    0E2F E2 02 01               2818 KSO450 LA     1(,@XR),@XR              INCR PT
    0E32 BD F0 00               2819        CLI    KSOPD0(,@XR),B@DEC0      IS BYTE A ZERO
    0E35 F2 01 0E               2820        JNE    KSO460                   NO, GO SET FRACTION SW ON
    0E38 0F 00 1195 1123        2821        SLC    KSOCB1,KSOI01(1)         DECR LXPONENT
    0E3E C0 84 0E2F             2822        BH     KSO450                   NO UNDERFLOW,REPEAT LOOP
    0E42 C0 87 1015             2823        B      KSO992                   UNDERFLOW, EXIT TO ERROR RTN
                                2824 *
                                2825 * TEST FOR A DIGIT
                                2826 *
    0E46 3C 01 1192             2827 KSO460 MVI    KSOMFS,KSOONN            SET FRACTION SW ON
    0E4A BD F0 00               2828 KSO470 CLI    KSOPD0(,@XR),B@DEC0      IS BYTE A DIGIT ?
    0E4D F2 82 46               2829        JL     KSO530                   NO, TEST FOR EXPONENT
                                2830 *
                                2831 * DETERMINE POSITIVE EXPONENT
                                2832 *
    0E50 3D 01 1192             2833 KSO480 CLI    KSOMFS,KSOONN            IS FRACTION SW ON ?
```

```
     0E54 F2 81 0D          2834        JE     KSO490                   YES, TEST BUCKET SW
     0E57 0E 00 1195 1123   2835        ALC    KSOCB1,KSOI01(1)         INCR EXPONENT
     0E5D F2 82 04          2836        JL     KSO490                   NO OVERFLOW, TEST BUCKET SW
     0E60 C0 87 1015        2837        B      KSO992                   OVERFLOW, EXIT TO ERROR RTN
                            2838 *
                            2839 * MOVE DIGIT TO MANTISSA AND TEST FOR COMPLETION
                            2840 *
     0E64 3D 01 1193        2841 KSO490 CLI    KSOCBS,KSOONN            IS THE BUCKET SW ON
     0E68 F2 81 14          2842        JE     KSO510                   YES, INCR TO NEXT BYTE
     0E6B 6C 00 00 00       2843        MVC    KSOPD0(1,@BR),KSOPD0(,@XR)  MOVE DICIT TO MANTISSA
     0E6F D2 01 01          2844        LA     1(,@BR),@BR              INCR PT
     0E72 0F 00 1191 1123   2845 KSO500 SLC    KSOEND,KSOI01           DECR MANTISSA COUNT,
     0E78 F2 84 04          2846        JH     KSO510                   NO UNDERFLUW, INCR TO NEXT BYTE
     0E7B 3C 01 1193        2847        MVI    KSOCBS,KSOONN           SET BUCKET SA ON
     0E7F E2 02 01          2848 KSO510 LA     1(,@XR),@XR             INCR PT
     0E82 BD F0 00          2849        CLI    KSOPD0(,@XR),B@DEC0     IS BYTE A DIGIT ?
     0E85 C0 02 0E50        2850        BNL    KSO480                   YES, TEST FRACTION SW
     0E89 BD 4B 00          2851 KSO520 CLI    KSOPD0(,@XR),B@DPNT     IS BYTE A DECIMAL POINT ?
     0E8C F2 01 07          2852        JNE    KSO530                   NO, TEST FOR AN EXPONENT
     0E8F E2 02 01          2853        LA     1(,@XR),@XR             INCR PT
     0E92 C0 87 0E46        2854        B      KSO460                   TEST FOR DIGIT
                            2855 *
                            2856 * TEST FOR POSSIBLE EXPONENT
                            2857 *
     0E96 BD C5 00          2858 KSO530 CLI    KSOPD0(,@XR),B@EXPC     VALUE HAVE A EXPONENT ?
     0E99 F2 01 48          2859        JNE    KSO590                   NO, TEST FOR A ZERO VALUE
     0E9C E2 02 01          2860        LA     1(,@XR),@XR             INCR PT
                            2861 *
                            2862 * DETERMINE SIGN OF EXPONENT
                            2863 *
     0E9F BD 4E 00          2864 KSO540 CLI    KSOPD0(,@XR),B@PLUS     IS SIGN POSITIVE ?
     0EA2 F2 81 0A          2865        JE     KSO550                   YES, INCR PAST SIGN
     0EA5 BD 60 00          2866        CLI    KSOPD0(,@XR),B@MINS     IS SIGN NEGATIVE ?
     0EA8 F2 01 07          2867        JNE    KSO560                   NO, GO CONVERT EXPONENT
     0EAB 3C 01 0EB7        2868        MVI    KSO570+@Q,KSOONN        SET NEGATIVE ADJUSTMENT SW ON
     0EAF E2 02 01          2869 KSO550 LA     1(,@XR),@XR             INCR PT
                            2870 *
                            2871 * CONVERT EXPONENT TO BINARY
                            2872 *
     0EB2 C0 87 102F        2873 KSO560 B      C4BIN2                   CONVERT TO BINARY
                            2874 *
                            2875 * ADJUST BUCKET EXPONENT WITH BINARY EXPONENT
                            2876 *
     0EB6 3D 00 1121        2877 KSO570 CLI    KSOI00,*-*              IF THE ADJUSTMENT SWITCH IS
     0EB6                   2878        ORG    KSO570,@Q              * ON THEN SUBTRACK THE BINARY
     0EB6 00           0EB6 2879        DC     XL1'00'               * EXPONENT FROM THE BUCKET
     0EBA                   2880        ORG
     0EBA F2 01 0D          2881        JNE    KSO575                * EXPONENT
     0EBD 0E 00 1195 1099   2882        ALC    KSOCB1,C4BVAL(1)        INCR THE EXPONENT
     0EC3 C0 02 1015        2883        BNL    KSO992                   OVERFLOW, EXIT TO ERROR RTN
     0EC7 F2 87 0A          2884        J      KSO580                   GO TEST FOR EXPONENT VALIDITY
     0ECA 0F 00 1195 1099   2885 KSO575 SLC    KSOCB1,C4BVAL(1)        DECR THE EXPONENT
     0ED0 C0 04 1015        2886        BNH    KSO992                   UNDERFLOW, EXIT TO ERROR RTN
                            2887 *
                            2888 * TEST FOR EXPONENT VALIDITY
                            2889 *
```

```
        #KSOVR - SET KEYWORD COMMAND OVERLAY
```

```
     0ED4 3D 1E 1195          2890 KSO580 CLI    KSOCB1,B@NXLO              LESS THAN THE MINIMUM EXPONENT ?
     0ED8 C0 82 1015          2891        BL     KSO992                    YES, EXIT TO ERROR RTN
     0EDC 3D E3 1195          2892        CLI    KSOCB1,B@NXHI             GREATER THAN THE MAXIMUM ?
     0EE0 C0 84 1015          2893        BH     KSO992                    YES, EXIT TO ERROR RTN
                              2894 *
                              2895 * TEST FOR A VALUE OF ZERO
                              2896 *
     0EE4 3D F0 1196          2897 KSO590 CLI    KSOCB2,B@DEC0             IS 1ST MANTISSA BYTE ZERO ?
     0EE8 F2 01 08            2898        JNE    KSO592                    NO, PACK MANTISSA
     0EEB 3C 00 1194          2899        MVI    KSOCB0,KSOE00             SET STATUS TO POSITIVE
     0EEF 3C 1E 1195          2900        MVI    KSOCB1,B@NXLO             SET EXPONENT TO MAX LOW.
                              2901 *
                              2902 * PACK THE VALUE
                              2903 *
     0EF3 C2 01 1120          2904 KSO592 LA     KSOEQU,@BR                RESTORE BASE
     0EF7 5C 00 63 75         2905        MVC    KSOEXP(,@BR),KSOCB1(1,@BR)  SAVE EXP
     0EFB 58 03 74 76         2906        MNN    KSOCB0(,@BR),KSOCB2(,@BR) SHIFT 1ST NUMBER
                              2907 *
                              2908 * PACK SHORT PRECISION MANTISSA
                              2909 *
     0EFF 58 01 75 77         2910 KSO594 MZN    KSOCB1(,@BR),KSOCB3(,@BR) PACK MANTISSA NUMBER
     0F03 58 03 75 78         2911        MNN    KSOCB1(,@BR),KSOCB4(,@BR) PACK MANTISSA NUMBER
     0F07 58 01 76 79         2912        MZN    KSOCB2(,@BR),KSOCB5(,@BR) PACK MANTISSA NUMBER
     0F0B 58 03 76 7A         2913        MNN    KSOCB2(,@BR),KSOCB6(,@BR) PACK MANTISSA NUMBER
     0F0F 58 01 77 7B         2914        MZN    KSOCB3(,@BR),KSOCB7(,@BR) PACK MANTISSA NUMBER
     0F13 58 03 77 7C         2915        MNN    KSOCB3(,@BR),KSOCB8(,@BR) PACK MANTISSA NUMBER
     0F17 38 40 03D0          2916 KSO595 TBN    $XIND1,$XPREC             IS PREC LONG ?
     0F1B F2 90 27            2917        JF     KSO597                    NO, SHIFT SHORT PREC EXP
                              2918 *
                              2919 * COMPLETE LONG PRECISION PACKING
                              2920 *
     0F1E 58 01 78 7D         2921 KSO596 MZN    KSOCB4(,@BR),KSOCB9(,@BR) PACK MANTISSA NUMBER
     0F22 58 03 78 7E         2922        MNN    KSOCB4(,@BR),KSOCBA(,@BR) PACK MANTISSA NUMBER
     0F26 58 01 79 7F         2923        MZN    KSOCB5(,@BR),KSOCBB(,@BR) PACK MANTISSA NUMBER
     0F2A 58 03 79 80         2924        MNN    KSOCB5(,@BR),KSOCBC(,@BR) PACK MANTISSA NUMBER
     0F2E 58 01 7A 81         2925        MZN    KSOCB6(,@BR),KSOCBD(,@BR) PACK MANTISSA NUMBER
     0F32 58 03 7A 82         2926        MNN    KSOCB6(,@BR),KSOCBE(,@BR) PACK MANTISSA NUMBER
     0F36 58 01 7B 83         2927        MZN    KSOCB7(,@BR),KSOCBF(,@BR) PACK MANTISSA NUMBER
     0F3A 58 03 7B 84         2928        MNN    KSOCB7(,@BR),KSOCBG(,@BR) PACK MANTISSA NUMBER
     0F3E 5C 00 7C 63         2929        MVC    KSOCB8(,@BR),KSOEXP(1,@BR)  RESTORE EXPO LONG PREC)
     0F42 F2 87 54            2930        J      KSO900                    MOVE TO VM
     0F45 5C 00 78 63         2931 KSO597 MVC    KSOCB4(,@BR),KSOEXP(1,@BR)  RESTORE EXP (SHORT PREC
                              2932 *
                              2933 *  MOVE VALUE TO VM
                              2934 *
     0F49 F2 87 4D            2935 KSO598 J      KSO900                    PUT CON TO VM

                              2937 ***********************************************************************
                              2938 *                                                                    *
                              2939 *  CHARACTER CONVERSION ROUTINE                                       *
                              2940 *                                                                    *
                              2941 ***********************************************************************
                              2942 *
                              2943 *  INITIALIZE THE CHARACTER ROUTINE
                              2944 *
     0F4C 7C 40 74            2945 KSO600 MVI    KSOCB0(,@BR),B@DTYP       SET CHAR STATUS BYTE
```

```
    0F4F 7C 40 86                2946 KSO610 MVI   KSOCHR(,@BR),B@BLNK       SET DATA CONSTANT BUCKET TO
    0F52 5C 0F 85 86             2947        MVC   KSOCHR-1(B@LELP,@BR),KSOCHR(,@BR)  * BLANKS
    0F56 7C 12 71               2948        MVI   KSOEND(,@BR),B@LCRV-1     SET MANTISSA COUNT
    0F59 D2 01 75               2949 KSO620 LA    KSOCB1(,@BR),@BR         SET BUCKET PT
                                2950 *
    0F5C E2 02 01               2951        LA    1(,@XR),@XR              INCR PT
    0F5F BD 7D 00               2952 KSO630 CLI   KSOPD0(,@XR),B@SQUO      AT DELIMITER ?
    0F62 F2 01 09               2953        JNE   KSO650                   NO, MOVE CHAR TO BUCKET
    0F65 E2 02 01               2954        LA    1(,@XR),@XR              INCR PT
    0F68 BD 7D 00               2955 KSO640 CLI   KSOPD0(,@XR),B@SQUO      A DELIMITER ?
    0F6B F2 01 1A               2956        JNE   KSO670                   NO, EXIT RTN
                                2957 *
                                2958 * MOVE CHARACTER FROM LINE TO BUCKET
                                2959 *
    0F6E 6C 00 00 00            2960 KSO650 MVC   KSOPD0(1,@BR),KSOPD0(,@XR)  MOVE CHAR TO BUCKET
    0F72 E2 02 01               2961        LA    1(,@XR),@XR              INCR PT
    0F75 D2 01 01               2962        LA    1(,@BR),@BR              INCR PT
    0F78 0E 00 1194 1123        2963        ALC   KSOCB0,KSOI01(1)         INCR BYTE COUNT IN STATUS BYTE
                                2964 *
                                2965 * TEST FOR FULL BUCKET
                                2966 *
    0F7E 0F 00 1191 1123        2967 KSO660 SLC   KSOEND,KSOI01            IS BUCKET FULL
    0F84 C0 84 0F5F             2968        BH    KSO630                   NO, INCR TO NEXT BYTE
                                2969 *
                                2970 * SET PUT ROUTINE PARAMETERS
                                2971 *
    0F88 3C ED 0FB8             2972 KSO670 MVI   KSO945+@Q,256-B@LCRV     PREPARE THE PUT ROUTINE
    0F8C 3C 12 0FF9             2973        MVI   KSO970+@DD2,B@LCRV-1     * WITH CHARACTER LERCH
    0F90 0C 01 0FF8 0FF9        2974        MVC   KSO970+@D1,KSO970+@DD2(2) * VALUES
                                2975 *
                                2976 * MOVE VALUE TO VM
                                2977 *
    0F96 F2 87 00               2978 KSO680 J     KSO900                   PUT VALUE TO VM
```

```
                                2980 ****************************************************************************
                                2981 *                                                                          *
                                2982 * KSETIT VIRTUAL MEMORY PUT ROUTINE                                        *
                                2983 *                                                                          *
                                2984 ****************************************************************************
                                2985 *
                                2986 * SET IOCR PARAMETER WITH DATA ELEMENT PAGE
                                2987 *
   0F99 C2 01 1120             2988 KSO900 LA    KSOEQU,@BR                  SET BASE
   0F9D 1C 01 144A 65          2989        MVC   I$VADR,KSOVAD(@CADDR,@BR) SET VADDR OF VALUE
                                2990 *
                                2991 * READ DATA ELEMENT PAGE
                                2992 *
   0FA2 C0 87 1349             2993 KSO910 B     I$MDFY                      GET PG VALUE IS IN
                                2994 *
                                2995 * SET POINTERS TO OUTPUT AREA AND VALUE
                                2996 *
   0FA6 35 01 144C             2997        L     I$CADR,@BR                  1ST BYTE IN PG
   0FAA C2 02 1194             2998 KSO930 LA    KSOCDB,@XR                  CADDR CONSTANT BUCKET
                                2999 *
                                3000 * PRESERVE STATUS BITS OF VM VALUE
                                3001 *
   0FAE 78 80 00               3002 KSO940 TBN   KSOPD0(,@BR),B@TRAC         IS VM VAL TRACE BIT ON
   0FB1 F2 90 03               3003        JF    KSO945                      NO, MOVE VAL TO VM
   0FB4 BA 80 00               3004        SBN   KSOPD0(,@XR),B@TRAC         YES, SET BUCKET TRACE BIT ON
                                3005 *
                                3006 * TEST FOR PG BOUNDARY OVERFLOW
                                3007 *
   0FB7 3D 00 144A             3008 KSO945 CLI   I$VADR,*-*                  TEST IF ENOUGH BYTES REMAIN
   0FB8                        3009        ORG   KSO945+@Q                   * IN THE PG TO CONTAIN THE
   0FB8 FB            0FB8     3010        DC    AL1(256-B@LISP)             * BUCKET VALUE
   0FBB                        3011        ORG
   0FBB F2 04 38               3012        JNH   KSO970                      YES, MOVE VALUE TO VM
                                3013 *
                                3014 * CALCULATE INSTRUCTION TO MOVE 1ST SEGMENT OF VALUE AND MOVE IT
                                3015 *
   0FBE 3C FF 0FD1             3016 KSO950 MVI   KSO955+@DD2,KSOPGB          CALCULATE BYTES TO MOVE
   0FC2 0F 00 0FD1 144A        3017        SLC   KSO955+@DD2,I$VADR(1)       * IN THE FIRST
   0FC8 0C 01 0FD0 0FD1        3018        MVC   KSO955+@D1,KSO955+@DD2(2)   * SEGMENT
   0FCE 6C 00 00 00            3019 KSO955 MVC   *-*(,@BR),*-*(@Q,@XR)       MOVE BUCKET VALUE TO BFR
                                3020 *
                                3021 * CALCULATE REMAINING SEGMENT MOVE INSTRUCTION
                                3022 *
   0FD2 0F 00 0FF7 0FCF        3023 KSO960 SLC   KSO970+@Q,KSO955+@Q(1)      SET LNG TO MOVE
   0FD8 0F 00 0FF7 1123        3024        SLC   KSO970+@Q,KSOI01(1)         DECR BY 1
   0FDE 0C 00 0FF8 0FF7        3025        MVC   KSO970+@D1(1),KSO970+@Q     SET DISP TO LAST BYTE OF VALUE
                                3026 *
                                3027 * ACCESS NEXT CONTIGOUS PAGE
                                3028 *
   0FE4 0E 00 1449 1123        3029 KSO965 ALC   I$VADR-1,KSOI01(1)          INCR PG NO.
   0FEA 3C 00 144A             3030        MVI   I$VADR,KSOOFF               SET PG DISP TO ZERO
   0FEE C0 87 1349             3031        B     I$MDFY                      GET NEXT PG
   0FF2 35 01 144C             3032        L     I$CADR,@BR                  RESET BFR ADDR
                                3033 *
                                3034 * MOVE FINAL SEGMENT TO BUFFER
                                3035 *
```

```
      #KSOVR - SET KEYWORD COMMAND OVERLAY


ERR LOC   OBJECT CODE       ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  10/02/22  PAGE  16

    0FF6 6C 00 00 00             3036 KSO970 MVC    *-*(,@BR),*-*(@Q,@XR)     MOVE THE FINAL SEGMENT IN
    0FF7                         3037        ORG    KSO970+@Q                 * THE DATA CONSTANT BUCKET
    0FF7 04               0FF7 3038        DC     AL1(B@LISP-1)             * TO THE VM BUFFER.
    0FF8 04               0FF8 3039        DC     AL1(B@LISP-1)             * INITIALLY SET TO MOVE
    0FF9 04               0FF9 3040        DC     AL1(B@LISP-1)             * A PACKED SHORT VALUE
                               3041 *
                               3042 * REPLACE PAGING ROUTINE MODULE
                               3043 *
    0FFA 3C 02 116C             3044 KSO972 MVI    KSODPL+@DCTRL,@DPUT       SET FUNC CODE
                               3045 *        DISK   KSODPL,WAIT              SAVE PAGING MODULE
    0FFE C0 87 0025             3046        B      $DISKN                   PERFORM PHYSICAL DISK OP
    1002 116C             1003 3047        DC     AL2(KSODPL)              DPL ADDRESS
    1004 C0 87 0025             3048        B      $DISKN                   WAIT AND CHECK DISK ERRORS
    1008 057F             1009 3049        DC     AL2($WAITF)              WAIT DPL ADDRESS
                               3050 *** END OF EXPANSION ***

                               3052 *
                               3053 * RESTORE CORE I/O ROUTINES
                               3054 *
                               3055 *
                               3056 * EXIT KSOVRL
                               3057 *
    100A C0 87 04A1             3058 KSO976 B      $CARPL                   EXIT KSOVRL

                               3060 ************************************************************************
                               3061 *                                                                      *
                               3062 * KSETIT ERROR PROCESSING ROUTINE                                       *
                               3063 *                                                                      *
                               3064 ************************************************************************
                               3065 *
    100E 3C 3C 03CD             3066 KSO991 MVI    $CAERR,@@E250            SET ERROR CONDITION CODE
    1012 F2 87 12               3067        J      KSO995                   CLEAR LINE PT
    1015 3C 3D 03CD             3068 KSO992 MVI    $CAERR,@@E251            SET ERROR CONDITION CODE
    1019 F2 87 0B               3069        J      KSO995                   CLEAR LINE PT
    101C 3C 3E 03CD             3070 KSO993 MVI    $CAERR,@@E252            SET ERROR CONDITION CODE
    1020 F2 87 04               3071        J      KSO995                   CLEAR LINE PT
    1023 3C 3F 03CD             3072 KSO994 MVI    $CAERR,@@E253            SET ERROR CONDITION CODE
    1027 C2 02 0000             3073 KSO995 LA     KSOE00,@XR               CLEAR LINE PT
    102B C0 87 0469             3074 KSO996 B      $CAERK                   ABORT KSETIT, PRINT ERROR MSG
                               3075 *
                               3076 *      $C4BD
```

```
      C4BIN2 - CONVERT DECIMAL TO BINARY ROUTINE


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  10/02/22  PAGE  17

                                3078+*                                                              *
                                3079+*                      INITIALIZATION                         *
                                3080+*                                                              *
                      102F 3081+C4BIN2 EQU   *                        ENTRY POINT
                      102F 3082+       USING C4BIN2,@BR              BASE VALUE
                                3083+*
     102F 34 01 1091            3084+       ST    C4B800+@OP1,@BR        SAVE CALLERS BASE REGISTER
     1033 C2 01 102F            3085+       LA    C4BIN2,@BR             LOAD BASE VALUE
                                3086+*
     1037 74 08 66              3087+       ST    C4B850+@OP1(,@BR),@ARR  SAVE RETURN ADDRESS
                                3088+*
     103A 74 02 6E              3089+       ST    C4BSAV(,@BR),@XR       SAVE VALUE OF POINTER
     103D 3C 0C 03CD            3090+       MVI   $CAERR,@@E122          SET ERROR CODE IN CASE
     1041 5C 01 6A 6B           3091+       MVC   C4BVAL(C4BLVL,@BR),C4BINI(,@BR) INIT VALUE TO ZERO
     1045 3C 04 109E            3092+C4B100 MVI   C4B900,4               INITLZ CHAR. COUNT
                                3093+*
                                3094+***   DETERMINE IF CHAR NUMERIC AND DECR CHAR COUNT
                                3095+*
     1049 F2 80 32              3096+C4B200 JC    C4B600,@NOP            SET TO UCB IF IMBEDDED BLANKS
                                3097+*                                  * ALLOWED
     104C BD F0 00              3098+C4B300 CLI   0(,@XR),C4BLOW         THIS CHAR NUMERIC ?
     104F F2 82 35              3099+       JL    C4B700                 NO, GOTO RETURN
                                3100+*
     1052 5F 00 6F 4E           3101+       SLC   C4B900(1,@BR),C4B590+@D1(,@BR) DECR CHAR COUNT
     1056 F2 82 35              3102+       JL    C4B800                 BR TO ERROR EXIT IF TOO MANY
                                3103+*
                                3104+***   MULTIPLY PREVIOUS VALUE BY TEN
                                3105+*
     1059 5E 01 6A 6A           3106+       ALC   C4BVAL(C4BLVL,@BR),C4BVAL(,@BR) DOUBLE PREVIOUS VALUE
     105D 5C 01 68 6A           3107+       MVC   C4BWRK(C4BLVL,@BR),C4BVAL(,@BR) SAVE DOUBLE VALUE
     1061 5E 01 6A 6A           3108+       ALC   C4BVAL(C4BLVL,@BR),C4BVAL(,@BR) QUADRUPLE PREVIOUS VALUE
     1065 5E 01 6A 6A           3109+       ALC   C4BVAL(C4BLVL,@BR),C4BVAL(,@BR) OCTUPLE PREVIOUS VALUE
     1069 5E 01 6A 68           3110+       ALC   C4BVAL(C4BLVL,@BR),C4BWRK(,@BR) ADD IN SAVED DOUBLE
                                3111+*
                                3112+***   ADD IN VALUE OF THIS CHAR AND INCR POINTER
                                3113+*
     106D 68 03 6C 00           3114+       MNN   C4BCHR(,@BR),0(,@XR)      FETCH NEMERIC VALUE OF NEW CHAR
     1071 5E 01 6A 6C           3115+       ALC   C4BVAL(C4BLVL,@BR),C4BCHR(,@BR) INCR VALU BY THIS CHAR
                                3116+*
     1075 E2 02 01              3117+       LA    @B1(,@XR),@XR          INCR POINTER TO NEXT CHAR
     1078 D0 87 1A              3118+       B     C4B200(,@BR)           GOTO DO IT AGAIN
                                3119+*                                                              *
                                3120+*                    ROUTINE TO SCAN BLANKS                    *
                                3121+*                                                              *
     107B E2 02 01              3122+C4B590 LA    @B1(,@XR),@XR          INCR POINTER TO NEXT CHAR
     107E BD 40 00              3123+C4B600 CLI   0(,@XR),@BLANK         IS THIS CHAR A BLANK ?
     1081 D0 01 1D              3124+       BNE   C4B300(,@BR)           RETURN IF NOT
     1084 D0 87 4C              3125+       B     C4B590(,@BR)           GET NEXT CHAR IF YES
```

```
                                  3127+*
                                  3128+***     ENDING ROUTINE
                                  3129+*
     1087 74 02 68                3130+C4B700 ST     C4BLEN(,@BR),@XR         PLACE VALUE OF POINTER
     108A 5F 01 68 6E             3131+       SLC    C4BLEN(2,@BR),C4BSAV(,@BR) SUBRACT ENTERING VALUE
                                  3132+*
     108E C2 01 0000              3133+C4B800 LA     *-*,@BR                  RESTORE CALLERS BR
                                  3134+*
     1092 C0 87 0000              3135+C4B850 B      *-*                      RETURN TO CALLING ROUTINE
                                  3136+*                                                             *
                                  3137+*                        WORK AREA AND CONSTANT              *
                                  3138+*                                                             *
     1096                    1097 3139+C4BWRK DS     CL2                      SAVE AREA FOR DOUBLED VALUE
                                  3140+*
                             1098 3141+C4BYT1 EQU    *                        FIRST BYTE OF BINARY VALUE
     1098                    1099 3142+C4BVAL DS     CL2                      SAVE AREA FOR BINARY VALUE
                                  3143+*
     109A 00                 109A 3144+C4BINI DC     XL1'00'                  INITIALIZE WA TO ZERO
                                  3145+*
     109B                    109B 3146+C4BCHR DS     CL1                      SAVE AREA FOR EACH NEW CHAR
     109B                         3147+       ORG    *-1                      INITIALIZE
     109B 00                 109B 3148+       DC     XL1'00'                  * TO ZERO
                                  3149+*
     109C                    109D 3150+C4BSAV DS     CL2                      SAVE AREA FOR XR
                                  3151+*
     109E                    109E 3152+C4B900 DS     CL1                      SAVE AREA FOR CHAR COUNTER
                                  3153+*                                                         *
                                  3154+*                     EQUATES FOR C4BIN2                    *
                                  3155+*                                                         *
                             1097 3156+C4BLEN EQU    C4BWRK                   ON RETURN WILL CONTAIN COUNT
                                  3157+*                                      * @XR INCREMENTED BY
                             0004 3158+C4BCHC EQU    4                        NUMBER OF CHAR TO CONVERT
                                  3159+*
                             00F0 3160+C4BLOW EQU    C'0'                     LOWEST NUMERIC CHARACTER
                                  3161+*
                             0002 3162+C4BLVL EQU    C4BVAL-C4BWRK            LENGTH OF BINARY VALUE
                                  3163+*
                             104A 3164+C4BLNK EQU    C4B200+@Q                LOCATION OF IMBEDDED BLANK IND
                                  3165+*
                             0087 3166+C4BSPC EQU    @UCB                     MOVED TO C4BLNK TO ALLOW BLANKS
                                  3167+*
                             1046 3168+C4BNMC EQU    C4B100+@Q                LOCATION OF CONVERSION COUNT
                                  3169+*
                             0080 3170+C4BNOP EQU    @NOP                     CHANGED IF IMBEDDED BLANK OK
                             109F 3171+C4END  EQU    *                        DEFINE END OF CODE
                                  3172+***                      END OF C4BIN2                        ***
                                  3173 *      $CANI
```

SCANIT - DELIMETER SCAN MODULE

```
3175+*****************************************************************************
3176+*  5703-XM1    COPYRIGHT IBM CORP. 1970                               *
3177+*                    REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083   *
3178+*                                                                     *
3179+*****************************************************************************
3180+*STATUS                                                               *
3181+*   VERSION 1 MODIFICATION 0                                          *
3182+*                                                                     *
3183+*FUNCTION                                                             *
3184+*   THE FUNCTION OF SCANIT IS TO SCAN PAST VALID DELIMITERS AND       *
3185+*   RETURN A POINTER TO THE FIRST CHARACTER THAT'S NOT A DELIMITER.   *
3186+*                                                                     *
3187+*ENTRY POINTS                                                         *
3188+*   * THE ENTRY POINT IS SCANIT.                                      *
3189+*   * THE CALLING SEQUENCE IS AS FOLLOWS:                             *
3190+*            B    SCANIT                                              *
3191+*     WITH REGISTER 2 (@XR) POINTING TO THE FIRST CHARACTER TO BE     *
3192+*     EXAMINED.                                                       *
3193+*                                                                     *
3194+*INPUT                                                                *
3195+*   NONE                                                              *
3196+*                                                                     *
3197+*OUTPUT                                                               *
3198+*   NONE                                                              *
3199+*                                                                     *
3200+*EXTERNAL REFERENCES                                                  *
3201+*   $CAERR - ERROR CODE SAVE AREA                                     *
3202+*                                                                     *
3203+*EXITS, NORMAL                                                        *
3204+*   NORMAL EXIT FROM SCANIT IS TO THE BYTE FOLLOWING THE BRANCH TO    *
3205+*   SCANIT IN THE CALLING ROUTINE. THE PSR (REGISTER 4) WILL CONTAIN  *
3206+*   A ZERO IF NO DELIMITERS WERE FOUND OR A HIGH CONDITION IF ONE OR  *
3207+*   MORE DELIMITERS WERE SCANNED.                                     *
3208+*                                                                     *
3209+*EXITS, ERROR                                                         *
3210+*   ERROR EXIT FROM SCANIT IS TO THE BYTE FOLLOWING THE BRANCH TO     *
3211+*   SCANIT IN THE CALLING ROUTINE. THE PSR WILL CONTAIN A LOW         *
3212+*   CONDITION.                                                        *
3213+*                                                                     *
3214+*TABLES/WORKAREAS                                                     *
3215+*   * SCACNT - AREA CONTAINING NUMBERS OF DELIMITERS SCANNED          *
3216+*   * SCAMMA - LOC WHERE SCACOM MAY BE MOVED IF ONE COMMA IS ALSO     *
3217+*     TO BE CONSIDERED A DELIMITER. MOVING SCACOF BACK INTO SCAMMA    *
3218+*     INDICATES THAT ONLY BLANKS SHOULD BE CONSIDERED DELIMITERS.     *
3219+*                                                                     *
3220+*ATTRIBUTES                                                           *
3221+*   RELOCATABLE AND RE-USABLE                                         *
3222+*                                                                     *
3223+*CHARACTER CODE DEPENDENCY                                            *
3224+*   THE OPERATION OF THIS MODULE DOES NOT DEPEND UPON A PARTICULAR    *
3225+*   INTERNAL REPRESENTATION OF THE EXTERNAL CHARACTER SET.            *
3226+*                                                                     *
3227+*NOTES                                                                *
3228+*   ERROR PROCEDURES                                                  *
3229+*      THE ONLY ERROR CONDITION DETECTED BY SCANIT IS THE CASE WHERE  *
3230+*      A CARRIAGE-RETURN CODE FOLLOWS A COMMA.  UPON RETURN TO THE    *
```

```
                          3231+*        CALLING ROUTINE, @PSR WILL BE SET TO A LOW CONDITION, THE        *
                          3232+*        ERROR CODE IS SET IN $CAERR, AND MG WILU BE POINTING TO THE      *
                          3233+*        CARRIAGE-RETURN CHARACTER.                                       *
                          3234+*                                                                         *
                          3235+*   REGISTER USAGE                                                        *
                          3236+*      REGISTER 2 (@XR) IS USED AS A POINTER ACROSS THE AREA BEING        *
                          3237+*      SCANNED FOR DELIMETERS.                                            *
                          3238+*                                                                         *
                          3239+*   SAVED/RESTORED AREAS                                                  *
                          3240+*      UPON ENTRY TO SCANIT, REGISTER 8 (@ARR) IS SAVED AND USED AS       *
                          3241+*      THE RETURN ADDRESS.                                                *
                          3242+*                                                                         *
                          3243+*   MODIFICATION CONSIDERATIONS                                           *
                          3244+*      NONE                                                               *
                          3245+*                                                                         *
                          3246+*   REQUIRED MODULES                                                      *
                          3247+*      * @SYSEQ - COMMON SYSTEM EQUATES                                   *
                          3248+*      * @FXDEQ - FIXED NUCLEUS ADDRESSES EQUATES                         *
                          3249+*                                                                         *
                          3250+*   OTHER                                                                 *
                          3251+*      SCANIT IS INITIALIZED TO BYPASS BLANKS ONLY. IF SCACOM IS          *
                          3252+*      MOVED TO SCAMMA, ONE COMMA WILL BE SCANNED ALONG WITH BLANKS.      *
                          3253+*      THE INSTRUCTION TO DO THIS IS AS FOLLOWS:                          *
                          3254+*            MVI   SCAMMA,SCACOM                                          *
                          3255+*                                                                         *
                          3256+*      TO DROP THE COMMA FROM ITS DELIMITER STATUS, SCACOF SHOULD BE      *
                          3257+*      MOVED TO SCAMMA, USING THE FOLLOWING INSTRUCTION:                  *
                          3258+*            MVI   SCAMMA,SCACOF                                          *
                          3259+*                                                                         *
                          3260+***********************************************************************************


                          3262+*
                          3263+*                         EQUATES USED IN THIS SUBROUTINE
                          3264+*
               0001       3265+SCAINC EQU    1                        TO INCREMENT POINTER
               0001       3266+SCACOM EQU    @BNE                     SWITCH TO ALLOW SCANNING COMMA
               0087       3267+SCACOF EQU    @UCB                     SWITCH TO SET OFF THE INDICATON
                          3268+*                                      * FOR SCANNING A COMMA
               109F       3269+SCANIT EQU    *                        ENTRY POINT TO THIS SUBROUTINE
  109F 34 08 10DB         3270+       ST     SCA500+@OP1,@ARR         SAVE RETURN ADDRESS
  10A3 34 02 10DD         3271+       ST     SCASVE,@XR               SAVE POINTER VALUE
  10A7 3C 04 03CD         3272+       MVI    $CAERR,@@E110            SET ERROR CODE
  10AB F2 87 03           3273+       J      SCA200                   GO TO PROCESS
  10AE E2 02 01           3274+SCA100 LA     SCAINC(,@XR),@XR         INCREMENT POINTER TO NEXT CHAR
  10B1 BD 40 00           3275+SCA200 CLI    0(,@XR),@BLANK           IS THIS CHAR BLANK ?
  10B4 C0 81 10AE         3276+       BE     SCA100                   YES, FETCH NEXT ONE
  10B8 BD 6B 00           3277+       CLI    0(,@XR),@COMMA           IS IT A COMMA ?
  10BB F2 87 10           3278+SCA250 JC     SCA400,@UCB              UCS TO RETURN -- OR NOP IF
                          3279+*                                      * SCAMMA IS ACTIVE AND CHAR
  10BE E2 02 01           3280+SCA300 LA     SCAINC(,@XR),@XR         INCREMENT POINTER TO NEXT CHAR
  10C1 BD 40 00           3281+       CLI    0(,@XR),@BLANK           IS THIS CHAR A BLANK ?
  10C4 C0 81 10BE         3282+       BE     SCA300                   YES, FETCH NEXT ONE
  10C8 BD 1F 00           3283+       CLI    0(,@XR),@EOS+1           IS THIS EOS ?
  10CB F2 82 0A           3284+       JL     SCA500                   IF NOT, SKIP ERROR ROUTINE
  10CE 34 02 10DF         3285+SCA400 ST     SCACNT,@XR               SAVE NEW POINTER VALUE
```

```
      SCANIT - DELIMETER SCAN MODULE


ERR LOC   OBJECT CODE     ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  10/02/22  PAGE  21

    10D2 0F 01 10DF 10DD       3286+      SLC    SCACNT(2),SCASVE           SET PSR TO EQUAL IF POINTER
                               3287+*                                      * NOT ADVANCED
    10D8 C0 87 0000            3288+SCA500 B     *-*                       YES, RETURN
                          10BC 3289+SCAMMA EQU   SCA250+@Q                 TO SET SCAN COMMA INDICATOR
                               3290+*
                               3291+*                 SAVE AREA
                               3292+*
                          10DC 3293+SCASV1 EQU   *                         FIRST BYTE OF SCASVE
    10DC                  10DD 3294+SCASVE DS    CL2                       ORIGINAL POINTER VALUE SAVE
    10DE                  10DF 3295+SCACNT DS    CL2                       SAVE AREA FOR TOTAL CHAR SCAN
                               3296+***                     END OF SCANIT                         ***
                               3297 *        PATCH
                               3298 *********************************************************************
                               3299 *       PATCH AREA 1                                              *
                               3300 *********************************************************************
                               3301 *   CALCULATE AREA LEFT IN THIS SECTOR
                               3302 *
                          10E0 3303 $$$$L1 EQU   *                         START PATCH AREA 1
    1100                       3304        ORG   *,256,0                    SET LOC CNTR TO NEXT SECTOR
                          1100 3305 $$$$T1 EQU   *                         DEFINE ADDR OF SCTR BNDRY
    10E0                       3306        ORG   $$$$L1                     SET LOC CNTR OF START
                               3307 *                                      * OF PATCH AREA
    10E0                  10FF 3308 $$$$$1 DS    CL($$$$T1-$$$$L1)          PATCH AREA
                               3309 *** END OF EXPANSION ***
```

```
                     3311 ********************************************************************************
                     3312 *                                                                              *
                     3313 * KSOVRL EQUATES FOR PROGRAM WORK AREAS AND CONSTANTS                          *
                     3314 *                                                                              *
                     3315 ********************************************************************************
                     3316 *
                1120 3317 KSOEQU EQU   X'1120'+$$ZERO          CADDR WORK AREA
                     3318 *
                     3319 * EQUATES REFERENCING CONSTANTS
                     3320 *
                0000 3321 KSOE00 EQU   0                       TO CLEAR LINE PT
                0000 3322 KSOOFF EQU   0                       TO SET SWS OFF
                0000 3323 KSOPD0 EQU   0                       LINE PT DISP OF 0
                0001 3324 KSOONN EQU   1                       TO SET SWS ON
                0001 3325 KSOPD1 EQU   1                       LINE PT DISP OF 1
                0002 3326 KSOLAC EQU   2                       DIGIT ACCUM LNG
                00FF 3327 KSOPGB EQU   255                     VM PG BOUNDARY
                     3328 *
                     3329 * CONSTANTS
                     3330 *
                1121 3331 KSOI00 EQU   X'1121'+$$ZERO          INTEGER OF 0
                1123 3332 KSOI01 EQU   X'1123'+$$ZERO          INTEGER OF 1
                1125 3333 KSOI02 EQU   X'1125'+$$ZERO          INTEGER OF 2
                112B 3334 KSOAAC EQU   X'112B'+$$ZERO          VADDR TO CADDR CONV CONSTANT
                116C 3335 KSODPL EQU   X'116C'+$$ZERO          DPL CADDR
                     3336 *
                     3337 * WORK AREAS
                     3338 *
                118F 3339 KSOCNT EQU   X'118F'+$$ZERO          ELEMENT COUNT
                117E 3340 KSOCRS EQU   X'117E'+$$ZERO          CHAR REF SW
                1189 3341 KSODGT EQU   X'1189'+$$ZERO          IDT INDEX ACCUM
                1190 3342 KSOELL EQU   X'1190'+$$ZERO          ARRAY ELEMENT LNG
                1191 3343 KSOEND EQU   X'1191'+$$ZERO          MAX LNG OF MANTISSA
                1183 3344 KSOEXP EQU   X'1183'+$$ZERO          EXP SAVE AREA
                1180 3345 KSOICS EQU   X'1180'+$$ZERO          INTERNAL CONS SW
                1182 3346 KSOLSA EQU   X'1182'+$$ZERO          LETTER SAVE AREA
                1192 3347 KSOMFS EQU   X'1192'+$$ZERO          FRACTION SN
                118B 3348 KSOSC1 EQU   X'118B'+$$ZERO          SUBSC 1
                118D 3349 KSOSC2 EQU   X'118D'+$$ZERO          SUBSC 2
                1193 3350 KSOCBS EQU   X'1193'+$$ZERO          CONSTANT BUCKET SN
                117F 3351 KSOSTS EQU   X'117F'+$$ZERO          SUBSC TERMINATION SW
                1185 3352 KSOVAD EQU   X'1185'+$$ZERO          CONTAINS VAR VADDR
                     3353 *
                     3354 * EQUATES REFERENCING PROGRAM
                     3355 *
                070C 3356 KSOLVT EQU   X'070C'                 CADDR 1ST LVT ENTRY
                0746 3357 KSOLDT EQU   X'0746'                 CADDR 1ST LOT ENTRY
                098A 3358 KSOCVT EQU   X'098A'                 CADDR 1ST CVT ENTRY
                09FE 3359 KSOCAT EQU   X'09FE'                 CADDR 1ST CAT ENTRY
                09C4 3360 KSONAT EQU   X'09C4'                 CADDR 1ST NAT ENTRY
                FEFF 3361 KSO2LS EQU   X'FEFF'                 RH BYTE IN 1ST PG OF FAT
                FFFF 3362 KSOLST EQU   X'FFFF'                 RH BYTE IN LAST PG OF FAT
                0B07 3363 KSO1PG EQU   X'0B07'                 CADDR OF RH BYTE OF 1ST PG FAT
                0C07 3364 KSO2PG EQU   X'0C07'                 CADDR OF RH BYTE OF 2ND PG FAT
                0095 3365 KSO1LN EQU   X'95'                   LNG OF 1ST SEGMENT OF FAT
                     3366 *
```

```
                         11A7 3367 KSOATB EQU    X'11A7'+$$ZERO           CADDR ALPHA TBL
                         1187 3368 KSOATI EQU    X'1187'+$$ZERO           ADDR TBL INDEX
                         1194 3369 KSOCDB EQU    X'1194'+$$ZERO           CADDR DATA BUCKET
                         1194 3370 KSOCB0 EQU    X'1194'+$$ZERO           DATA BUCKET DISP OF 0
                         1195 3371 KSOCB1 EQU    X'1195'+$$ZERO           DATA BUCKET DISH OF 1
                         1196 3372 KSOCB2 EQU    X'1196'+$$ZERO           DATA BUCKET DISP OF 2
                         1197 3373 KSOCB3 EQU    X'1197'+$$ZERO           DATA BUCKET DISP OF 3
                         1198 3374 KSOCB4 EQU    X'1198'+$$ZERO           DATA BUCKET DISP OF 4
                         1199 3375 KSOCB5 EQU    X'1199'+$$ZERO           DATA BUCKET DISP OF 5
                         119A 3376 KSOCB6 EQU    X'119A'+$$ZERO           DATA BUCKET DISP OF 6
                         119B 3377 KSOCB7 EQU    X'119B'+$$ZERO           DATA BUCKET DISP OF 7
                         119C 3378 KSOCB8 EQU    X'119C'+$$ZERO           DATA BUCKET DISP OF 8
                         119D 3379 KSOCB9 EQU    X'119D'+$$ZERO           DATA BUCKET DISP OF 9
                         119E 3380 KSOCBA EQU    X'119E'+$$ZERO           DATA BUCKET DISP OF 10
                         119F 3381 KSOCBB EQU    X'119F'+$$ZERO           DATA BUCKET DISP OF 11
                         11A0 3382 KSOCBC EQU    X'11A0'+$$ZERO           DATA BUCKET DISP OF 12
                         11A1 3383 KSOCBD EQU    X'11A1'+$$ZERO           DATA BUCKET DISP OF 13
                         11A2 3384 KSOCBE EQU    X'11A2'+$$ZERO           DATA BUCKET DISP OF 14
                         11A3 3385 KSOCBF EQU    X'11A3'+$$ZERO           DATA BUCKET DISP OF 15
                         11A4 3386 KSOCBG EQU    X'11A4'+$$ZERO           DATA BUCKET DISP OF 16
                         11A6 3387 KSOCHR EQU    X'11A6'+$$ZERO           DATA BUCKET DISP OF 18
                              3388 *
                         FFFF 3389          END
```

TOTAL STATEMENTS IN ERROR IN THIS ASSEMBLY =   0

```
$$$$$$  001  0C20 2551
$$$$$1  032  10FF 3308
$$$$L1  001  10E0 3303    3306   3308
$$$$T1  001  1100 3305    3308
$$$CMD  001  0020 0658
$$$DAT  001  0040 0657
$$$EPL  001  0091 0654
$$$ERN  001  0080 0708
$$$FUN  001  0010 0659
$$$NLN  001  00A0 0704
$$$STD  001  0081 0653
$$BNLN  001  0605 0634    0636
$$CDBS  001  08C0 0684
$$CDND  001  0666 0643
$$CDRD  001  0890 0682    0684
$$CKEY  001  0603 0632
$$CKFF  001  0B3D 0664
$$COFF  001  0B44 0663
$$CSNS  001  209C 0693
$$DATB  001  0BBF 0665
$$EOSA  001  0AFE 0662
$$ERSK  001  1C00 0703
$$FITS  001  1D00 0711
$$FLIB  001  06FF 0710
$$ILEN  001  0601 0628    0630   0634
$$ILHD  001  0600 0626    0628
$$INLN  001  0607 0641    0643   0645
$$INND  001  06FA 0645
$$KBDT  001  09E1 0652    0656
$$KBSN  001  09E2 0656    0661
$$KLD1  001  0600 0716
$$KLD2  001  0700 0718
$$KLD3  001  0C00 0720
$$LPOS  001  09EB 0661
$$PCNT  001  07E9 0677
$$PLYN  001  2004 0691
$$PRES  001  0890 0650    0652   0662   0663   0664   0665   0682
$$PRFL  001  2143 0695
$$PRNT  001  0707 0671    0672   0676   0677
$$PRTN  001  0782 0672
$$PSIO  001  07CE 0676
$$PYCD  001  2200 0697
$$PYMP  001  2000 0689    0691   0693   0695   0697
$$SLIB  001  1C00 0706
$$TPCD  001  0606 0636    0641
$$UPAR  001  0602 0630    0632
$$WSPB  001  1E00 0709
$$XIND  001  06FF 0707    0710
$$ZERO  001  0000 0222    0223   0225   0226   0227   0231   0689   3317   3331   3332   3333   3334   3335
                          3339   3340   3341   3342   3343   3344   3345   3346   3347   3348   3349   3350
                          3351   3352   3367   3368   3369   3370   3371   3372   3373   3374   3375   3376
                          3377   3378   3379   3380   3381   3382   3383   3384   3385   3386   3387
$ABORT  001  0010 0335
$BASIC  001  0080 0393
$BIGCD  001  0080 0469
$BLDPL  001  0579 0602    0604
```

```
                              CROSS REFERENCE

SYMBOL  LEN VALUE DEFN   REFERENCES                        VER 15, MOD 00  10/02/22  PAGE  25

$BLNOE  001  0569 0592
$BLOAD  001  0522 0583   0585  0588  0601  0602
$BLRTN  001  0550 0591   0592
$BRSAV  001  03C5 0280   0281
$BSADR  001  0587 0607   0609
$BUFPT  001  03E3 0488   0489
$CABLD  001  04B4 0561   0562
$CAERK  001  0469 0538   0541  3074
$CAERR  001  03CD 0286   0288  2726* 3066* 3068* 3070* 3072* 3090* 3272*
$CAIPL  001  049D 0557   0559
$CALLI  001  0008 0478
$CARDI  001  0001 0249
$CARPL  001  04A1 0559   0561  3058
$CIENT  001  0483 0548   0549
$CIEXT  001  0480 0547   0548
$CIMSK  001  0476 0544   0547
$CISUS  001  0496 0552   0557
$CLBFR  001  0010 0436
$CMDKY  001  0008 0348
$CMODE  001  0002 0398
$CONFG  001  03DD 0461   0471
$CRPOS  001  03E2 0487   0488
$CRTAD  001  044D 0526   0527
$CRTAV  001  0002 0342
$CRTDN  001  0002 0366
$CRTIN  001  03D3 0363   0370
$CRTNO  001  0004 0345
$CRTPU  001  0004 0367
$CRTSP  001  0008 0368
$CRTUP  001  0001 0365
$CRUSH  001  0080 0474
$CSDPL  001  050E 0573   0574
$C0001  001  0464 0530   0536
$DATE   001  043A 0511   0512
$DBGUF  001  03E0 0473   0482
$DBLOK  001  0001 0423
$DFDET  001  03E8 0494   0495
$DISKN  001  0025 0225   2571  2573  3046  3048
$DKERR  001  0008 0404
$DKSIZ  001  03D7 0448   0456  0497
$DK100  001  0001 0450
$DK200  001  0002 0451
$DK400  001  0004 0452
$DK600  001  0008 0453
$DK800  001  0010 0454
$DPLSV  001  0449 0522   0524
$DTNMB  001  0040 0269
$DTRDR  001  0040 0357
$ENDNU  001  0600 0616   0626  0650  0671  0707  0716  0718  0720  1354
$ERDPL  001  046F 0541   0543
$ERFIL  001  0040 0296
$ERHRD  001  0004 0428
$ERKEY  001  0080 0300
$ERLOG  001  0345 0230
$ERMAD  001  0472 0543   0544
$ERPND  001  0004 0401
```

```
$ERRCT  001  03CF 0302
$ERRPG  001  03CE 0290
$ERSFL  001  0035 0295
$ERSTK  001  0030 0293
$ER050  001  0363 0231
$ER1N2  001  0050 0298
$EXADR  001  0517 0576   0578
$EXCMD  001  0001 0330
$EXFTR  001  043B 0512   0517
$FCIND  001  0010 0408
$FDIND  001  0040 0415
$FEARR  001  0004 0223
$FEMAP  001  0588 0609   0610
$FILIB  001  03DA 0459   0460
$FITIN  001  0010 0384
$FUIND  001  0020 0413
$GUFIO  001  0583 0606   0607
$GUFIR  001  0008 0258
$HISTE  001  042E 0509   0510
$HIST1  001  0435 0510   0511
$HRDER  001  0020 0354
$INDR1  001  03D4 0370   0396
$INDR2  001  03D5 0396   0421
$INDR3  001  03D6 0421   0448
$INLNO  001  03CF 0288   0290   0302   0309
$INRPT  001  0020 0266
$IOIND  001  03D2 0337   0363
$IOPGS  001  0010 0477
$IOYES  001  0002 0252
$IPLDV  001  05FF 0613   0616
$IRKEY  001  0020 0476
$KEYBD  001  03E1 0482   0487
$KEYCD  001  03C3 0246   0280
$KEYDT  001  0040 0390
$KE090  001  00DE 0226
$KE130  001  01D5 0227
$KSOVR  001  0C27 2554
$KYBSY  001  0010 0263
$LDRTN  001  0571 0601
$LEVEL  001  03DF 0471   0473
$LIST   001  0002 0425
$LMRGN  001  03C1 0241   0243
$LNPTR  001  0080 0360
$LOADB  001  054A 0585
$LOADR  001  051A 0578   0581
$LPRIO  001  03EA 0495
$LPROS  001  03E5 0490   0492
$LPRP3  001  03E4 0489   0490
$MOUNT  001  0020 0439
$MPDWN  001  0001 0339
$NEXTB  001  03E6 0492   0493
$NEXTL  001  03E7 0493   0494
$NOENB  001  0008 0431
$NOLST  001  0004 0255
$NUCBS  001  03C0 0238   0239
$NWRKF  001  0080 0444
```

```
$NWRKR  001  0040 0441
$PASWD  001  042D 0508   0509
$PAUSD  001  04BA 0562   0564
$PAUSE  001  0002 0332
$PGMDT  001  0020 0387
$PGMST  001  0010 0351
$PKERT  001  0419 0506   0508
$PLST1  001  0454 0527   0528
$PLST2  001  045B 0528   0529
$PLST3  001  0462 0529   0530
$PRDEV  001  044B 0524   0526
$PRESN  001  0002 0375
$PROCI  001  0001 0372
$PRPOS  001  03C2 0243   0246
$PSDBR  001  04FA 0567
$PSDXR  001  04F2 0566   0567
$PSTEP  001  0004 0333
$PSTMT  001  0008 0334
$PTCH1  001  03F5 0497   0501
$READY  001  0080 0417
$REORD  001  0040 0475
$RLOAD  001  051E 0581   0583
$RMRGN  001  03C0 0239   0241
$RSTR   001  04D6 0564   0566  0568  0573
$RUNIT  001  0001 0311
$SFAID  001  050D 0569
$SPRNT  001  0465 0536   0538
$SRTRN  001  04FE 0568   0569
$STEPT  001  0002 0312
$SWPCR  001  0511 0574   0576
$TABLN  001  03CB 0283   0286
$TFLOW  001  0008 0318
$TRACE  001  0004 0313
$TRALL  001  0010 0319
$TROVR  001  054E 0588   0591
$TRUNK  001  0080 0271
$TRVAR  001  0020 0320
$UNMSK  001  048D 0549   0552
$USRDR  001  03DC 0460   0461
$VMDEF  001  0080 0324
$VOLF1  001  03FE 0503   0504
$VOLF2  001  040E 0505
$VOLID  001  03F6 0501   0502  0506
$VOLR1  001  03F6 0502   0503
$VOLR2  001  0406 0504   0505
$WAITF  001  057F 0604   0606  2574  3049
$WFDEF  001  0040 0518
$WFLOK  001  0008 0381
$WFNME  001  0443 0517   0522
$WSIND  001  0004 0378
$XIND1  001  03D0 0309   0328  2586  2916
$XIND2  001  03D1 0328   0337
$XIND3  001  03D8 0456   0459
$XPREC  001  0040 0321   2586  2916
$XRSAV  001  03C7 0281   0283
$ZTRAD  001  05A2 0610
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES
$12K	001	0004	0465	
$16CKY	001	0008	0467	
$16K	001	0002	0464	
$22IMP	001	0001	0462	
#$$KSO	001	0C20	2548	2550
#$@KSO	001	0005	2549	
#$KSOV	001	0AC8	2547	
#KSOVR	001	0000	0001	
@@E001	001	0000	1258	1260
@@E003	001	0001	1260	1262
@@E004	001	0002	1262	1264
@@E005	001	0003	1264	1266
@@E006	001	0004	1266	1268
@@E007	001	0005	1268	1270
@@E008	001	0006	1270	1272
@@E009	001	0007	1272	1274
@@E010	001	0008	1274	1276
@@E011	001	0009	1276	1278
@@E012	001	000A	1278	1280
@@E013	001	000B	1280	1282
@@E014	001	000C	1282	1284
@@E015	001	000D	1284	1286
@@E016	001	000E	1286	1288
@@E017	001	000F	1288	1290
@@E018	001	0010	1290	1292
@@E019	001	0011	1292	1294
@@E020	001	0012	1294	1296
@@E021	001	0013	1296	1298
@@E023	001	0014	1298	1300
@@E024	001	0015	1300	1302
@@E025	001	0016	1302	1304
@@E026	001	0017	1304	1306
@@E027	001	0018	1306	1308
@@E028	001	0019	1308	1310
@@E029	001	001A	1310	1312
@@E030	001	001B	1312	1314
@@E031	001	001C	1314	1316
@@E032	001	001D	1316	1318
@@E035	001	001E	1318	1320
@@E036	001	001F	1320	1322
@@E037	001	0020	1322	1324
@@E038	001	0021	1324	1326
@@E039	001	0022	1326	1328
@@E040	001	0023	1328	1330
@@E041	001	0024	1330	1332
@@E042	001	0025	1332	1334
@@E043	001	0026	1334	1336
@@E044	001	0027	1336	1338
@@E045	001	0028	1338	1340
@@E046	001	0029	1340	1342
@@E060	001	002A	1342	1344
@@E080	001	002B	1344	
@@E100	001	0000	0730	0732
@@E101	001	0001	0732	0734
@@E102	001	0002	0734	0736
@@E103	001	0003	0736	0738

SYMBOL	LEN	VALUE	DEFN	REFERENCES	
@@E110	001	0004	0738	0740	3272
@@E112	001	0005	0740	0742	
@@E113	001	0006	0742	0744	
@@E114	001	0007	0744	0746	
@@E115	001	0008	0746	0748	
@@E116	001	0009	0748	0750	
@@E117	001	000A	0750	0752	
@@E120	001	000B	0752	0754	
@@E122	001	000C	0754	0756	3090
@@E123	001	000D	0756	0758	
@@E124	001	000E	0758	0760	
@@E129	001	000F	0760	0762	
@@E130	001	0010	0762	0764	
@@E131	001	0011	0764	0766	
@@E133	001	0012	0766	0768	
@@E134	001	0013	0768	0770	
@@E135	001	0014	0770	0772	
@@E136	001	0015	0772	0774	
@@E137	001	0016	0774	0776	
@@E138	001	0017	0776	0778	
@@E139	001	0018	0778	0780	
@@E142	001	0019	0780	0782	
@@E143	001	001A	0782	0784	
@@E150	001	001B	0784	0786	
@@E151	001	001C	0786	0788	
@@E160	001	001D	0788	0790	
@@E162	001	001E	0790	0792	
@@E163	001	001F	0792	0794	
@@E164	001	0020	0794	0796	
@@E200	001	0021	0796	0798	
@@E205	001	0022	0798	0800	
@@E210	001	0023	0800	0802	
@@E211	001	0024	0802	0804	
@@E212	001	0025	0804	0806	
@@E213	001	0026	0806	0808	
@@E215	001	0027	0808	0810	
@@E216	001	0028	0810	0812	
@@E217	001	0029	0812	0814	
@@E220	001	002A	0814	0816	
@@E221	001	002B	0816	0818	
@@E222	001	002C	0818	0820	
@@E223	001	002D	0820	0822	
@@E225	001	002E	0822	0824	
@@E226	001	002F	0824	0826	
@@E227	001	0030	0826	0828	
@@E228	001	0031	0828	0830	
@@E229	001	0032	0830	0832	
@@E230	001	0033	0832	0834	
@@E232	001	0034	0834	0836	
@@E234	001	0035	0836	0838	
@@E237	001	0036	0838	0840	
@@E240	001	0037	0840	0842	
@@E241	001	0038	0842	0844	
@@E242	001	0039	0844	0846	
@@E248	001	003A	0846	0848	
@@E249	001	003B	0848	0850	

```
                                CROSS REFERENCE

SYMBOL   LEN VALUE DEFN    REFERENCES                      VER 15, MOD 00  10/02/22  PAGE  30

@@E250   001  003C 0850    0852  3066
@@E251   001  003D 0852    0854  3068
@@E252   001  003E 0854    0856  3070
@@E253   001  003F 0856    0858  3072
@@E254   001  0040 0858    0860
@@E255   001  0041 0860    0862
@@E256   001  0042 0862    0864  2726
@@E300   001  0043 0864    0866
@@E301   001  0044 0866    0868
@@E302   001  0045 0868    0870
@@E303   001  0046 0870    0872
@@E304   001  0047 0872    0874
@@E305   001  0048 0874    0876
@@E308   001  0049 0876    0878
@@E310   001  004A 0878    0880
@@E315   001  004B 0880    0882
@@E316   001  004C 0882    0884
@@E320   001  004D 0884    0886
@@E325   001  004E 0886    0888
@@E330   001  004F 0888    0890
@@E335   001  0050 0890    0892
@@E338   001  0051 0892    0894
@@E340   001  0052 0894    0896
@@E350   001  0053 0896    0898
@@E351   001  0054 0898    0900
@@E352   001  0055 0900    0902
@@E360   001  0056 0902    0904
@@E361   001  0057 0904    0906
@@E362   001  0058 0906    0908
@@E371   001  0059 0908    0910
@@E380   001  005A 0910    0912
@@E390   001  005B 0912    0914
@@E400   001  005C 0914    0916
@@E410   001  005D 0916    0918
@@E415   001  005E 0918    0920
@@E417   001  005F 0920    0922
@@E420   001  0060 0922    0924
@@E430   001  0061 0924    0926
@@E432   001  0062 0926    0928
@@E433   001  0063 0928    0930
@@E450   001  0064 0930    0932
@@E451   001  0065 0932    0934
@@E460   001  0066 0934    0936
@@E461   001  0067 0936    0938
@@E464   001  0068 0938    0940
@@E465   001  0069 0940    0942
@@E466   001  006A 0942    0944
@@E467   001  006B 0944    0946
@@E469   001  006C 0946    0948
@@E470   001  006D 0948    0950
@@E471   001  006E 0950    0952
@@E473   001  006F 0952    0954
@@E474   001  0070 0954    0956
@@E475   001  0071 0956    0958
@@E476   001  0072 0958    0960
@@E477   001  0073 0960    0962
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES
@@E478	001	0074	0962	0964
@@E479	001	0075	0964	0966
@@E480	001	0076	0966	0968
@@E481	001	0077	0968	0970
@@E482	001	0078	0970	0972
@@E483	001	0079	0972	0974
@@E484	001	007A	0974	0976
@@E485	001	007B	0976	0978
@@E486	001	007C	0978	0980
@@E487	001	007D	0980	0982
@@E488	001	007E	0982	0984
@@E489	001	007F	0984	0986
@@E490	001	0080	0986	0988
@@E491	001	0081	0988	0990
@@E492	001	0082	0990	0992
@@E493	001	0083	0992	0994
@@E494	001	0084	0994	0996
@@E495	001	0085	0996	0998
@@E496	001	0086	0998	1000
@@E497	001	0087	1000	1002
@@E498	001	0088	1002	1004
@@E500	001	0089	1004	1006
@@E501	001	008A	1006	1008
@@E530	001	008B	1008	1010
@@E531	001	008C	1010	1012
@@E535	001	008D	1012	1014
@@E540	001	008E	1014	1016
@@E541	001	008F	1016	1018
@@E542	001	0090	1018	1020
@@E543	001	0091	1020	1022
@@E544	001	0092	1022	1024
@@E545	001	0093	1024	1026
@@E546	001	0094	1026	1028
@@E547	001	0095	1028	1030
@@E548	001	FFFF	1234	
@@E549	001	0096	1030	1032
@@E550	001	0097	1032	1034
@@E551	001	0098	1034	1036
@@E552	001	0099	1036	1038
@@E553	001	009A	1038	1040
@@E554	001	009B	1040	1042
@@E555	001	009C	1042	1044
@@E556	001	009D	1044	1046
@@E558	001	009E	1046	1048
@@E570	001	009F	1048	1050
@@E571	001	00A0	1050	1052
@@E572	001	00A1	1052	1054
@@E573	001	00A2	1054	1056
@@E574	001	00A3	1056	1058
@@E575	001	FFFF	1236	
@@E578	001	00A4	1058	1060
@@E579	001	FFFF	1238	
@@E580	001	FFFF	1240	
@@E585	001	00A5	1060	1062
@@E595	001	FFFF	1242	
@@E597	001	FFFF	1244	

SYMBOL	LEN	VALUE	DEFN	REFERENCES
@@E598	001	FFFF	1246	
@@E600	001	00A6	1062	1064
@@E601	001	00A7	1064	1066
@@E602	001	00A8	1066	1068
@@E603	001	00A9	1068	1070
@@E604	001	00AA	1070	1072
@@E606	001	00AB	1072	1074
@@E607	001	00AC	1074	1076
@@E608	001	00AD	1076	1078
@@E609	001	00AE	1078	1080
@@E610	001	00AF	1080	1082
@@E611	001	00B0	1082	1084
@@E612	001	00B1	1084	1086
@@E613	001	00B2	1086	1088
@@E614	001	00B3	1088	1090
@@E700	001	00B4	1090	1092
@@E701	001	00B5	1092	1094
@@E710	001	00B6	1094	1096
@@E712	001	00B7	1096	1098
@@E713	001	00B8	1098	1100
@@E714	001	00B9	1100	1102
@@E715	001	00BA	1102	1104
@@E716	001	00BB	1104	1106
@@E717	001	00BC	1106	1108
@@E718	001	00BD	1108	1110
@@E720	001	00BE	1110	1112
@@E721	001	00BF	1112	1114
@@E723	001	00C0	1114	1116
@@E724	001	00C1	1116	1118
@@E725	001	00C2	1118	1120
@@E726	001	00C3	1120	1122
@@E727	001	00C4	1122	1124
@@E728	001	00C5	1124	1126
@@E729	001	00C6	1126	1128
@@E730	001	00C7	1128	1130
@@E732	001	00C8	1130	1132
@@E752	001	00C9	1132	1134
@@E753	001	00CA	1134	1136
@@E754	001	00CB	1136	1138
@@E755	001	00CC	1138	1140
@@E756	001	00CD	1140	1142
@@E757	001	00CE	1142	1144
@@E758	001	00CF	1144	1146
@@E759	001	00D0	1146	1148
@@E760	001	00D1	1148	1150
@@E761	001	00D2	1150	1152
@@E762	001	00D3	1152	1154
@@E763	001	00D4	1154	1156
@@E764	001	00D5	1156	1158
@@E765	001	00D6	1158	1160
@@E766	001	00D7	1160	1162
@@E767	001	00D8	1162	1164
@@E768	001	00D9	1164	1166
@@E769	001	00DA	1166	1168
@@E770	001	00DB	1168	1170
@@E771	001	00DC	1170	1172

SYMBOL	LEN	VALUE	DEFN	REFERENCES		
@@E772	001	00DD	1172	1174		
@@E773	001	00DE	1174	1176		
@@E774	001	00DF	1176	1178		
@@E775	001	00E0	1178	1180		
@@E776	001	00E1	1180	1182		
@@E777	001	00E2	1182	1184		
@@E778	001	00E3	1184	1186		
@@E779	001	00E4	1186	1188		
@@E780	001	00E5	1188	1190		
@@E781	001	00E6	1190	1192		
@@E782	001	00E7	1192	1194		
@@E783	001	00E8	1194	1196		
@@E784	001	00E9	1196	1198		
@@E785	001	00EA	1198	1200		
@@E786	001	00EB	1200	1202		
@@E790	001	00EC	1202	1204		
@@E791	001	00ED	1204	1206		
@@E792	001	00EE	1206	1208		
@@E793	001	00EF	1208	1210		
@@E794	001	00F0	1210	1212		
@@E795	001	00F1	1212	1214		
@@E796	001	00F2	1214	1216		
@@E797	001	00F3	1216	1218		
@@E798	001	00F4	1218	1220		
@@E800	001	FFFF	1248			
@@E801	001	FFFF	1250			
@@E802	001	FFFF	1252			
@@E803	001	FFFF	1254			
@@E804	001	FFFF	1256			
@@E900	001	00F5	1220	1222		
@@E901	001	00F6	1222	1224		
@@E902	001	00F7	1224	1226		
@@E903	001	00F8	1226	1228		
@@E905	001	00F9	1228	1230		
@@E906	001	00FA	1230	1232		
@@E910	001	00FB	1232			
@ARR	001	0008	0015	3087	3270	
@ASIGN	001	007C	0070			
@ASTER	001	005C	0068			
@BCRDL	001	0050	0087			
@BE	001	0081	0042			
@BF	001	0090	0051			
@BH	001	0084	0040			
@BL	001	0082	0041			
@BLANK	001	0040	0064	3123	3275	3281
@BM	001	0082	0053			
@BNE	001	0001	0045	3266		
@BNH	001	0004	0043			
@BNL	001	0002	0044			
@BNM	001	0002	0056			
@BNOL	001	0020	0049			
@BNOZ	001	0008	0048			
@BNP	001	0004	0055			
@BNZ	001	0001	0057			
@BOL	001	00A0	0047			
@BOZ	001	0088	0046			

```
@BP      001  0084  0052
@BR      001  0001  0012    2559  2560* 2598  2603  2605  2606  2606  2612  2620  2629  2630* 2632*
                           2633* 2634  2636  2637* 2642  2643* 2645* 2646* 2647  2649  2650* 2655
                           2656  2656  2657  2657  2658* 2659* 2662* 2674  2676  2678  2682  2694
                           2694  2695  2701  2710  2710  2711  2712  2716  2718  2718  2720  2724
                           2728  2729  2731  2736  2736  2738  2740  2740  2741  2741  2750  2752
                           2753  2757  2757  2758  2758  2763  2763  2768  2773  2778  2780  2791
                           2792  2793  2794  2794  2802  2806  2812  2812* 2843  2844  2844* 2904*
                           2905  2905  2906  2906  2910  2910  2911  2911  2912  2912  2913  2913
                           2914  2914  2915  2915  2921  2921  2922  2922  2923  2923  2924  2924
                           2925  2925  2926  2926  2927  2927  2928  2928  2929  2929  2931  2931
                           2945  2946  2947  2947  2948  2949  2949* 2960  2962  2962* 2988* 2989
                           2997* 3002  3019  3032* 3036  3082  3084  3085* 3087  3089  3091  3091
                           3101  3101  3106  3106  3107  3107  3108  3108  3109  3109  3110  3110
                           3114  3115  3115  3118  3124  3125  3130  3131  3131  3133*
@BT      001  0010  0050
@BZ      001  0081  0054
@B1      001  0001  0062    3117  3122
@CADDR   001  0002  0141    2103  2104  2105  2989
@CARDL   001  0060  0086    0643
@CHARA   001  00C1  0071
@CHARF   001  00C6  0072
@CHARR   001  00D9  0073
@CHARZ   001  00E9  0074
@CLOFF   001  0010  0093
@CLON    001  0011  0092
@COMMA   001  006B  0065    3277
@CPLUS   001  004E  0078
@DADDR   001  0002  0139
@DBFR1   001  0004  0128
@DBFR2   001  0005  0129
@DCALK   001  0001  0080
@DCBCY   001  0009  0114    1932
@DCBT1   001  0050  0116    1935
@DCNT    001  0003  0127
@DCST1   001  0040  0115    1933
@DCTRL   001  0000  0124    3044*
@DCYL    001  0001  0125
@DD2     001  0003  0029    2592* 2593  2973* 2974  3016* 3017* 3018
@DGET    001  0001  0133
@DOLAR   001  005B  0067
@DOP2    001  0004  0027
@DPLNG   001  0006  0131
@DPOS    001  0000  0132
@DPUT    001  0002  0134    3044
@DSAD    001  0002  0126
@DSBCY   001  0004  0105    1870
@DSCS1   001  0000  0106    1871
@DSIVF   001  0003  0137
@DSPIN   001  0002  0130
@DTRSZ   001  0018  0084
@DVBCY   001  0007  0107    1929
@DVRFY   001  0031  0135
@DWAIT   001  00FF  0136
@DWBCY   001  0005  0102    1926
@DWSIZ   001  00C0  0104
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES
@DWTB1	001	0003	0103	1927
@DZERO	001	00F0	0063	
@D1	001	0002	0025	2593* 2974* 3018* 3025* 3101
@EOF	001	001C	0076	
@EOFTC	001	0075	0161	
@EOS	001	001E	0075	1942 3283
@FDDBC	001	0000	0194	
@FDE1	001	000C	0199	
@FDFNA	001	000B	0197	
@FDHLN	001	0002	0207	
@FDLNC	001	0002	0192	
@FDNSC	001	0003	0209	
@FDSD	001	0000	0205	
@FLACE	001	0009	0196	
@FLDBC	001	0001	0195	
@FLENT	001	0004	0200	
@FLFNA	001	0002	0198	
@FLHLN	001	0002	0208	
@FLLNC	001	0002	0193	
@FLNSC	001	0001	0210	
@FLSD	001	0001	0206	
@HDRLN	001	0007	0091	0671
@IAR	001	0010	0016	
@INDEX	001	0001	0155	0156
@INST3	001	0003	0031	
@INST4	001	0004	0032	
@INST5	001	0005	0033	
@INST6	001	0006	0034	
@I1IAR	001	00C0	0019	
@LINSZ	001	00F4	0083	0645
@MAPEN	001	0005	0088	
@MINCR	001	2000	0082	
@MINUS	001	0060	0079	
@NOP	001	0080	0039	3096 3170
@NUMBR	001	007B	0069	
@OPD2	001	0004	0028	
@OP1	001	0003	0026	2709* 3084* 3087* 3270*
@OP2	001	0005	0030	
@PCTRL	001	0000	0148	
@PDATA	001	0003	0150	
@PGCSZ	001	0020	0081	0082
@PPLNG	001	0004	0147	
@PRCNT	001	0001	0149	
@PRETR	001	00C0	0153	
@PRINT	001	0040	0151	0153
@PSR	001	0004	0014	
@PWAIT	001	00FF	0157	
@P1IAR	001	0020	0017	
@P2IAR	001	0040	0018	
@Q	001	0001	0023	2591* 2868* 2878 2972* 3009 3019 3023 3023* 3024* 3025 3036 3037 3164 3168 3289
@REGL	001	0002	0011	
@RETRN	001	0080	0152	0153
@RLDWN	001	004F	0158	
@RTRNC	001	0080	0160	
@SBLN	001	0005	0169	

```
@SBLNL   001  0002 0183
@SCTSZ   001  0100 0099
@SDFLN   001  0007 0089
@SDF0    001  0000 0165
@SDF1    001  0001 0166
@SDF2    001  0002 0167
@SDF3    001  0003 0168
@SECCY   001  0030 0085
@SIST    001  0001 0180
@SLASH   001  0061 0066
@SLAST   001  0002 0182
@SMIDL   001  0003 0181
@SNULL   001  0080 0172
@SONLY   001  0000 0179
@STEXT   001  0007 0171
@STYPE   001  0006 0170
@TBCNT   001  0000 0159
@TBLEF   001  0010 0154   0156
@TBLIX   001  0011 0156
@UCB     001  0087 0038   3166   3267   3278
@UPARW   001  005A 0077
@VADDR   001  0002 0140   1663   2099   2111   2112   2113   2113   2127   2130   2132   2156   2157   2158
                         2196   2199   2202   2205   2208   2211   2214   2223   2226   2229   2232   2235
                         2578   2581   2634   2636   2647   2649   2763
@VENTA   001  0056 0112   1930   2185
@VMDDV   001  00FE 0113
@VMFD1   001  0000 0108
@VMFD2   001  0001 0109
@VMRS3   001  0002 0111
@VMTRL   001  0001 0110
@VOLID   001  0006 0090
@VQ      001  0001 0024
@WSFIT   001  0500 0100
@WSTBL   001  0503 0101
@XR      001  0002 0013   2579   2582   2597*  2598   2599*  2603   2605*  2611*  2612*  2613   2618   2620*
                         2624   2655   2663   2663*  2682*  2683   2688   2699   2703   2709   2711*  2712*
                         2720   2724   2728   2729   2731   2738   2750   2753   2767*  2768*  2773*  2798
                         2800   2806*  2807   2813   2818   2818*  2819   2828   2843   2848   2848*  2849
                         2851   2853   2853*  2858   2860   2860*  2864   2866   2869   2869*  2951   2951*
                         2952   2954   2954*  2955   2960   2961   2961*  2998*  3004   3019   3036   3073*
                         3089   3098   3114   3117   3117*  3122   3122*  3123   3130   3271   3274   3274*
                         3275   3277   3280   3280*  3281   3283   3285
@ZERO    001  0000 0061   2791
B$ADMK   001  0001 1567
B$ADSW   001  159D 1566
B$ARMK   001  0001 1552
B$ARSW   001  0A45 1551
B$BABF   001  1D00 1357
B$BCKT   001  1590 1479
B$BDPL   001  19E8 1431
B$BDSA   001  19EA 1432
B$BINO   001  1A6A 1495
B$BRLN   001  19F1 1430
B$BROP   001  1AF7 1536
B$BRVA   001  19EF 1429
B$BRVP   001  19EE 1428
```

```
B$BTAB   001  1996 1427
B$CADR   001  1AF9 1537
B$CASA   001  0000 1372
B$CASC   001  0671 1376
B$CASM   001  0608 1374
B$CBAS   001  14BB 1502
B$CBFA   001  0CBC 1457
B$CCGT   001  0600 1382
B$CCLS   001  0695 1388
B$CCON   001  001F 1455
B$CDAT   001  0600 1368
B$CDEF   001  0600 1369
B$CDIM   001  0673 1370
B$CDUM   001  0000 1406
B$CEND   001  0600 1404   1405
B$CEOF   001  0600 1405
B$CFOR   001  0600 1377
B$CGET   001  06A3 1385
B$CGSB   001  0690 1383
B$CGTO   001  06B3 1381
B$CIFA   001  0600 1379
B$CIFC   001  0600 1380
B$CIMG   001  0600 1394
B$CINP   001  0600 1389
B$CLTA   001  0000 1371
B$CLTC   001  0669 1375
B$CLTM   001  0600 1373
B$CMAT   001  0600 1395
B$CMGT   001  0665 1396
B$CMIN   001  06D3 1397
B$CMPR   001  069B 1400
B$CMPT   001  069B 1399
B$CMPU   001  0600 1401
B$CMRD   001  06D0 1398
B$CNXT   001  0600 1378
B$CPCT   001  0CA8 1460
B$CPRT   001  0600 1392
B$CPRU   001  0600 1393
B$CPSE   001  06E7 1402
B$CPUT   001  0600 1386
B$CPWA   001  0CA6 1531
B$CRAD   001  150D 1501
B$CRBS   001  1509 1503
B$CREA   001  06CF 1390
B$CREM   001  0000 1367
B$CRMK   001  0001 1579
B$CRSR   001  06E3 1391
B$CRST   001  06A6 1387
B$CRSW   001  0E42 1578
B$CRTN   001  06CF 1384
B$CSBF   001  0600 1354   1368  1369  1370  1373  1374  1375  1376  1377  1378  1379  1380  1381
                         1382  1383  1384  1385  1386  1387  1388  1389  1390  1391  1392  1393
                         1394  1395  1396  1397  1398  1399  1400  1401  1402  1403  1404  1407
                         1408  1409  1410  1411
B$CSCN   001  14B0 1476
B$CSMK   001  0007 1582
```

SYMBOL	LEN	VALUE	DEFN
B$CSSW	001	14BC	1581
B$CSTP	001	06D6	1403
B$CSTR	001	14CC	1500
B$CSXA	001	2000	1360
B$CTYP	001	0A5F	1454
B$CVPD	001	0C5D	1459
B$CVPG	001	0CA5	1458
B$CWRK	001	F500	1528
B$DIST	001	0700	1420
B$DLNK	001	1B37	1526
B$DL4T	001	1A6B	1497
B$DPWA	001	0E46	1532
B$DST2	001	073A	1421
B$ERMK	001	0007	1555
B$ERSW	001	0993	1554
B$FACA	001	0E53	1463
B$FAIS	001	15AC	1480
B$FAIW	001	15A0	1481
B$FCON	001	0A46	1453
B$FORT	001	1B0E	1522
B$FPWA	001	15AC	1533
B$FRMK	001	0007	1573
B$FRSW	001	16CC	1572
B$FSC1	001	0E4C	1464
B$FSC2	001	0E4D	1465
B$FSMK	001	0007	1564
B$FSSW	001	0E5C	1563
B$FSVA	001	0E4F	1466
B$FTND	001	1B0B	1524
B$FTPT	001	1B0D	1523
B$FVME	001	15A2	1485
B$FVMP	001	15A4	1486
B$FVMS	001	15A6	1487
B$FVPE	001	15A8	1482
B$FVPP	001	15AA	1483
B$FVPS	001	15AC	1484
B$GBSW	001	08AF	1557
B$GBWK	001	0001	1558
B$GETC	001	0867	1434
B$GPTR	001	0878	1436
B$GTBF	001	1E00	1358
B$IFMK	001	0007	1576
B$IFSW	001	16E5	1575
B$INVT	001	1B38	1516
B$KWMK	001	0001	1570
B$KWSW	001	159E	1569
B$LBAS	001	185E	1507
B$LBSV	001	18E7	1505
B$LDRP	001	1A00	1355
B$LINE	001	07D0	1422
B$LIST	001	1853	1489
B$LRTN	001	18EB	1506
B$LSTR	001	1862	1504
B$LTYP	001	18F2	1490
B$MATR	001	18F3	1492
B$MBMK	001	0007	1591

```
B$MBSW   001  1903 1590
B$MFBK   001  1B8F 1518
B$MGMK   001  0007 1588
B$MGSW   001  18FF 1587
B$MPMK   001  0007 1594
B$MPSW   001  1981 1593
B$MRMK   001  0007 1585
B$MRSW   001  0DDE 1584
B$NUMC   001  0873 1435
B$NXMK   001  0007 1561
B$NXSW   001  071D 1560
B$PARP   001  0A41 1443
B$PBNL   001  0A01 1449
B$PCAD   001  0A40 1444
B$PCDL   001  09D3 1448
B$PCPG   001  0A35 1447
B$PECT   001  0A44 1451
B$PERC   001  0A39 1450
B$PFAE   001  0033 1441
B$PFCL   001  009D 1442
B$PFNC   001  094E 1439
B$PFWP   001  0015 1440
B$PNBY   001  0A41 1445
B$PPWA   001  0A35 1530
B$PRM1   001  1AF3 1534
B$PTBF   001  1F00 1359
B$PUTC   001  093A 1438
B$PVAD   001  0A43 1446
B$RMRK   001  1AE6 1499
B$RTRN   001  1AF5 1535
B$SABF   001  1C00 1356
B$SCAN   001  1514 1478
B$SCAT   001  13C8 1473
B$SCON   001  001B 1456
B$SCVT   001  12E0 1471
B$SDPL   001  07DA 1424
B$SFAB   001  0E48 1468
B$SFNT   001  143C 1474
B$SLDT   001  109C 1470
B$SLVT   001  1062 1469
B$SNAT   001  131A 1472
B$SPAT   001  07E0 1425
B$SSTA   001  1BAC 1520
B$STAS   001  061B 1409
B$STIF   001  0606 1411
B$STMA   001  061B 1410
B$STML   001  0600 1408
B$STRL   001  0600 1407
B$SVRB   001  0E46 1467
B$SYMB   001  0DBC 1462
B$TCD2   001  0001 1540
B$TLTH   001  0002 1541   1542
B$TOD1   001  0000 1539
B$TOTB   001  1AF8 1542
B$TTAB   001  1AFA 1538   1542
B$TYPE   001  0739 1423
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES
B$WORK	001	15A0	1527	
B$ZDBN	001	19F2	1494	
B@ABAS	001	0007	2127	2728
B@ACD1	001	0001	2124	2125 2720 2724 2729
B@ACD2	001	0003	2125	2126 2731 2738
B@AFLG	001	0000	2119	
B@ALLA	001	005C	1944	
B@AMAX	001	0005	2126	2127
B@BLNK	001	0040	1953	2946
B@BLSZ	001	0100	2078	2217 2220 2223 2238 2241
B@BREQ	001	0084	1733	
B@BRHI	001	0088	1734	
B@BRLO	001	0082	1732	
B@BRNE	001	0094	1736	
B@BRNH	001	0098	1737	
B@BRNL	001	0092	1735	
B@CADD	001	0006	1602	
B@CADF	001	0058	1643	
B@CBAS	001	0003	2130	2753
B@CBNX	001	004A	1636	
B@CBRA	001	0046	1634	
B@CBRC	001	0044	1633	
B@CBRD	001	0048	1635	
B@CBRS	001	004C	1637	
B@CCLS	001	005E	1646	
B@CCMC	001	0042	1632	
B@CCMF	001	0040	1631	
B@CCNT	001	001F	2056	
B@CCSA	001	003E	1630	
B@CDCA	001	006A	1652	
B@CDDL	001	006C	1653	
B@CDIV	001	000C	1605	
B@CDMN	001	0001	2129	2130 2750
B@CDWA	001	006E	1654	
B@CEOF	001	0070	1655	
B@CEOP	001	0068	1651	
B@CFCI	001	0016	1610	
B@CFN0	001	0012	1608	
B@CFN1	001	0014	1609	
B@CFOR	001	004E	1638	
B@CGET	001	0052	1640	
B@CHAR	001	0000	2069	
B@CHLT	001	0004	1601	
B@CIEX	001	00C5	2029	
B@CIMH	001	0066	1650	
B@CINI	001	0056	1642	
B@CIPI	001	00D7	2032	
B@CIS2	001	00E2	2035	
B@CMF1	001	0018	1611	
B@CMF2	001	001A	1612	
B@CMF3	001	001C	1613	
B@CMMA	001	006B	1964	2703
B@CMPY	001	000A	1604	
B@CMSM	001	001E	1614	
B@CNEG	001	0010	1607	
B@CNXT	001	0050	1639	

SYMBOL	LEN	VALUE	DEFN	REFERENCES			
B@COLN	001	007A	1966				
B@CPMK	001	00FF	1874	1878	1882	1883	1917
B@CPRS	001	0060	1647				
B@CPRU	001	0062	1648				
B@CPUT	001	0054	1641				
B@CPWR	001	000E	1606				
B@CRSR	001	005A	1644				
B@CRST	001	005C	1645				
B@CSA1	001	0036	1626				
B@CSA2	001	0038	1627				
B@CSB1	001	003A	1628				
B@CSC1	001	002A	1620				
B@CSD0	001	002E	1622				
B@CSD1	001	0030	1623				
B@CSD2	001	0032	1624				
B@CSF1	001	0022	1616				
B@CSF2	001	0024	1617				
B@CSTA	001	0034	1625				
B@CSTC	001	0028	1619				
B@CSTF	001	0020	1615				
B@CSTH	001	0064	1649				
B@CSTX	001	003C	1629				
B@CSUB	001	0008	1603				
B@CSVC	001	0002	1600				
B@CTYP	001	0020	2054				
B@CUSC	001	002C	1621				
B@CUSF	001	0026	1618				
B@CVAR	001	005B	1943	2618			
B@DAMK	001	0080	2122				
B@DASA	001	00FF	1883				
B@DASC	001	0040	1887				
B@DASM	001	0038	1885				
B@DCGT	001	0050	1893				
B@DCLS	001	0054	1899				
B@DDAT	001	0024	1879				
B@DDEF	001	0034	1880				
B@DDIM	001	0004	1881				
B@DDUM	001	00FF	1917				
B@DEC0	001	00F0	2012	2613 2683 2793 2807 2819 2828 2849 2897			
B@DEC1	001	00F1	2013	2688			
B@DEC2	001	00F2	2014				
B@DEC3	001	00F3	2015				
B@DEC4	001	00F4	2016				
B@DEC5	001	00F5	2017				
B@DEC6	001	00F6	2018				
B@DEC7	001	00F7	2019				
B@DEC8	001	00F8	2020				
B@DEC9	001	00F9	2021				
B@DEND	001	0058	1915	1916			
B@DEOF	001	0058	1916				
B@DFOR	001	0028	1888				
B@DGET	001	0040	1896				
B@DGSB	001	0020	1894				
B@DGTO	001	0044	1892				
B@DIFA	001	0048	1890				
B@DIFC	001	004C	1891				

```
B@DIGS   001  007B 1946
B@DIMG   001  003C 1905
B@DINP   001  0000 1900
B@DIVD   001  0061 1963
B@DLTA   001  00FF 1882
B@DLTC   001  0040 1886
B@DLTM   001  0038 1884
B@DL01   001  0001 2197   2200
B@DL02   001  0003 2200   2203
B@DL03   001  0005 2203   2206
B@DL04   001  0007 2206   2209
B@DL05   001  0009 2209   2212
B@DL06   001  000B 2212   2215
B@DL07   001  0045 2215   2218
B@DL08   001  0145 2218   2221
B@DL09   001  0245 2221   2224
B@DL10   001  0289 2224   2227
B@DL11   001  02C3 2227   2230
B@DL12   001  02FD 2230   2233
B@DL13   001  0337 2233   2236
B@DL14   001  0371 2236   2239
B@DL15   001  0471 2239   2242
B@DL16   001  0507 2242
B@DMAT   001  0008 1906
B@DMGT   001  0044 1907
B@DMIN   001  0038 1908
B@DMPR   001  0048 1911
B@DMPT   001  004C 1910
B@DMPU   001  0054 1912
B@DMRD   001  003C 1909
B@DNXT   001  0044 1889
B@DPNT   001  004B 1954   2813   2851
B@DPRT   001  002C 1903
B@DPRU   001  0030 1904
B@DPSE   001  0050 1913
B@DPUT   001  0040 1897
B@DREA   001  000C 1901
B@DREM   001  00FF 1878
B@DRSR   001  005C 1902
B@DRST   001  0050 1898
B@DRTN   001  005C 1895
B@DSCY   001  0004 1870
B@DSIF   001  001C 1919
B@DSLT   001  0010 1918
B@DSML   001  0010 1920
B@DSNS   001  0018 1872
B@DSS1   001  0000 1871
B@DSTP   001  0054 1914
B@DTBN   001  0010 1936
B@DTB1   001  0050 1935
B@DTCY   001  0009 1932
B@DTSN   001  0010 1934
B@DTS1   001  0040 1933
B@DTYP   001  0040 2048   2945
B@DVCY   001  0007 1929
B@DVC1   001  0056 1930
```

```
B@DWCY  001  0005 1926
B@DWT1  001  0003 1927
B@D1MK  001  0080 2120
B@D2MK  001  00C0 2121
B@EOST  001  001E 1942
B@EQUL  001  007E 1968
B@EXPC  001  00C5 1945   2858
B@FOFL  001  005C 1947
B@FVAD  001  0001 2132
B@GETC  001  0001 2071
B@GETE  001  00FF 2072
B@GETS  001  0000 2070
B@GRTR  001  006E 1965
B@ICON  001  0050 2027
B@LADD  001  0001 1671
B@LADF  001  0002 1712
B@LADV  001  0008 2156   2177
B@LBIN  001  0002 2081   2082  2088  2718  2720  2724
B@LBNX  001  0003 1705
B@LBRA  001  0003 1703
B@LBRC  001  0004 1702
B@LBRD  001  0003 1704
B@LBRS  001  0001 1706
B@LCCA  001  0004 2112
B@LCCC  001  0001 1664   1702
B@LCDV  001  0004 2157   2178
B@LCER  001  0001 1662   1726
B@LCFN  001  0004 2113
B@LCLN  001  0002 1667   1718  1719  1726
B@LCLS  001  0001 1715
B@LCMC  001  0001 1701
B@LCMF  001  0001 1700
B@LCNA  001  0006 2111
B@LCNN  001  0001 1665   1690  1699  1711  1723
B@LCOP  001  0001 1661   1669  1670  1671  1672  1673  1674  1675  1676  1677  1678  1679  1680
                         1681  1682  1683  1684  1685  1686  1687  1688  1689  1690  1691  1692
                         1693  1694  1695  1696  1697  1698  1699  1700  1701  1702  1703  1704
                         1705  1706  1707  1708  1709  1710  1711  1712  1713  1714  1715  1716
                         1717  1718  1719  1720  1721  1722  1723  1724
B@LCRV  001  0013 2155   2175  2752  2948  2972  2973
B@LCSA  001  0002 1699
B@LCVA  001  0002 1663   1677  1678  1679  1680  1681  1682  1683  1684  1685  1686  1688  1689
                         1691  1692  1693  1694  1695  1696  1697  1702  1703  1704  1705  1707
                         1708  1709  1721  1722
B@LCXX  001  0001 1666   1698  1710  1712  1716  1717
B@LDAT  001  0004 1825
B@LDCA  001  0003 1721
B@LDDL  001  0003 1722
B@LDDM  001  0004 2085
B@LDEF  001  0003 1826
B@LDIM  001  0003 1827
B@LDIN  001  0004 2084   2085  2086
B@LDIV  001  0001 1674
B@LDMN  001  0002 2082   2111  2112  2124  2125  2126  2129  2156  2157  2694  2695  2728  2729
                         2731  2736  2738  2740  2750  2753  2757
B@LDSN  001  0004 2086
```

```
B@LDWA   001  0002 1723
B@LELP   001  0010 2154   2794   2947
B@LEND   001  0003 1854
B@LEOF   001  0001 1724
B@LEOP   001  0001 1720
B@LERC   001  0003 1726
B@LESP   001  0008 2153
B@LESS   001  004C 1955
B@LET$   001  005B 1975
B@LET#   001  007B 1976
B@LET@   001  007C 1977
B@LETA   001  00C1 1979
B@LETB   001  00C2 1981
B@LETC   001  00C3 1982
B@LETD   001  00C4 1983
B@LETE   001  00C5 1984
B@LETF   001  00C6 1985
B@LETG   001  00C7 1986
B@LETH   001  00C8 1987
B@LETI   001  00C9 1988
B@LETJ   001  00D1 1989
B@LETK   001  00D2 1990
B@LETL   001  00D3 1991
B@LETM   001  00D4 1992
B@LETN   001  00D5 1993
B@LETO   001  00D6 1994
B@LETP   001  00D7 1995
B@LETQ   001  00D8 1996
B@LETR   001  00D9 1997
B@LETS   001  00E2 1998
B@LETT   001  00E3 1999
B@LETU   001  00E4 2000
B@LETV   001  00E5 2001
B@LETW   001  00E6 2002
B@LETX   001  00E7 2003
B@LETY   001  00E8 2004
B@LETZ   001  00E9 2005
B@LEXP   001  0008 2044
B@LFCI   001  0003 1679
B@LFNA   001  0002 2158   2179
B@LFN0   001  0003 1677
B@LFN1   001  0003 1678
B@LFOR   001  0003 1707
B@LFRT   001  0004 2099   2100
B@LGET   001  0003 1709
B@LGSB   001  0005 1833
B@LGTO   001  0004 1832
B@LHLT   001  0001 1670
B@LIEX   001  0002 2030
B@LIFN   001  0003 2093
B@LILP   001  0009 2152   2170   2171   2172   2591   2592
B@LIMG   001  0001 1844
B@LIMH   001  0003 1719
B@LINI   001  0002 1711
B@LINP   001  0005 1839
B@LIPI   001  0003 2033
```

```
B@LISP  001  0005 2151    2159   2165   2166   2167   3010   3038   3039   3040
B@LIS2  001  0005 2036
B@LIVT  001  0001 2109
B@LKCL  001  0005 1838
B@LKFR  001  0003 1829
B@LKGT  001  0003 1835
B@LKIF  001  0002 1831
B@LKON  001  0002 1864
B@LKPT  001  0003 1836
B@LKPU  001  000A 1843
B@LKRR  001  0007 1841
B@LKRT  001  0005 1837
B@LKTO  001  0002 1858
B@LLET  001  0003 1828
B@LL01  001  0002 2196    2197
B@LL02  001  0002 2199    2200
B@LL03  001  0002 2202    2203
B@LL04  001  0002 2205    2206
B@LL05  001  0002 2208    2209
B@LL06  001  0002 2211    2212
B@LL07  001  003A 2214    2215
B@LL08  001  0100 2217    2218
B@LL09  001  0100 2220    2221
B@LL10  001  0044 2223    2224
B@LL11  001  003A 2226    2227
B@LL12  001  003A 2229    2230
B@LL13  001  003A 2232    2233
B@LL14  001  003A 2235    2236
B@LL15  001  0100 2238    2239
B@LL16  001  0096 2241    2242
B@LMAT  001  0003 1845
B@LMF1  001  0003 1680
B@LMF2  001  0003 1681
B@LMF3  001  0003 1682
B@LMGT  001  0006 1846
B@LMIN  001  0008 1847
B@LMPR  001  0008 1850
B@LMPT  001  0006 1849
B@LMPU  001  000D 1851
B@LMPY  001  0001 1673
B@LMRD  001  0007 1848
B@LMSM  001  0003 1683
B@LNEG  001  0001 1676
B@LNEX  001  0004 1830
B@LNXT  001  0003 1708
B@LPAR  001  004D 1956    2624
B@LPRS  001  0002 1716
B@LPRT  001  0005 1842
B@LPRU  001  0002 1717
B@LPSE  001  0005 1852
B@LPUT  001  0002 1710
B@LPWR  001  0001 1675
B@LREA  001  0004 1840
B@LREM  001  0003 1824
B@LRSR  001  0001 1713
B@LRST  001  0001 1714
```

```
B@LRTN  001  0006 1834
B@LSA1  001  0003 1695
B@LSA2  001  0003 1696
B@LSB1  001  0003 1697
B@LSC1  001  0003 1689
B@LSDF  001  0004 2079
B@LSD0  001  0003 1691
B@LSD1  001  0003 1692
B@LSD2  001  0003 1693
B@LSF1  001  0003 1685
B@LSF2  001  0003 1686
B@LSKW  001  0002 2095
B@LSNO  001  0002 2088
B@LSPT  001  0003 2103   2106
B@LSTA  001  0003 1694
B@LSTC  001  0003 1688
B@LSTE  001  0004 1859
B@LSTF  001  0003 1684
B@LSTH  001  0003 1718
B@LSTP  001  0004 1853
B@LSTX  001  0002 1698
B@LSUB  001  0001 1672
B@LSVC  001  0001 1669
B@LTHN  001  0004 1860
B@LTYP  001  0001 2089
B@LUFN  001  0002 2096
B@LUSC  001  0002 1690
B@LUSF  001  0001 1687
B@LVPG  001  0100 2183   2186   2582
B@MINS  001  0060 1962   2800   2866
B@MULT  001  005C 1959
B@NAAR  001  001D 2147   2177   2229
B@NCAR  001  001D 2148   2178   2232
B@NCRV  001  001D 2146   2175   2226
B@NDGT  001  000A 2139   2145
B@NEQL  001  007F 1969
B@NFRT  001  000A 2098   2100
B@NICN  001  0006 2141   2143
B@NIEL  001  0007 2143   2159   2165   2170
B@NIFN  001  0018 2092
B@NIVR  001  0001 2142   2143
B@NIVT  001  0057 2108
B@NLDV  001  0122 2145   2167   2172   2223
B@NLRV  001  001D 2144   2166   2171   2214
B@NLTR  001  001D 2138   2144   2145   2146   2147   2148   2149
B@NSKW  001  0004 2094
B@NSPT  001  0028 2102
B@NUFN  001  001D 2149   2179   2235
B@NVPG  001  0100 2182   2186
B@NXHI  001  00E3 2063   2892
B@NXLO  001  001E 2062   2890   2900
B@NXZR  001  0080 2061   2062   2063   2792
B@PLUS  001  004E 1957   2798   2864
B@POWR  001  005A 1958
B@PREC  001  0020 2050
B@PROD  001  0023 2159
```

```
B@PRPL   001  0002 1746
B@PRPN   001  0001 1745
B@PRPR   001  0004 1748
B@PRPS   001  0003 1747
B@PRRC   001  0007 1751
B@PRRL   001  0008 1752
B@PRSL   001  0005 1749
B@PRSS   001  0006 1750
B@PTAB   001  0000 2104
B@PTAD   001  0001 2105
B@PTSA   001  0002 2106
B@PUD1   001  0006 1762
B@PUD2   001  0007 1763
B@PUI0   001  0001 1756
B@PUI1   001  0004 1757
B@PUI2   001  0005 1758
B@PUNL   001  0002 1760
B@PUNS   001  0003 1761
B@PURE   001  0020 1766
B@PUTM   001  0010 1765
B@RPAR   001  005D 1960   2699
B@SADV   001  00E8 2177   2180
B@SAVL   001  0B76 2173   2190
B@SAVS   001  065E 2168   2189
B@SCDV   001  0074 2178   2180
B@SCLN   001  005E 1961
B@SCRV   001  0227 2175   2189  2190
B@SDMK   001  0080 2090
B@SEXP   001  0004 2043
B@SFAT   001  0196 2180   2189  2190  2241
B@SFNA   001  003A 2179   2180
B@SFRT   001  0028 2100
B@SIEL   001  003F 2170   2173
B@SIES   001  0023 2165   2168
B@SIGN   001  0010 2052   2802
B@SLDL   001  0A32 2172   2173
B@SLDS   001  05AA 2167   2168
B@SLVL   001  0105 2171   2173
B@SLVS   001  0091 2166   2168
B@SQUO   001  007D 1967   2952  2955
B@STAT   001  0000 2042
B@TASA   001  0012 1777
B@TASC   001  001E 1783
B@TASM   001  0018 1779
B@TASS   001  007B 1784
B@TCGT   001  0030 1792
B@TCLS   001  0042 1798
B@TDAT   001  0006 1773
B@TDEF   001  0009 1774
B@TDIM   001  000C 1775
B@TDUM   001  0078 1816
B@TEND   001  0072 1814
B@TEOF   001  0075 1815
B@TFOR   001  0021 1786
B@TGET   001  0039 1795
B@TGSB   001  0033 1793
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES
B@TGTO	001	002D	1791	
B@TIFA	001	0027	1788	
B@TIFC	001	002A	1789	
B@TIFS	001	007D	1790	
B@TIMG	001	0054	1804	
B@TINP	001	0045	1799	
B@TLTA	001	000F	1776	
B@TLTC	001	001B	1780	
B@TLTM	001	0015	1778	
B@TLTS	001	0079	1781	
B@TMAS	001	007C	1785	
B@TMAT	001	0057	1805	
B@TMGT	001	005A	1806	
B@TMIN	001	005D	1807	
B@TMLS	001	007A	1782	
B@TMPR	001	0066	1810	
B@TMPT	001	0063	1809	
B@TMPU	001	0069	1811	
B@TMRD	001	0060	1808	
B@TNXT	001	0024	1787	
B@TPRT	001	004E	1802	
B@TPRU	001	0051	1803	
B@TPSE	001	006C	1812	
B@TPUT	001	003C	1796	
B@TRAC	001	0080	2046	3002 3004
B@TREA	001	0048	1800	
B@TREM	001	0003	1772	
B@TRSR	001	004B	1801	
B@TRST	001	003F	1797	
B@TRTN	001	0036	1794	
B@TSTP	001	006F	1813	
B@VMC1	001	0056	2185	
B@VMLB	001	F0CD	2190	
B@VMSB	001	F5E5	2189	
B@VMSZ	001	0000	2186	2188 2189 2190
B@VMTB	001	0000	2188	
B@ZNEG	001	00D0	2059	
B@ZPOS	001	00F0	2058	
C4BCHC	001	0004	3158	
C4BCHR	001	109B	3146	3114* 3115
C4BINI	001	109A	3144	3091
C4BIN2	001	102F	3081	2693 2873 3082 3085
C4BLEN	002	1097	3156	3130* 3131*
C4BLNK	003	104A	3164	
C4BLOW	001	00F0	3160	3098
C4BLVL	002	0002	3162	3091 3106 3107 3108 3109 3110 3115
C4BNMC	004	1046	3168	
C4BNOP	001	0080	3170	
C4BSAV	002	109D	3150	3089* 3131
C4BSPC	001	0087	3166	
C4BVAL	002	1099	3142	2695 2882 2885 3091* 3106 3106* 3107 3108 3108* 3109 3109* 3110*
				3115* 3162
C4BWRK	002	1097	3139	3107* 3110 3156 3162
C4BYT1	001	1098	3141	
C4B100	004	1045	3092	3168
C4B200	003	1049	3096	3118 3164

```
C4B300   003  104C 3098   3124
C4B590   003  107B 3122   3101   3125
C4B600   003  107E 3123   3096
C4B700   003  1087 3130   3099
C4B800   004  108E 3133   3084* 3102
C4B850   004  1092 3135   3087*
C4B900   001  109E 3152   3092* 3101*
C4END    001  109F 3171
I$ADJX   001  0D56 2319
I$ADST   001  0C9D 2274
I$BASE   001  0C60 2276
I$BRCN   001  117B 2328
I$BSET   001  119D 2327
I$B1SW   001  0040 2384
I$B2SW   001  0020 2386
I$CADR   001  144C 2365   2997   3032
I$CALL   001  12B1 2359
I$CBM1   001  0D43 2295
I$CBN1   001  0D3E 2291
I$CBN2   001  0D3F 2292
I$CBN3   001  0D40 2293
I$CBN4   001  0D41 2294
I$CFBS   001  0AE3 2342
I$CLFA   001  0D4A 2301
I$CLVA   001  0D49 2300
I$CL1C   001  0D46 2298
I$CL1F   001  0D44 2296
I$CL2C   001  0D47 2299
I$CL2F   001  0D45 2297
I$CPG1   001  1600 2256
I$CPUF   001  0A27 2338
I$CSCT   001  0D5A 2314
I$CSSW   001  0010 2388
I$CSXA   001  2000 2255
I$CUPF   001  0A85 2340
I$CVAD   001  1358 2353
I$DATA   001  0D53 2282
I$DAT1   001  0D55 2283
I$DMSW   001  0BC1 2336
I$ECSW   001  0004 2392
I$ERRC   001  0CBC 2281
I$FACT   001  0DD1 2321
I$FADD   001  075D 2344
I$FATE   001  0DE6 2322
I$FATP   001  0DE8 2323
I$FDVD   001  0919 2349
I$FMPY   001  082A 2347
I$FSUB   001  0751 2345
I$FWRK   001  0607 2265
I$IMC1   001  0DCE 2312
I$IMLN   001  0DC6 2308
I$IMPT   001  0DCC 2311
I$INDR   001  0DC5 2307
I$INIT   001  0607 2264
I$INTR   001  0C5C 2268
I$IRSW   001  0CDE 2288
```

```
I$I700  001  0E24 2350
I$LBFR  001  12B6 2360
I$LDBR  001  1329 2357
I$LDXR  001  1330 2358    2577   2580
I$LOCK  001  1354 2355
I$MDFY  001  1349 2354    2993   3031
I$MOD4  001  130B 2351
I$NCPG  001  000A 2376
I$NDSW  001  0002 2394
I$NISW  001  0080 2382
I$NPAG  001  0C68 2269
I$PARM  001  0D57 2284
I$PGDS  001  144A 2363
I$PGNO  001  1449 2362
I$PGTB  001  14CA 2366
I$PLRT  001  15E2 2367
I$PSTK  001  15CA 2368
I$PUB1  001  0DC8 2309
I$PUB2  001  0DCA 2310
I$RESW  001  0CE9 2289
I$RNMK  001  0001 2304
I$RNSW  001  0D5C 2303
I$RTRN  001  12D3 2361
I$SDCT  001  0D59 2316
I$SDPT  001  0DD0 2313
I$SFCT  001  0D5A 2317
I$SFFO  001  0D5D 2325
I$SICT  001  0D5B 2318
I$SLLC  001  0BA1 2332
I$SLNG  001  0BA2 2331
I$SNSW  001  0001 2396
I$SSCT  001  0D58 2315
I$STAK  001  0D4E 2277
I$STCK  001  0B50 2330
I$STHA  001  0D51 2287
I$STKB  001  0639 2266
I$STKI  001  0D4F 2278
I$STSW  001  0008 2390
I$TFSW  001  0D28 2290
I$ULNG  001  0C3A 2335
I$UNLK  001  1350 2356
I$USTK  001  0BB0 2334
I$VADR  001  144A 2364    2989* 3008   3017   3029* 3030*
I$WRK1  001  0D59 2285
I$WRK2  001  0D5B 2286
I$XAD1  001  0C89 2273
I$XAD2  001  0C82 2272
I$XAD3  001  0C7B 2271
I$XAD4  001  0C74 2270
I$XERR  001  0CAB 2275
I$XIAR  001  0D4C 2280
I$XPAG  001  0C61 2279
KSOAAC  001  112B 3334    2712
KSOATB  001  11A7 3367    2599
KSOATI  001  1187 3368    2606* 2633   2646   2657   2657* 2659   2660   2660* 2661   2661*
KSOCAT  001  09FE 3359    2645
```

```
KSOCBA  001 119E 3380   2922
KSOCBB  001 119F 3381   2923
KSOCBC  001 11A0 3382   2924
KSOCBD  001 11A1 3383   2925
KSOCBE  001 11A2 3384   2926
KSOCBF  001 11A3 3385   2927
KSOCBG  001 11A4 3386   2928
KSOCBS  001 1193 3350   2841  2847*
KSOCB0  001 1194 3370   2791* 2802* 2899* 2906* 2945* 2963*
KSOCB1  001 1195 3371   2792* 2821* 2835* 2882* 2885* 2890  2892  2900* 2905  2910* 2911* 2949
KSOCB2  001 1196 3372   2812  2897  2906  2912* 2913*
KSOCB3  001 1197 3373   2910  2914* 2915*
KSOCB4  001 1198 3374   2911  2921* 2922* 2931*
KSOCB5  001 1199 3375   2912  2923* 2924*
KSOCB6  001 119A 3376   2913  2925* 2926*
KSOCB7  001 119B 3377   2914  2927* 2928*
KSOCB8  001 119C 3378   2915  2929*
KSOCB9  001 119D 3379   2921
KSOCDB  001 1194 3369   2998
KSOCHR  001 11A6 3387   2793* 2794  2794* 2946* 2947  2947*
KSOCNT  001 118F 3339   2710  2710* 2740* 2757* 2763
KSOCRS  001 117E 3340   2629  2642  2674  2716  2778
KSOCVT  001 098A 3358   2632
KSODGT  001 1189 3341   2655* 2656  2656* 2662
KSODPL  001 116C 3335   2572  3044* 3047
KSOELL  001 1190 3342   2741* 2752* 2758*
KSOEND  001 1191 3343   2845* 2948* 2967*
KSOEQU  001 1120 3317   2559  2560  2637  2650  2904  2988
KSOEXP  001 1183 3344   2905* 2929  2931
KSOE00  001 0000 3321   2899  3073
KSOICS  001 1180 3345   2780
KSOI00  001 1121 3331   2634  2647  2718  2720  2724  2877
KSOI01  001 1123 3332   2605  2612  2620  2682  2736  2741  2758  2768  2773  2806  2821  2835
                        2845  2963  2967  3024  3029
KSOI02  001 1125 3333   2606
KSOLAC  001 0002 3326   2657  2660  2661  2710
KSOLDT  001 0746 3357   2658
KSOLSA  001 1182 3346   2598* 2603
KSOLST  001 FFFF 3362   2581
KSOLVT  001 070C 3356   2630
KSOMFS  001 1192 3347   2827* 2833
KSONAT  001 09C4 3360   2643
KSOOFF  001 0000 3322   2676  3030
KSOONN  001 0001 3324   2629  2642  2674  2678  2701  2716  2778  2780  2827  2833  2841  2847
                        2868
KSOPD0  001 0000 3323   2598  2603  2613  2618  2624  2655  2683  2688  2699  2703  2798  2800
                        2807  2813  2819  2828  2843  2843* 2849  2851  2858  2864  2866  2952
                        2955  2960  2960* 3002  3004*
KSOPD1  001 0001 3325   2634  2636  2647  2649
KSOPGB  001 00FF 3327   3016
KSOSC1  001 118B 3348   2694* 2718  2729  2736*
KSOSC2  001 118D 3349   2694  2695* 2731  2738* 2740  2750  2757
KSOSTS  001 117F 3351   2676* 2678* 2701
KSOVAD  001 1185 3352   2597  2611  2636* 2649* 2711  2728* 2753* 2763* 2989
KSOVRL  001 0C27 2558
KSO1LN  001 0095 3365   2579
```

```
KSO1PG  001  0B07 3363    2579*
KSO190  004  0C2B 2571
KSO192  004  0C4D 2586
KSO194  004  0C54 2591
KSO195  004  0C62 2597    2587
KSO2LS  001  FEFF 3361    2578
KSO2PG  001  0C07 3364    2582*
KSO200  004  0C6E 2603    2607
KSO205  003  0C75 2605
KSO210  004  0C80 2611    2604
KSO215  003  0C87 2613
KSO220  003  0C8D 2618
KSO225  003  0C96 2624    2619
KSO230  003  0C9C 2629
KSO240  004  0CAA 2633    2631  2664
KSO250  003  0CC4 2642    2625
KSO260  004  0CD2 2646    2644
KSO270  004  0CEB 2655    2614
KSO280  003  0D16 2674    2651
KSO284  003  0D1C 2676
KSO286  003  0D22 2678    2675  2704
KSO290  003  0D25 2682    2677  2684
KSO295  003  0D2F 2688
KSO300  004  0D36 2693
KSO310  003  0D43 2699
KSO320  004  0D5B 2709    2700
KSO321  004  0D80 2724    2719
KSO322  004  0D87 2726    2722
KSO323  004  0D8F 2728    2721  2725
KSO324  004  0D93 2729
KSO326  004  0D9B 2731
KSO330  004  0DA3 2736    2739
KSO335  004  0DB2 2740    2737  2742
KSO337  003  0DBE 2746
KSO340  004  0DC1 2750    2717
KSO345  004  0DD0 2757    2759
KSO347  004  0DDC 2763    2746
KSO350  004  0DE0 2767    2709*
KSO360  004  0DE7 2772    2638
KSO370  003  0DF2 2778
KSO375  003  0DF9 2780
KSO400  003  0E00 2791
KSO410  003  0E0D 2798
KSO420  003  0E1C 2806    2799  2808
KSO430  003  0E1F 2807    2801
KSO440  003  0E26 2812
KSO450  003  0E2F 2818    2822
KSO460  004  0E46 2827    2820  2854
KSO470  003  0E4A 2828    2814
KSO480  004  0E50 2833    2850
KSO490  004  0E64 2841    2834  2836
KSO500  006  0E72 2845
KSO510  003  0E7F 2848    2842  2846
KSO520  003  0E89 2851
KSO530  003  0E96 2858    2829  2852
KSO540  003  0E9F 2864
```

```
KSO550  003  0EAF 2869   2865
KSO560  004  0EB2 2873   2867
KSO570  004  0EB6 2877   2868* 2878
KSO575  006  0ECA 2885   2881
KSO580  004  0ED4 2890   2884
KSO590  004  0EE4 2897   2859
KSO592  004  0EF3 2904   2898
KSO594  004  0EFF 2910
KSO595  004  0F17 2916
KSO596  004  0F1E 2921
KSO597  004  0F45 2931   2917
KSO598  003  0F49 2935
KSO600  003  0F4C 2945   2779
KSO610  003  0F4F 2946
KSO620  003  0F59 2949
KSO630  003  0F5F 2952   2968
KSO640  003  0F68 2955
KSO650  004  0F6E 2960   2953
KSO660  006  0F7E 2967
KSO670  004  0F88 2972   2956
KSO680  003  0F96 2978
KSO900  004  0F99 2988   2781  2930  2935  2978
KSO910  004  0FA2 2993
KSO930  004  0FAA 2998
KSO940  003  0FAE 3002
KSO945  004  0FB7 3008   2591* 2972* 3003  3009
KSO950  004  0FBE 3016
KSO955  004  0FCE 3019   3016* 3017* 3018  3018* 3023
KSO960  006  0FD2 3023
KSO965  006  0FE4 3029
KSO970  004  0FF6 3036   2592* 2593  2593* 2973* 2974  2974* 3012  3023* 3024* 3025  3025* 3037
KSO972  004  0FFA 3044
KSO976  004  100A 3058
KSO991  004  100E 3066   2635
KSO992  004  1015 3068   2823  2837  2883  2886  2891  2893
KSO993  004  101C 3070   2689  2702  2705  2730  2732  2751
KSO994  004  1023 3072   2648
KSO995  004  1027 3073   2727  3067  3069  3071
KSO996  004  102B 3074
SCACNT  002  10DF 3295   3285* 3286*
SCACOF  001  0087 3267
SCACOM  001  0001 3266
SCAINC  001  0001 3265   3274  3280
SCAMMA  003  10BC 3289
SCANIT  001  109F 3269   2772  2774
SCASVE  002  10DD 3294   3271* 3286
SCASV1  001  10DC 3293
SCA100  003  10AE 3274   3276
SCA200  003  10B1 3275   3273
SCA250  003  10BB 3278   3289
SCA300  003  10BE 3280   3282
SCA400  004  10CE 3285   3278
SCA500  004  10D8 3288   3270* 3284
```

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

OL105 I   THE CODE LENGTH OF #KSOVR IS  4352 DECIMAL.
OL103 I   TOTAL NUMBER OF LIBRARY SECTORS REQUIRED IS     8
          NAME-#KSOVR,PACK-R1R1R1,UNIT-R1,RETAIN-P,LIBRARY-R,CATEGORY-000
```

```
           OVERLAY LINKAGE EDITOR CORE USAGE MAP     10/02/22


    START      CATEGORY   NAME AND      CODE LENGTH
    ADDRESS               ENTRY      HEXADECIMAL DECIMAL

    0C20          0     #KSOVR           1100    4352


OL100 I   THE TOTAL CORE USED BY #KSOVR IS  4352 DECIMAL.
OL101 I   THE START CONTROL ADDRESS OF THIS MODULE IS 0C20.
OL104 I   TOTAL NUMBER OF LIBRARY SECTORS REQUIRED IS   18
          NAME-#KSOVR,PACK-R1R1R1,UNIT-R1,RETAIN-P,LIBRARY-O
```