```
 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  13/01/22  PAGE   1

#ZUTMO    MODULE
```

```
    0000                        1 #ZUTMO START 0
                                2        PRINT ON,NODATA
                                3 *      @SYS  EXP-N
                              214+       PRINT ON
                              215 *      @FXD  EXP-N
                              620+       PRINT ON
                              621 *      @SPF  EXP-N
                             1084+       PRINT ON
                             1085 *      @ERM  EXP-N
                             1707+       PRINT ON
                             1708 *      @CAN  EXP-N
                             1811+       PRINT ON
                             1812 *      @B@E  EXP-N
                             2712+       PRINT ON
                             2713 *      @WKA  EXP-N
                             2783+       PRINT ON
```

```
                             2785 ***********************************************************************
                             2786 *  5703-XM1 COPYRIGHT IBM CORP. 1970                                  *
                             2787 *          REFER TO INSTRUCTIONS ON COPY RIGHT NOTICE, 120-2083        *
                             2788 *                                                                     *
                             2789 ***********************************************************************
                             2790 *STATUS                                                               *
                             2791 *    VERSION 1 MODIFICATION 0                                          *
                             2792 *                                                                     *
                             2793 *FUNCTION                                                              *
                             2794 *    * ZUTMON PRINTS A LIST OF TEN OPTIONS WHICH MAY BE INVOKED:       *
                             2795 *            CD = CORE DUMP                                            *
                             2796 *            DD = DISK DUMP                                            *
                             2797 *            VM = VIRTUAL MEMORY DUMP                                   *
                             2798 *            CP = CORE PATCH                                           *
                             2799 *            DP = DISK PATCH                                           *
                             2800 *            DC = DISK COMPARE                                         *
                             2801 *            DW = DISK WRITE                                           *
                             2802 *            H  = HALT SYSTEM                                          *
                             2803 *            T  = TRACE CHANGE                                         *
                             2804 *            R  = RETURN TO OPERATING SYSTEM                           *
                             2805 *                                                                     *
                             2806 *    FOLLOWING IS A DESCRIPTION OF EACH OPTION:                        *
                             2807 *    * CD - CORE DUMP                                                  *
                             2808 *      ENTERING 'CD' IN REPLY TO THE OPTION LIST INVOKES THE CORE DUMP *
                             2809 *      OPTION.  THE START 4ND END ADDRESSES ARE REQUESTED, EACH OF     *
                             2810 *      WHICH SHOULD BE A FOUR-CHARACTER, HEXADECIMAL NUMBER ENTERED    *
                             2811 *      VIA THE KEYBOARD.  AFTER THE ADDRESSES HAVE BEEN DETERMINED TO  *
                             2812 *      BE IN REAL OR SAVED CORE, THE DUMP IS MADE, BEGINNING AT THE    *
                             2813 *      START ADDRESS AND TERMINATING AT THE END ADDRESS OR WHEN THE    *
                             2814 *      END OF CORE IS REACHED.                                         *
                             2815 *    * DD - DISK DUMP                                                  *
                             2816 *      ENTERING 'DD' IN REPLY TO THE OPTION LIST INVOKES THE DISK DUMP *
                             2817 *      OPTION.  THE BEGINNING ADDRESS AND THE SECTOR COUNT INDICATING  *
                             2818 *      THE LENGTH OF THE DUMP ARE REQUETED.  THE REPLIES SHOULD BE A   *
                             2819 *      FOUR-CHARACTER, HEXADECIMAL ADDRESS AND A DECIMAL SECTOR COUNT. *
                             2820 *      THE SECTOR COUNT IS CONVERTED TO BINARY, AND THE DUMP IS MADE   *
                             2821 *      BEGINNING AT THE SPECIFIED DISK ADDRESS AND TERMINATING WHEN    *
                             2822 *      THE NUMBER OF SECTORS HAVE BEEN DUMPED OR THE LAST SECTOR ON    *
                             2823 *      DISK HAS BEEN DUMPED.                                           *
                             2824 *    * VM - VIRTUAL MEMORY DUMP                                        *
                             2825 *      ENTERING 'VM' IN REPLY TO THE OPTION LIST INVOKES THE VIRTUAL   *
                             2826 *      MEMORY DUMP OPTION.  THE BEGINNING AND ENDING LINE NUMBERS ARE  *
                             2827 *      REQUESTED, EACH OF WHICH SHOULD BE A FOUR-CHARACTER DECIMAL     *
                             2828 *      NUMBER ENTERED VIA THE KEYBOARD.  ZDUMPV IS CALLED TO PERFORM   *
                             2829 *      THE VIRTUAL MEMORY DUMP.                                        *
                             2830 *    * CP - CORE PATCH OPTION                                          *
                             2831 *      ENTERING 'CP' IN REPLY TO THE OPTION LIST INVOKES THE CORE PATCH*
                             2832 *      OPTION.  THE START ADDRESS AND PATCH DATA ARE REQUESTED.  THE   *
                             2833 *      REPLY FOR THE START ADDRESS SHOULD BE A FOUR-CHARACTER,         *
                             2834 *      HEXADECIMAL NUMBER.  THE PATCH DATA SHOULD BE CONTIGUOUS        *
                             2835 *      HEXADECIMAL CHARACTERS.  IF NO CHANGE IS DESIRED, A SPACE SHOULD*
                             2836 *      BE ENTERED,  WHEN THE PATCH DATA IS TERMINATED BY A CARRIAGE    *
                             2837 *      RETURN AND THE ADDRESS IS DETERMINED TO BE IN SAVED OR REAL     *
                             2838 *      CORE, THE PATCH DATA IS PLACED AT THE SPECIFIED ADDRESS.        *
                             2839 *    * DP - DISK PATCH                                                 *
                             2840 *      ENTERING 'DP' IN REPLY TO THE OPTION LIST INVOKES THE DISK PATCH*
```

```
                          2841 *      OPTION.  THE START DISK ADDRESS, THE DISPLACEMENT FROM IT, AND   *
                          2842 *      THE PATCH DATA ARE REQUESTED.  THE REPLIES TO THESE REQUESTS      *
                          2843 *      SHOULD BE A SECTOR BOUNDARY, FOUR-CHARACTER, HEXADECIMAL DISK     *
                          2844 *      ADDRESS, A HEXADECIMAL DISPLACEMENT FROM THAT ADDRESS, AND        *
                          2845 *      CONTIGUOUS, HEXADECIMAL PATCH DATA.  IF NO CHANGE IS DESIRED, A   *
                          2846 *      SPACE SHOULD BE ENTERED.  WHEN THE PATCH DATA IS TERMINATED BY    *
                          2847 *      A CARRIAGE RETURN, THE PATCH DATA IS PLACED AT THE SPECIFIED      *
                          2848 *      DISPLACEMENT FROM THE ADDRESS INDICATED.                          *
                          2849 *    * DC - DISK COMPARE                                                 *
                          2850 *      ENTERING 'DC' IN REPLY TO THE OPTION LIST INVOKES THE DISK        *
                          2851 *      COMPARE OPTION.  THE TWO DISK ADDRESSES TO BE COMPARED AND THE    *
                          2852 *      NUMBER OF SECTORS TO BE COMPARED ARE REQUESTED.  THE REPLIES      *
                          2853 *      SHOULD BE TWO FOUR-CHARACTER, HEXADECIMAL ADDRESSES TO BE         *
                          2854 *      COMPARED, AND A DECIMAL COUNT OF SECTORS TO BE COMPARED.          *
                          2855 *      IF THE DATA AT THE TWO ADDRESSES IS NOT ENTIRELY EQUAL FOR THE    *
                          2856 *      WHOLE SECTOR, THE FIRST NON-EQUAL BYTES WILL BE DOCUMENTED FOR    *
                          2857 *      BOTH SECTORS WITH THE DISK ADDRESS OF EACH SECTOR, THE DISPLACE-  *
                          2858 *      MENT, AND THE DATA FOUND THERE.  THE COMPARISON IS CONTINUED FOR  *
                          2859 *      THE SPECIFIED NUMBER OF SECTORS OR UNTIL THE END OF THE PHYSICAL  *
                          2860 *      DISK IS REACHED.                                                  *
                          2861 *    * DW - DISK WRITE                                                   *
                          2862 *      ENTERING 'DW' IN REPLY TO THE OPTION LIST INVOKES THE DISK WRITE  *
                          2863 *      OPTION.  THE ADDRESS OF THE SECTOR TO BE WRITTEN AND THE ADDRESS   *
                          2864 *      WHERE IT IS TO BE WRITTEN ARE REQUESTED.  THE REPLIES SHOULD BE   *
                          2865 *      TWO FOUR-CHARACTER, HEXADECIMAL ADDRESSES.  THE SECTOR AT THE     *
                          2866 *      FIRST ADDRESS SPECIFIED WILL PE WRITTEN AT THE SECOND ADDRESS     *
                          2867 *      SPECIFIED.                                                        *
                          2868 *    * H - HALT SYSTEM                                                   *
                          2869 *      ENTERING 'H' IN REPLY TO THE OPTION LIST INVOKES THE HALT         *
                          2870 *      OPTION.  ALL OF CORE THAT WAS SAVED OTHER THAN THE SAVED NUCLEUS  *
                          2871 *      AREA, INCLUDING ANY PATCHES MADE TO SAVED CORE, IS RESTORE        *
                          2872 *      AFTER SAVED CORE IS RESTORED, A HARD SYSTEM HALT (HALT CODE =     *
                          2873 *      D5) RESULTS.                                                      *
                          2874 *    * R - RETURN TO OPERATING SYSTEM                                    *
                          2875 *      ENTERING 'R' IN REPLY TO THE OPTION LIST INVOKES THE RETURN TO    *
                          2876 *      THE OPERATING SYSTEM OPTION.  SAVED CORE, EXCLUDING THE NUCLEUS   *
                          2877 *      AND INCLUDING ANY CORE PATCHES MADE, IS RESTORED AND CONTROL IS   *
                          2878 *      RETURNED TO THE OPERATING SYSTEM.                                 *
                          2879 *    * T - TRACE CHANGE                                                  *
                          2880 *      ENTERING 'T' IN REPLY TO THE OPTION LIST INVOKES THE TRACE        *
                          2881 *      CHANGE OPTION.  THE CURRENT STATUS OF THE SYSTEM TRACE FEATURE    *
                          2882 *      IS REVERSED AND CONTROL IS PASSED TO THE OPERATING SYSTEM IN      *
                          2883 *      THE SAME MANNER AS IN THE RETURN OPTION.                          *
                          2884 *                                                                        *
                          2885 *ENTRY POINTS                                                            *
                          2886 *    * THE ENTRY IS THE FIRST EXECUTABLE INSTRUCTION, ZUT010.  ZUTMON    *
                          2887 *      IS ENTERED BY SWITCHING SYSTEM RESET/SYSTEM START IN SUCCESSION   *
                          2888 *      AND IS LOADED AND BRANCHED TO BY DEXMGS.  BOTH THE BASE REGISTER  *
                          2889 *      AND INDEX REGISTER ARE USED.                                      *
                          2890 *                                                                        *
                          2891 *INPUT                                                                   *
                          2892 *    * ALL INPUT IS BY WAY OF THE KEYBOARD AND VARIES ACCORDING TO THE   *
                          2893 *      OPTION SELECTED.  SEE THE OPTION FUNCTION FOR SPECIFIC INPUT.     *
                          2894 *                                                                        *
                          2895 *OUTPUT                                                                  *
                          2896 *    * ALL OUTPUT IS TO THE MATRIX PRINTER AND VARIES ACCORDING TO THE   *
```

```
                         2897 *      OPTION SELECTED.  SEE THE OPTION FUNCTION FOR SPECIFIC OUTPUT.  *
                         2898 *                                                                      *
                         2899 *EXTERNAL REFERENCES                                                   *
                         2900 *      $$PRES($KEYCD) - ENTRY TO ENABLE KEYBOARD.                       *
                         2901 *      $$INLN - INPUT LINE BUFFER.                                      *
                         2902 *      $$PRNT - ENTRY TO PRINT OUTPUT.                                  *
                         2903 *      DL2ICS - ENTRY TO DISK IOCR.                                     *
                         2904 *      $DISKN - ENTRY TO PHYSICAL DISK ROUTINE IN THE SYSTEM NUCLEUS.   *
                         2905 *      C4BIN2 - ENTRY TO CONVERT DECIMAL TO BINARY.                     *
                         2906 *      SCANIT - ENTRY TO VALIDATE INPUT DATA.                           *
                         2907 *      $CAIPL - ENTRY TO RETURN TO OPERATING SYSTEM.                    *
                         2908 *      $PAUSD($$RTRN) - ENTRY TO SWAP CORE ROUTINE.                     *
                         2909 *      $RSTR  - EXIT TO RESTORE CORE.                                   *
                         2910 *                                                                      *
                         2911 *EXITS, NORMAL                                                         *
                         2912 *   NORMAL EXIT IS TO $RSTR UPON THE 'H', 'R', OR 'T' OPTIONS.          *
                         2913 *                                                                      *
                         2914 *EXITS, ERROR                                                          *
                         2915 *   NONE.                                                              *
                         2916 *                                                                      *
                         2917 *TABLES/WORK AREAS                                                     *
                         2918 *   * A TWO-SECTOR BUFFER IS MAINTAINED IN CORE FOR DISK I/O.           *
                         2919 *   * A TABLE OF VALID PRINTABLE CHARACTERS IS USED IN THE              *
                         2920 *      INTERPRETATION OF CORE AND DISK DUMPS.                           *
                         2921 *                                                                      *
                         2922 *ATTRIBUTES                                                            *
                         2923 *   ZUTMON IS RELOCATABLE                                              *
                         2924 *                                                                      *
                         2925 *CHARACTER CODE DEPENDENCY                                             *
                         2926 *   THE OPERATION OF THIS MODULE DEPENDS UPON A CLASSIFICATION OF       *
                         2927 *   THE EXTERNAL CHARACTER SET BY MEANS OF A TABLE.  THE TABLE IS       *
                         2928 *   CONSTRUCTED FOR THE EBCDIC CHARACTER SET AND IS ARRANGED SO         *
                         2929 *   THAT REDEFINITION OF ALL CHARACTER CONSTANTS, BY REASSEMBLY,        *
                         2930 *   WILL RESULT IN A CORRECT TABLE FOR THE NEW DEFINITIONS IF THE       *
                         2931 *   EXTERNAL CHARACTER SET REMAINS UNCHANGED.                           *
                         2932 *   THE DECIMAL NUMBERS MUST BE CODED SO THAT THE LOW ORDER FOUR        *
                         2933 *   BITS WHEN CONSIDERED AS A BINARY INTEGER, IDENTIFY THE VALUE        *
                         2934 *   OF THE DIGIT.                                                       *
                         2935 *                                                                      *
                         2936 *NOTES                                                                 *
                         2937 *   ERROR PROCEDURES                                                   *
                         2938 *     * IF THE RESPONSE TO THE OPTION LIST IS NOT VALID, A QUESTION     *
                         2939 *       MARK WILL BE PRINTED, AND THE OPTION LIST WILL BE REPRINTED     *
                         2940 *       TO ALLOW A CORRECT ENTRY.                                       *
                         2941 *     * IF THE PATCH DATA ENTERED FOR THE CORE PATCH OR DISK PATCH      *
                         2942 *       OPTIONS IS INVALID, A QUESTION MARK WILL BE PRINTED AND ALL     *
                         2943 *       OF THE PATCH DATA MUST BE REENTERED.                            *
                         2944 *     * IF ANY RESPONSES TO ALL OTHER REQUESTS SHOULD EVER BE           *
                         2945 *       INVALID, THE REQUEST WILL BE REPRINTED TO ALLOW A CORRECT       *
                         2946 *       ENTRY.                                                          *
                         2947 *                                                                      *
                         2948 *   REGISTER USAGE                                                     *
                         2949 *     BOTH THE INDEX AND BASE REGISTERS ARE USED DURING EXECUTION.      *
                         2950 *                                                                      *
                         2951 *   SAVED/RESTORED AREAS                                               *
                         2952 *     NONE                                                             *
```

```
                          2953 *                                                            *
                          2954 *     MODIFICATION CONSIDERATIONS                            *
                          2955 *        WHEN BRANCH CONDITIONS ARE SET TO ZDUMPV TO PERFORM THE      *
                          2956 *        VIRTUAL MEMORY DUMP, THEY ARE SET SUCH THAT ZDUMPV OVERLAYS  *
                          2957 *        THE LAST PART OF ZUTMON.   ZDUMPV RETURNS TO ZUTMON AT A     *
                          2958 *        SPECIFIC ADDRESS WHICH PRECEDES THE OVERLAID SECTION. UPON   *
                          2959 *        RETURN TO ZUTMON, THE OVERLAID SECTION IS RESTORED.  IF THIS *
                          2960 *        RETURN ADDRESS.         , SHOULD BE CHANGED, THE BRANCH ADDRESS  *
                          2961 *        IN ZDUMPV SHOULD BE CHANGED ACCORDINGLY.             *
                          2962 *                                                            *
                          2963 *     REQUIRED MODULES                                       *
                          2964 *        @SYSEQ – COMMON SYSTEM EQUATES.                      *
                          2965 *        @FXDEQ – SYSTEM NUCLEUS ADDRESSES & INDICATOR VALUES EQUATES.  *
                          2966 *        @CANEQ – COMMON CORE LOCATIONS OUTSIDE NUCLEUS EQUATES.    *
                          2967 *        @SPFEQ – SYSTEM PROGRAM FILE EQUATES.                *
                          2968 *        DL2ICS – TWO TRACK LOGICAL IOCR.                     *
                          2969 *        SCANIT – VALIDITY CHECKING SUBROUTINE.              *
                          2970 *        C4BIN2 – CONVERT DECIMAL TO BINARY ROUTINE.         *
                          2971 *                                                            *
                          2972 *     OTHER                                                  *
                          2973 *        IF ZUTMON IS ENTERED (SYSTEM RESET/SYSTEM START) TWICE IN     *
                          2974 *        SUCCESSION, IT WILL BE PART OF SAVED CORE AFTER THE SECOND    *
                          2975 *        ENTRY.  THIS SHOULD ONLY BE DONE TO PATCH OR DUMP ZUTMON.     *
                          2976 ***********************************************************************
```

```
                                2978 *****************************************************************************
                                2979 *                                                                           *
                                2980 *   ZUTMON - THIS IS THE SELECTION PROGRAM TO CHOOSE A F.E. UTILITY         *
                                2981 *                                                                           *
                                2982 *****************************************************************************
                                2983 * A UTILITY WILL BE CHOSEN AFTER A MESSAGE PRINTS, THE MESSAGE IS:          *
                                2984 * '> CD,DD,VM,CP,DP,DC,DW, H,R,T,M...'                                      *
                                2985 * PROGRAM SELECTION WILL BE DETERMINED BY THE LETTER TYPED ON THE           *
                                2986 * KEYBOARD ACCORDING TO THE FOLLOWING TABLE --                              *
                                2987 *                                                                           *
                                2988 *                    CD = CORE DUMP OPTION                                  *
                                2989 *                    DD = DISK DUMP OPTION                                  *
                                2990 *                    VM = VIRTUAL MEMORY DUMP OPTION                        *
                                2991 *                    CP = CORE PATCH OPTION                                 *
                                2992 *                    DP = DISK PATCH OPTION                                 *
                                2993 *                    DC = DISK COMPARE OPTION                               *
                                2994 *                    DW = DISK WRITE OPTION                                 *
                                2995 *                    H  = HALT SYSTEM OPTION                                *
                                2996 *                    R  = RETURN TO SYSTEM OPTION                           *
                                2997 *                    T  = CHANGE TRACE OPTION                               *
                                2998 *                                                                           *
                                2999 *****************************************************************************
                                3000 *     HDR IZUTMO
                                3001 *****************************************************************************
                                3002 * PROGRAM HEADER FOR DISK LOAD                                              *
                                3003 *****************************************************************************
                                3004 *#$ZUTM EQU   X'1C14'                      DISK ADDR OF #ZUTMO
                                3005 *#$$ZUT EQU   X'0C00'                      CORE LOAD ADDRESS OF #ZUTMO
                                3006 *#$@ZUT EQO   020                          SECTOR COUNT OF #ZUTMO
   0C00                         3007        ORG   #$$ZUT                        CORE LOAD ADDRESS
             0C00 3008 $$$$$$ EQU   *                            FIRST LOCATION IN PROGRAM
   0C00 7BE9E4E3D4D6            0C05 3009        DC    CL6'#ZUTMO'               PROGRAM NAME
   0C06 5C                      0C06 3010        DC    IL1'092'                  PROGRAM NUMBER OF OZUTMO
                                0C07 3011 #ZUTM EQU   *                         ENTRY POINT TO PROGRAM
                                3012 *** END OF EXPANSION ***

                                3014 *****************************************************************************
                                3015 * SELECTION ROUTINE - ZUTMON ENTRY POINT                                   *
                                3016 *****************************************************************************
                                3017 * THIS ROUTINE SAVES AND CHANGES PRINTER STATUS, PROVIDES CHOICE           *
                                3018 * OF OPTION, AND SELECTS THE ROUTINE TO PERFORM THE OPTION.                *
                                3019 *                                                                           *
                                3020 * THE TRACE, HALT, AND RETURN OPTIONS ARE IN THIS SECTION. CONTROL         *
                                3021 * WILL ALWAYS RETURN TO THIS ROUTINE UPON COMPLETION OF THE SPECIFIED      *
                                3022 * FUNCTION                                                                  *
                                3023 *****************************************************************************
                                3024 *
                                3025 * ENTER ZUTMON F.E. UTILITY AID
                                3026 *
                                3027 *ZUT010 ENTER ENTRY                        NAME
                                0C07 3028 ZUT010 EQU   *                         MODULE ENTRY POINT
                                3029 *** END OF EXPANSION ***

   0C07 3C 80 0476              3031        MVI   $CIMSK,@NOP               MASK INTERRUPTS
   0C0B 0C 00 0DA1 03C1         3032        MVC   ZUTLMA,$LMRGN(1)          SAVE LEFT MARGIN
   0C11 C0 87 1D52              3033        B     ZUT900                    FIND PRINT POSITION
```

```
                              3034 *ZUT012 PRNT  ZUT540                       EXECUTE CARRAGE RETURN
     0C15 C0 87 0707         3035 ZUT012 B     $$PRNT                         PRINT ON MATRIX PRINTER
     0C19 0D6D          0C1A 3036        DC    AL2(ZUT540)                    PPL ADDRESS
                              3037 *** END OF EXPANSION ***

     0C1B 3C 00 03C2         3039 ZUT015 MVI   $PRPOS,0                       SET POSITION TO ZERO
     0C1F 0C 00 0DA0 03C0    3040        MVC   ZUTMAR,$RMRGN(1)               SAVE RT MARGIN
     0C25 3C 82 03C0         3041        MVI   $RMRGN,ZUTMRR                  SET RIGHT MARGIN
     0C29 0C 01 0D9D 044B    3042        MVC   ZUTPRR,$PRDEV(@CADDR)          SAVE SYSTEM PRINTER
     0C2F 0C 01 044B 0D9B    3043        MVC   $PRDEV(@CADDR),ZUTPRT          CHANGE TO CONSOLE PRINTER
     0C35 3B 08 03D2         3044        SBF   $IOIND,$CMDKY                  TURN OFF COMMAND KEYS
                              3045 *
                              3046 * PRINT OPTION LIST AND WAIT FOR REPLY
                              3047 *
                              3048 *ZUT020 PRNT  ZUT540                       SPACE CARRAIGE
     0C39 C0 87 0707         3049 ZUT020 B     $$PRNT                         PRINT ON MATRIX PRINTER
     0C3D 0D6D          0C3E 3050        DC    AL2(ZUT540)                    PPL ADDRESS
                              3051 *** END OF EXPANSION ***

                              3053 *      PRNT  ZUT510                        PRINT REQUEST FOR CHOICE
     0C3F C0 87 0707         3054        B     $$PRNT                         PRINT ON MATRIX PRINTER
     0C43 0D65          0C44 3055        DC    AL2(ZUT510)                    PPL ADDRESS
                              3056 *** END OF EXPANSION ***

                              3058 *      PRNT  $WAITF                        WAIT FOR END OF PRINT
     0C45 C0 87 0707         3059        B     $$PRNT                         PRINT ON MATRIX PRINTER
     0C49 057F          0C4A 3060        DC    AL2($WAITF)                    PPL ADDRESS
                              3061 *** END OF EXPANSION ***

     0C4B C0 87 0DA5         3063        B     ZUTIRI                         * GO TEST FOR INTERRUPTS

                              3065 *
                              3066 * GET KEY DATA, TEST FOR ERROR, REPRINT MESSAGE FOLLOWING '?' IF ERROR
                              3067 * PRESENT
                              3068 *
     0C4F C0 87 0D31         3069        B     ZUTKEY                         GET KEYBOARD DATA
     0C53 38 FF 0D99         3070        TBN   ZUTKER,ZUT820                  TEST FOR ERROR
     0C57 C0 10 0CB4         3071        BT    ZUT030                         GO PRINT ERROR MESSAGE
                              3072 *
                              3073 * TEST FOR EACH OPTION AND BRANCH TO THE ROUTINE THAT PERFORMS
                              3074 * THAT FUNCTION
                              3075 *
     0C5B BD C3 00           3076        CLI   0(,@XR),C'C'                   IS THIS A CORE FUNCTION
     0C5E F2 01 0E           3077        JNE   ZUT021                         IF NOT GO TO DISK TESTS
     0C61 BD C4 01           3078        CLI   1(,@XR),C'D'                   IS THIS A CORE DUMP
     0C64 C0 81 10E3         3079        BE    ZCORED                         GO TO CORE DUMP
     0C68 BD D7 01           3080        CLI   1(,@XR),C'P'                   IS THIS A PATCH
     0C6B C0 81 10A9         3081        BE    ZCD006                         GO TO CORE PATCH
     0C6F BD C4 00           3082 ZUT021 CLI   0(,@XR),C'D'                   IS THIS A DISK OPERATION
     0C72 F2 01 1C           3083        JNE   ZUT022                         SKIP DISK OPS
     0C75 BD C4 01           3084        CLI   1(,@XR),C'D'                   IS THIS A DUMP
     0C78 C0 81 10FB         3085        BE    ZDUMDK                         GO DUMP DISK
     0C7C BD D7 01           3086        CLI   1(,@XR),C'P'                   IS THIS A PATCH
     0C7F C0 81 1092         3087        BE    ZCD005                         GO TO DISK PATCH
     0C83 BD C3 01           3088        CLI   1(,@XR),C'C'                   IS THIS A DISK COMPARE
     0C86 C0 81 10CF         3089        BE    ZDCOMP                         GO TO DISK COMPARE
```

```
      #ZUTMO - F.E. UTILITY AID PROGRAM


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT         VER 15, MOD 00  13/01/22  PAGE   9

     0C8A BD E6 01            3090         CLI   1(,@XR),C'W'            IS THIS A SECTOR WRITE
     0C8D C0 81 10BA          3091         BE    ZUTCOP                 GO WRITE DISK
     0C91 BD C8 00            3092 ZUT022 CLI    0(,@XR),C'H'           IS THIS A HALT
     0C94 C0 81 0CE1          3093         BE    ZUT033                 RETURN AND HALT SYSTEM
     0C98 BD D9 00            3094         CLI   0(,@XR),C'R'           IS THIS A RETURN TO SYSTEM
     0C9B C0 81 0CDB          3095         BE    ZUT031                 RETURN AND RUN
     0C9F BD E5 00            3096         CLI   0(,@XR),C'V'           IS THIS A VM DUMP
     0CA2 C0 81 0F28          3097         BE    ZDMVM                  GO TO VM
     0CA6 BD E3 00            3098         CLI   0(,@XR),C'T'           IS THIS A TRACE CHANGE
     0CA9 C0 81 0CC8          3099         BE    ZUTTFL                 GO TO TRACE FLIP
     0CAD BD D4 00            3100         CLI   0(,@XR),C'M'           IS THIS A LIBRARY MAP      1-3
     0CB0 C0 81 0E22          3101         BE    ZUTLIB                 GO TO LIBRARY MAPPING     1-3
                              3102 *
                              3103 * IF OPTION ENTERED IS NOT VALID GO REPRINT OPTION LIST.
                              3104 *
                              3105 *ZUT030 PRNT  ZUT530                 ELSE ASK FOR ENTRY AGAIN
     0CB4 C0 87 0707          3106 ZUT030 B      $$PRNT                 PRINT ON MATRIX PRINTER
     0CB8 0D69           0CB9 3107         DC    AL2(ZUT530)            PPL ADDRESS
                              3108 *** END OF EXPANSION ***
                              3109 *      PRNT  $WAITF                  WAIT FOR END
     0CBA C0 87 0707          3110         B     $$PRNT                 PRINT ON MATRIX PRINTER
     0CBE 057F           0CBF 3111         DC    AL2($WAITF)            PPL ADDRESS
                              3112 *** END OF EXPANSION ***

     0CC0 C0 87 0DA5          3114         B     ZUTIRI                 * GO TEST FOR INTERRUPTS
     0CC4 C0 87 0C39          3115         B     ZUT020                 REPEAT REQUEST MESSAGE
                         0CC8 3116 ZUTTFL EQU   *                       START OF TRACE FLIP
     0CC8 3D 80 054E          3117         CLI   $TROVR,@NOP            IS TRACE OFF ?
     0CCC F2 81 08            3118         JE    ZUTTF1                 IF OFF GO TURN TRACE ON
     0CCF 3C 80 054E          3119         MVI   $TROVR,@NOP            ELSE TURN OFF TRACE
     0CD3 C0 87 0CDB          3120         B     ZUT031                 GO RETURN TO BIS

     0CD7 3C 87 054E          3122 ZUTTF1 MVI    $TROVR,@UCB            TURN ON TRACE
     0CDB 0C 01 04FE 0DA4     3123 ZUT031 MVC    $SRTRN(@CADDR),ZUT750  MOVE CAIPL ADDR INTO PAUSD
                              3124 *ZUT033 PRNT  ZUT540
     0CE1 C0 87 0707          3125 ZUT033 B      $$PRNT                 PRINT ON MATRIX PRINTER
     0CE5 0D6D           0CE6 3126         DC    AL2(ZUT540)            PPL ADDRESS
                              3127 *** END OF EXPANSION ***
                              3128 *
                              3129 * RESTORE THE STATUS OF THE SYSTEM PRINTER
                              3130 *
     0CE7 0C 00 0D9F 0DA1     3131         MVC   ZUTTIP(1),ZUTLMA       PUT MARGIN COUNT IN WORK AREA
     0CED 0F 00 0D9F 0E21     3132 ZUT034 SLC    ZUTTIP(1),CVBH94       REDUCE MARGIN COUNT
     0CF3 F2 82 0A            3133         JM    ZUT035                 EXIT IF COUNT GONE
                              3134 *      PRNT  ZUT541                   SPACE OVER BY PRINTING BLANK
     0CF6 C0 87 0707          3135         B     $$PRNT                 PRINT ON MATRIX PRINTER
     0CFA 0D71           0CFB 3136         DC    AL2(ZUT541)            PPL ADDRESS
                              3137 *** END OF EXPANSION ***

     0CFC C0 87 0CED          3139         B     ZUT034                 GO REDUCE COUNTER BY ONE
     0D00 0C 00 03C1 0DA1     3140 ZUT035 MVC    $LMRGN,ZUTLMA(1)       RESTORE LEFT MARGIN
     0D06 0C 00 03C0 0DA0     3141         MVC   $RMRGN,ZUTMAR(1)       RESTORE RIGHT MARGIN
     0D0C 0C 01 044B 0D9D     3142         MVC   $PRDEV(@CADDR),ZUTPRR  RESTORE PRINT DEVICE
     0D12 3B 03 03D1          3143         SBF   $XIND2,$EXCMD+$PAUSE   TURN OFF EXECUTE AND PAUSE
                              3144 *
                              3145 * RETURN TO RESTORE CORE SAVED WHEN ZUTMON WAS BROUGHT IN
```

#ZUTMO - F.E. UTILITY AID PROGRAM

                           3146 *
     0D16 C0 87 04D6   3147 ZUT036 B $RSTR   GO TO RESTORE
                           3148 * END OF SELECTION ROUTINE

```
      #ZUTMO - F.E. UTILITY AID PROGRAM

ERR LOC   OBJECT CODE     ADDR STMT SOURCE STATEMENT           VER 15, MOD 00  13/01/22  PAGE  11
                               3150 *        PATCH 23,1                    PATCH AREA
                               3151 ***********************************************************************
                               3152 *   PATCH AREA 1                                                     *
                               3153 ***********************************************************************
     0D1A                 0D30 3154 $$$$$L DS CL23                     PATCH AREA FOR PROGRAM
                               3155 ***********************************************************************
```

```
                          3157 ************************************************************************
                          3158 * ENTER ZUTKEY - SUBROUTINE TO GET AND VALIDATE KEYBOARD ENTRIES      *
                          3159 ************************************************************************
                          3160 *
                     0D31 3161 ZUTKEY EQU    *                                *
                          3162 *ZUT200 ENTER EXIT=ZUTKR,@BR,,@ARR
                     0D31 3163 ZUT200 EQU    *                       MODULE ENTRY POINT
    0D31 34 01 0D60       3164        ST     ZUTKR0+@OP1,@BR         SAVE @BR
    0D35 34 08 0D64       3165        ST     ZUTKR2+@OP1,@ARR        SAVE RETURN ADDRESS
                          3166 *** END OF EXPANSION ***

    0D39 C2 02 0607       3168        LA     $$INLN,@XR                       *
    0D3D C0 87 0890       3169        B      $$PRES                  BRANCH TO KEYBOARD ROUTIPE
    0D41 38 10 03C3       3170 ZUT210 TBN    $KEYCD,$KYBSY           TEST FOR KEY BUSY
    0D45 C0 10 0D41       3171        BT     ZUT210                  RETURN TO TEST IF BUSY
    0D49 C0 87 0DA5       3172        B      ZUTIRI                  * GO TEST FOR INTERRUPTS
    0D4D C0 87 1051       3173        B      SCANIT                  GO TO SCAN INPUT DATA
    0D51 3B FF 0D99       3174        SBF    ZUTKER,ZUT820           SET ERPOR SWITCH OFF
    0D55 C0 02 0D5D       3175        BNL    ZUT220                  TEST FOR ERROR
    0D59 3A FF 0D99       3176        SBN    ZUTKER,ZUT820           SET ERROR SWITCH ON
                     0D5D 3177 ZUT220 EQU    *
                          3178 *ZUTKR  EXIT  @BR,,RETURN
    0D5D C2 01 0000       3179 ZUTKR0 LA     *-*,@BR                 RESTORE @BR
    0D61 C0 87 0000       3180 ZUTKR2 B      *-*                     RETURN TO CALLING PROGRAM
                          3181 *** END OF EXPANSION ***
                          3182 *
                          3183 *    END OF KEYBOARD PACKAGE - ERROR WILL BE ON IF COMMA-EOS OCCURS
                          3184 *
                          3185 ************************************************************************
                          3186 *
                          3187 ************************************************************************
                          3188 *      SECTION DC'S                                                    *
                          3189 ************************************************************************
                          3190 *
                          3191 *ZUT510 PPL   FUNC=@PRINT,CNT=ZUT800,CADDR=ZUT700
                     0D65 3192 ZUT510 EQU    *                       PPL ADDRESS
    0D65 40         0D65 3193        DC     AL1(@PRINT)             FUNCTION REQUESTED
    0D66 23         0D66 3194        DC     AL1(ZUT800)             PRINT COUNT
    0D67 0D75       0D68 3195        DC     AL2(ZUT700)             DATA ADDRESS
                          3196 *** END OF EXPANSION ***

                          3198 *ZUT530 PPL   FUNC=@PRZETR,CNT=ZUT810,CADDR=ZUT710
                     0D69 3199 ZUT530 EQU    *                       PPL ADDRESS
    0D69 C0         0D69 3200        DC     AL1(@PRETR)             FUNCTION REQUESTED
    0D6A 01         0D6A 3201        DC     AL1(ZUT810)             PRINT COUNT
    0D6B 0D98       0D6C 3202        DC     AL2(ZUT710)             DATA ADDRESS
                          3203 *** END OF EXPANSION ***

                          3205 *ZUT540 PPL   FUNC=@RETRN,CNT=@RTRNC
                     0D6D 3206 ZUT540 EQU    *                       PPL ADDRESS
    0D6D 80         0D6D 3207        DC     AL1(@RETRN)             FUNCTION REQUESTED
    0D6E 80         0D6E 3208        DC     AL1(@RTRNC)             PRINT COUNT
    0D6F 0000       0D70 3209        DC     AL2(*-*)                DATA ADDRESS
                          3210 *** END OF EXPANSION ***

                          3212 *ZUT541 PPL   FUNC=@PRINT,CNT=ZUT810,CADDR=ZUTBLK
```

```
      #ZUTMO - F.E. UTILITY AID PROGRAM
```

```
                           0D71 3213 ZUT541 EQU    *                           PPL ADDRESS
      0D71 40              0D71 3214        DC     AL1(@PRINT)                  FUNCTION REQUESTED
      0D72 01              0D72 3215        DC     AL1(ZUT810)                  PRINT COUNT
      0D73 0DA2            0D74 3216        DC     AL2(ZUTBLK)                  DATA ADDRESS
                           3217 *** END OF EXPANSION ***
                           3218 *      END OF CONSTANTS
                           3219 *********************************************************************
                           3220 *      MESSAGE CONSTANTS                                            *
                           3221 *********************************************************************
                           0023 3222 ZUT800 EQU    35                          LENGTH OF OPTION LIST
                           0D75 3223 ZUT700 EQU    *
      0D75 6E40C3C46BC4C46B 0D97 3224       DC     CL(ZUT800)'> CD,DD,VM,CP,DP,DC,DW, H,R,T,M... '
                           0D98 3225 ZUT710 EQU    *
      0D98 6F              0D98 3226        DC     CL1'?'                       QUESTION ENTRY
                           3227 *
                           3228 *********************************************************************
                           3229 *      SELECTION ROUTINE CONSTANTS AND EQUATES                      *
                           3230 *********************************************************************
                           3231 *
      0D99                 0D99 3232 ZUTKER DS    CL1                          ERROR INDICATOR FOR KEY ROUTINE
      0D9A 0707            0D9B 3233 ZUTPRT DC    AL2($$PRNT)                  CONSTANT FOR CONSOLE PRINTER
      0D9C                 0D9D 3234 ZUTPRR DS    CL2                          PRINT DEVICE BUCKET
      0D9E                 0D9F 3235 ZUTTIP DS    CL2
      0DA0                 0DA0 3236 ZUTMAR DS    CL1
      0DA1                 0DA1 3237 ZUTLMA DS    CL1
      0DA2 40              0DA2 3238 ZUTBLK DC    CL1' '                       CONSTANT FOR RESTORING PR POS
      0DA3 049D            0DA4 3239 ZUT750 DC    AL2($CAIPL)                  POINTER TO $CAIPL
                           3240 *
                           3241 * EQUATES
                           3242 *
                           0001 3243 ZUT810 EQU    1                           LENGTH OF REQUEST MSG
                           00FF 3244 ZUT820 EQU    X'FF'                       EQUATE FOR SWITCH TO FF
                           0000 3245 ZUT830 EQU    X'00'                       EQUATE FOR SWITCH TO 00
                           0082 3246 ZUTMRR EQU    130                         RIGHT MARGIN SETTING IN ZUTMON

                           3248 *********************************************************************
                           3249 *      ROUTINE TO CHECK FOR THE PRESENCE OF INTERRUPT REQUEST       *
                           3250 *********************************************************************
                           3251 *
                           0DA5 3252 ZUTIRI EQU    *                           ENTRY TO CHECK FOR INTERRUPTS
      0DA5 34 08 0DB8      3253        ST     ZUTRET+@OP1,@ARR             SAVE RETURN ADDRESS
      0DA9 3D 80 0496      3254        CLI    $CISUS,@NOP                  IF INTERRUPT IS PRESENT
      0DAD 3C 87 0496      3255        MVI    $CISUS,@UCB                  * RESTORE MASK AND
      0DB1 C0 81 0C39      3256        BE     ZUT020                       * REPRINT OPTION LIST
      0DB5 C0 87 0000      3257 ZUTRET B      *-*                          RETURN TO CALLING PROGRAM
                           3258 *
                           3259 *********************************************************************
```

```
                             3261 ********************************************************************
                             3262 *       ROUTINE TO CONVERT BINARY TO HEXADECIMAL CHARACTERS            *
                             3263 ********************************************************************
                             3264 *
                        00F0 3265 CVBH20 EQU    X'F0'                        MASK FOR ZONE BITS
                        00FA 3266 CVBH25 EQU    X'FA'                        UPPER BOUND
                        0001 3267 CVBH35 EQU    X'01'                        DISP FORM 1 BYTE
                        0DB9 3268 CVBHEX EQU    *                            ENTRY
    0DB9 34 08 0DDC          3269        ST     CVBH50+@OP2,@ARR             SET LENGTH BYTE ADDR IN INST
    0DBD 36 08 0E1F          3270        A      CVBH90,@ARR                  CALC RETURN ADDR
    0DC1 34 08 0E1C          3271        ST     CVBH80+@OP1,@ARR             SAVE RETURN ADDR
    0DC5 34 01 0E14          3272        ST     CVBH76+@OP1,@BR              SAVE BASE REGISTER
    0DC9 34 02 0E18          3273        ST     CVBH78+@OP1,@XR              SAVE INDEX REGISTER
    0DCD 35 01 0DDC          3274        L      CVBH50+@OP2,@BR              XR POINTS TO PARAMETERS
    0DD1 75 02 02            3275        L      @CADDR(,@BR),@XR             GET SOURCE ADDR INTO XR
    0DD4 75 01 04            3276        L      @CADDR+@CADDR(,@BR),@BR      OUTPUT ADDR IN BASE REG
    0DD7 0C 00 0E1D 0000     3277 CVBH50 MVC    CVBHCT(1),*-*                PICKUP LENGTH COUNT
    0DDD 68 02 00 00         3278 CVBH52 MNZ    0(,@BR),0(,@XR)              MOVE 1ST HEX CHAR TO OUTPUT
    0DE1 68 03 01 00         3279        MNN    1(,@BR),0(,@XR)              MOVE 2ND HEX CHAR TO OUTPUT
    0DE5 7A F0 00            3280        SBN    0(,@BR),CVBH20               MAKE ZONE BITS HEX 'F' FOR BOT
    0DE8 7A F0 01            3281        SBN    1(,@BR),CVBH20               * OUTPUT CHARACTERS
    0DEB 7D FA 00            3282        CLI    0(,@BR),CVBH25               IS 1ST CHAR DIGIT 0-9 ?
    0DEE F2 82 05            3283        JL     CVBH55                       BRANCH IF YES
    0DF1 4E 00 00 0E20       3284        ALC    0(1,@BR),CVBH92              NO, MAKE IT ALPHA A-F
    0DF6 7D FA 01            3285 CVBH55 CLI    1(,@BR),CVBH25               IS 2ND CHAR DIGIT 0-9 ?
    0DF9 F2 82 05            3286        JL     CVBH60                       BRANCH IF YES
    0DFC 4E 00 01 0E20       3287        ALC    1(1,@BR),CVBH92              NO, MAKE IT ALPHA A-F
    0E01 D2 01 02            3288 CVBH60 LA     @CADDR(,@BR),@BR             MOVE OUTPUT PT
    0E04 E2 02 01            3289        LA     CVBH35(,@XR),@XR             MOVE INPUT CHAR PT
    0E07 0F 00 0E1D 0E21     3290        SLC    CVBHCT(1),CVBH94             ARE ALL CHARS CONVERTED ?
    0E0D C0 01 0DDD          3291        BNZ    CVBH52                       NO, GO CONVERT NEXT BYTE
                             3292 *
                             3293 *       RETURN FROM ROUTINE
                             3294 *
    0E11 C2 01 0000          3295 CVBH76 LA     *-*,@BR                      RESTORE BASE REG
    0E15 C2 02 0000          3296 CVBH78 LA     *-*,@XR                      RESTORE XR
    0E19 C0 87 0000          3297 CVBH80 B      *-*                          RETURN
                             3298 *
                             3299 *       CVBHEX STORAGE AREAS
                             3300 *
    0E1D                     0E1D 3301 CVBHCT DS     CL1                     BYTE COUNT FOR INPUT DATA
                             3302 *
                             3303 *       CVBHEX CONSTANTS
                             3304 *
    0E1E 0005                0E1F 3305 CVBH90 DC     XL2'0005'               LENGTH OF PARAMETER LIST
    0E20 C7                  0E20 3306 CVBH92 DC     XL1'C7'                 MAGIC CONSTANT
    0E21 01                  0E21 3307 CVBH94 DC     XL1'01'                 CONSTANT 1
                             3308 ********************************************************************
```

```
                          3310 *********************************************************************
                          3311 *
                          3312 * LOAD IN THE LIBRARY MAPPING OVERLAY (OPTION 'M')
                          3313 *
   0E22 C0 87 051E        3314 ZUTLIB B     $RLOAD                    LOAD AND BRANCH          1-3
   0E26 0E28         0E27 3315        DC    AL(@DADDR)(ZUTLBM)        * TO ZLBMAP              1-3
                          3316 *
                     2008 3317 ZUTLBD EQU   X'2008'                   ZLBMAP DISK ADDRESS      1-3
                     0002 3318 ZUTLBL EQU   2                         ZLBMAP LENGTH            1-3
                     1100 3319 ZUTLBC EQU   X'1100'                   ZLBMAP CORE ADDRESS      1-3
                          3320 *
                          3321 *ZUTLBM DPL   FUNC=@DGET,DADDR=WLBM,CNT=MULB,CADDR=IPSSZLB       1-3
                     0E28 3322 ZUTLBM EQU   *                         DISK PARAMETER LIST
   0E28 01           0E28 3323        DC    AL1(@DGET)                REQUESTED FUNCTION
   0E29 2008         0E2A 3324        DC    AL2(#$ZLBM)               DISK ADDRESS
   0E2B 02           0E2B 3325        DC    AL1(#$@ZLB)               SECTOR COUNT
   0E2C 1100         0E2D 3326        DC    AL2(#$$ZLB)               BUFFER ADDRESS
                          3327 *** END OF EXPANSION ***
```

```
      CVBHEX - CONVERT HEX TO EBCDIC


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT            VER 15, MOD 00  13/01/22  PAGE  16

                                3329 ********************************************************************************
                                3330 * VIRTUAL MEMORY DUMP ROUTINE - PASSES BEGINNING AND ENDING DUMP       *
                                3331 * ADDRESSES TO ZDUMPV AND RESTORES OVERLAID ZUTMON TO CORE UPON RTN     *
                                3332 ********************************************************************************
                                3333 *
     0F00                       3334         ORG    *,256,0                    SET UP IC TO A SECTOR BOUNDARY
     0F00 35 10 0FAB            3335         L      UVMCVB,@IAR               POINT TO CVBHEX WITHOUT ARR CHGE
                                3336 *                                        VM START AND END ADDR
     0F04              0F05 3337 UVM100 DS    CL2                             VM START ADDRESS
     0F06              0F07 3338 UVM101 DS    CL2                             VM END ADDRESS
                                3339 *
                                3340 * RESTORE OVERLAID SECTION OF ZUTMON TO CORE
                                3341 *
                       0F08 3342 UVMEND EQU   *                               POINTER TO SHUT DOWN VIM DUMP
                                3343 *CS020 EQU   X'122E'                     T E M P  HJS TEMP
     0F08 3C 80 122F            3344         MVI    ZCS020+@Q,@NOP            RESET FIRST TIME IN ZCSAVE
     0F0C 0C 01 1044 0F8D       3345         MVC    DL2RAD(@CADDR),UVMZUD     PUT ZUTMON ADDRESS IN DL2RAD
     0F12 0E 01 1044 0587       3346         ALC    DL2RAD(@CADDR),$BSADR     ADD DISPLACEMENT FACTOR
                                3347 *         DSKL2 UVMDP2,WAIT               CALL DISK
     0F18 C0 87 0FAC            3348         B      DL2ICS                    PERFORM RELATIVE DISK OP
     0F1C 0F86         0F1D 3349         DC     AL2(UVMDP2)               DPL ADDRESS
     0F1E C0 87 0025            3350         B      $DISKN                    WAIT AND CHECK DISK ERRORS
     0F22 057F         0F23 3351         DC     AL2($WAITF)               WAIT DPL ADDRESS
                                3352 *** END OF EXPANSION ***
     0F24 C0 87 0C39            3353         B      ZUT020                    CHANGE TO ZUT020 IF NO PRINT P
                                3354 *
                                3355 * SET UP REQUEST MESSAGES FOR LINE NUMBERS AND PRINT THEM
                                3356 *
                       0F28 3357 ZDMVM  EQU   *                               VM DUMP START
     0F28 0C 0C 118B 0F9C       3358         MVC    ZCDMV1,UVMLN1(UVMLNL)     MOVE IN LINE NUMBER MSG
     0F2E 0C 0C 11A8 0FA9       3359         MVC    ZCDMV2,UVMLN2(UVMLNL)     MOVE IN LINE NUMBER MSG
                                3360 *ZDV001 PRNT  ZCDM01                      PRINT REQUEST
     0F34 C0 87 0707            3361 ZDV001 B     $$PRNT                    PRINT ON MATRIX PRINTER
     0F38 121B         0F39 3362         DC     AL2(ZCDM01)               PPL ADDRESS
                                3363 *** END OF EXPANSION ***

     0F3A C0 87 0D31            3365         B      ZUTKEY                    GET KEY DATA
     0F3E C0 87 1CE2            3366         B      C4BIN2                    CONVERT TO HEX
     0F42 C0 04 0F34            3367         BNH    ZDV001                    ASK AGAIN IF IN ERROR
     0F46 0C 01 0F05 1D4C       3368         MVC    UVM100,C4BVAL(@CADDR)     MOVE DATA TO PARAMETER BUCKET
                                3369 *ZDV002 PRNT  ZCDM02                      PRINT REQUEST
     0F4C C0 87 0707            3370 ZDV002 B     $$PRNT                    PRINT ON MATRIX PRINTER
     0F50 121F         0F51 3371         DC     AL2(ZCDM02)               PPL ADDRESS
                                3372 *** END OF EXPANSION ***

     0F52 C0 87 0D31            3374         B      ZUTKEY                    GET KEY DATA
     0F56 C0 87 1CE2            3375         B      C4BIN2                    CONVERT TO HEX
     0F5A C0 04 0F4C            3376         BNH    ZDV002                    ASK AGAIN IF IN ERROR
     0F5E 0C 01 0F07 1D4C       3377         MVC    UVM101,C4BVAL(@CADDR)     MOVE IN PARAMETER
                                3378 *
                                3379 * GET ZDUMPV FROM DISK AND OVERLAY ZUTMON IN CORE
                                3380 *
     0F64 0C 01 1044 0F8F       3381         MVC    DL2RAD(@CADDR),UVMZDU     PUT ZDUMPV ADDR IN DL2RAD
                       0F6A 3382 ZUTLGO EQU   *
     0F6A 0E 01 1044 0587       3383         ALC    DL2RAD(@CADDR),$BSADR     ADD ON DISPLACEMENT FACTOR
                                3384 *         DSKL2 UVMDP1,WAIT               CALL DISK
```

CVBHEX - CONVERT HEX TO EBCDIC

```
     0F70 C0 87 0FAC           3385          B     DL2ICS                 PERFORM RELATIVE DISK OP
     0F74 0F80           0F75 3386          DC    AL2(UVMDP1)            DPL ADDRESS
     0F76 C0 87 0025          3387          B     $DISKN                 WAIT AND CHECK DISK ERRORS
     0F7A 057F           0F7B 3388          DC    AL2($WAITF)            WAIT DPL ADDRESS
                              3389 *** END OF EXPANSION ***

     0F7C C0 87 1107          3391          B     UVMG00                 BRANCH TO VM DUMP


                              3393 ********************************************************************
                              3394 * DPL'S FOR VIRTUAL MEMORY DUMP ROUTINE                           *
                              3395 ********************************************************************
                              3396 *
                              3397 *UVMDP1 DPL    FUNC=@DGET,DADDR=UVMDR1,CNT=UVML01,CADDR=UVMADD
                         0F80 3398 UVMDP1 EQU    *                       DISK PARAMETER LIST
     0F80 01             0F80 3399          DC    AL1(@DGET)             REQUESTED FUNCTION
     0F81 0000           0F82 3400          DC    AL2(UVMDR1)            DISK ADDRESS
     0F83 0E             0F83 3401          DC    AL1(UVML01)            SECTOR COUNT
     0F84 1100           0F85 3402          DC    AL2(UVMADD)            BUFFER ADDRESS
                              3403 *** END OF EXPANSION ***


                              3405 *UVMDP2 DPL    FUNC=@DGET,DADDR=UVMDR2,CNT=UVML01,CADDR=UVMADD
                         0F86 3406 UVMDP2 EQU    *                       DISK PARAMETER LIST
     0F86 01             0F86 3407          DC    AL1(@DGET)             REQUESTED FUNCTION
     0F87 0005           0F88 3408          DC    AL2(UVMDR2)            DISK ADDRESS
     0F89 0E             0F89 3409          DC    AL1(UVML01)            SECTOR COUNT
     0F8A 1100           0F8B 3410          DC    AL2(UVMADD)            BUFFER ADDRESS
                              3411 *** END OF EXPANSION ***


                              3413 ********************************************************************
                              3414 * VIRTUAL MEMORY DUMP ROUTINE CONSTANTS AND EQUATES               *
                              3415 ********************************************************************
                              3416 *
                              3417 * EQUATES
                              3418 *
                         1107 3419 UVMG00 EQU    X'1107'                 ENTRY TO ZDUMPV
                         000D 3420 UVMLNL EQU    13                      MESSAGE LENGTH
                         0000 3421 UVMDR1 EQU    X'0000'                 DADDR FOR DPL
                         0005 3422 UVMDR2 EQU    X'0005'                 DADDR FOR DPL
                         000E 3423 UVML01 EQU    X'0E'                   MAXIMUM EXPECTED LOAD
                         000D 3424 UVML02 EQU    X'0D'                   *
                         1100 3425 UVMADD EQU    X'1100'                 CADDR FOR BOTH DISK OPERATIONS
                              3426 *
                              3427 * CONSTANTS
                              3428 *
     0F8C 1C14           0F8D 3429 UVMZUD DC    AL2(#$ZUTM)             ZUTMON DISK ADDR
     0F8E 1BA4           0F8F 3430 UVMZDU DC    AL2(#$ZDUM)             POINTER TO ZDUMPV DISK ADDR
     0F90 C6C9D9E2E340D3C9 0F9C 3431 UVMLN1 DC    CL(UVMLNL)'FIRST LINE #.'
     0F9D D3C1E2E340D3C9D5 0FA9 3432 UVMLN2 DC    CL(UVMLNL)'LAST LINE #..'


                              3434 ********************************************************************
     0FAA 0DB9           0FAB 3435 UVMCVB DC    AL2(CVBHEX)
                              3436 ********************************************************************
                              3437 *
                              3438 *      $DL2P
```

```
                             3440+*************************************************************************
                             3441+*  5703-XM1  COPYRIGHT IBM CORP 1970                             *
                             3442+*              REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE. 120-2083 *
                             3443+*                                                                *
                             3444+*************************************************************************
                             3445+*STATUS -                                                        *
                             3446+*    VERSION 1 MODIFICATION 0                                     *
                             3447+*                                                                *
                             3448+*FUNCTION                                                         *
                             3449+*    * DL2ICS CONVERTS A RELATIVE DISK ADDRESS TO A PHYSICAL DISK  *
                             3450+*      ADDRESS AND COMBINES IT WITH A BASE ADDRESS PLACED IN DL2RAD *
                             3451+*      BY THE CALLER.                                             *
                             3452+*    * THE RELATIVE DISK ADDRESS IS A TWO BYTE CYLINDER SECTOR COUNT *
                             3453+*      IN THE CALLERS DISK PARAMETER LIST (DPL).                  *
                             3454+*    * THE COUNT IS A CYLINDER SECTOR DISPLACEMENT FROM THE BASE   *
                             3455+*      ADDRESS PLACED IN DL2RAD                                   *
                             3456+*    * DL2ICS IS USED TO PROCESS DATA ON THE FIXED OR REMOVABLE DISK *
                             3457+*      ON EITHER DRIVE AND PROVIDES THE INTERFACE TO $DISKN.       *
                             3458+*    * THE PHYSICAL DISK ADDRESS IS PLACED IN A COPY OF THE USERS DPL *
                             3459+*      IN DL2ICS AND A CALL IS MADE TO $DISKN TO PERFORM THE REQUESTED *
                             3460+*      OPERATION.                                                 *
                             3461+*                                                                *
                             3462+*ENTRY POINTS                                                     *
                             3463+*    * THE ENTRY IS DL2ICS. THE BASE REGISTER IS SAVED AND RESTORED *
                             3464+*      ON RETURN. THE INDEX REGISTER IS NOT USED.                 *
                             3465+*    * THE FORMAT OF THE CALLING SEQUENCE IS AS FOLLOWS:          *
                             3466+*         B   DL2ICS                                             *
                             3467+*         DC  AL2(PARMLT)                                        *
                             3468+*      WHERE PARMLT IS THE ADDR OF THE PARAMETER LIST TO BE PROCESSED. *
                             3469+*                                                                *
                             3470+*INPUT                                                            *
                             3471+*    * THE INPUT IS A TWO BYTE BASE DISK ADDRESS PLACED IN         *
                             3472+*      DL2RAD AND A SIX BYTE DPL. THE SAME FORMAT AS THE DPL FOR   *
                             3473+*      $DISKN EXCEPT FOR THE DISK ADDRESS WHICH IS A RELATIVE CYLINDER *
                             3474+*      AND SECTOR DISPLACEMENT FROM THE BASE ADDRESS IN DL2RAD.    *
                             3475+*                                                                *
                             3476+*OUTPUT                                                           *
                             3477+*    NONE.                                                        *
                             3478+*                                                                *
                             3479+*EXTERNAL REFERENCES                                              *
                             3480+*    $DISKN - ENTRY TO PHYSICAL DISK ROUTINE IS THE SYSTEM NUCLEUS. *
                             3481+*                                                                *
                             3482+*EXITS, NORMAL                                                    *
                             3483+*    NORMAL - EXIT IS TO THE FIRST INSTRUCTION FOLLOWING THE POINTER *
                             3484+*    TO THE DPL. THE BASE REGISTER IS RESTORED. THE RETURN ADDRESS *
                             3485+*    IS THE ADDRESS RECALL REGISTER (ARR) +2.                     *
                             3486+*                                                                *
                             3487+*EXITS, ERROR                                                     *
                             3488+*    NONE                                                         *
                             3489+*                                                                *
                             3490+*TABLES/WORK AREAS                                                *
                             3491+*    * THE CONSTANTS AND WORK AREAS RESIDE AT THE END OF THE EXECUTABLE*
                             3492+*      CODE AND ARE REFERENCED BY A DISPLACEMENT RELATIVE TO THE VALUE *
                             3493+*      IN INDEX REGISTER 1 (@BR).                                 *
                             3494+*    * DL2SEC AND DL2SAD ARE EQUATED TO OPERAND LOCATIONS IN THE    *
                             3495+*      EXECUTABLE CODE TO ELIMINATE EXCESS WORKING STORAGE.        *
```

```
                                   3496+*                                                             *
                                   3497+*ATTRIBUTES                                                   *
                                   3498+*   * DL2ICS IS REUSABLE                                      *
                                   3499+*                                                             *
                                   3500+*CHARACTER CODE DEPENDENCY                                    *
                                   3501+*    THE OPERATION OF THIS MODULE DOES NOT DEPEND UPON A PARTICULAR  *
                                   3502+*     INTERNAL REPRESENTATION OF THE EXTERNAL CHARACTER SET.  *
                                   3503+*                                                             *
                                   3504+*NOTES                                                        *
                                   3505+*   ERROR PROCEDURES                                          *
                                   3506+*     NONE                                                    *
                                   3507+*                                                             *
                                   3508+*   REGISTER USAGE                                            *
                                   3509+*    INDEX REGISTER 1 (@BR) IS SAVED AND RESTORED. THIS REGISTER IS *
                                   3510+*    USED DURING EXECUTION. REGISTER 2 (@BR) IS NOT USED.     *
                                   3511+*                                                             *
                                   3512+*   SAVED/RESTORED AREAS                                       *
                                   3513+*     NONE                                                    *
                                   3514+*                                                             *
                                   3515+*   MODIFICATION CONSIDERATIONS                               *
                                   3516+*     NONE                                                    *
                                   3517+*                                                             *
                                   3518+*   REQUIRED MODULES                                          *
                                   3519+*    @SYSEQ - COMMON SYSTEM EQUATES.                          *
                                   3520+*    @FXDEQ - SYSTEM NUCLEUS ADDRESSES AND INDICATORS VALUES EQUATES *
                                   3521+*                                                             *
                                   3522+*   OTHER                                                     *
                                   3523+*    DL2ICS MAY BE USED TO CONVERT THE DISK ADDRESS ONLY AND NOT TO *
                                   3524+*    CALL $DISKN IF THE USER MOVES A UCB CODE TO DL2SWH.      *
                                   3525+*    THIS OPTION IS NOT STANDARD USAGE.                       *
                                   3526+****************************************************************************
                      0FB0         3527+      USING DL2000,@BR              ESTABLISH ADDRESSABILITY
                                   3528+*
                      0001         3529+DL2E01 EQU   X'01'                  FIELD LENGTH OF 1
                      0002         3530+DL2E02 EQU   X'02'                  FIELD LENGTH OF 2
                      0018         3531+DL2E18 EQU   X'18'                  HEX TRACK SECTOR COUNT
                      0060         3532+DL2E60 EQU   X'60'                  PHYSICAL SECTOR COUNT
                      0083         3533+DL2TSD EQU   X'83'                  MASK OFF TRACK SPINDLE DISK
                      007C         3534+DL2E7C EQU   X'7C'                  MASK OUT SECTOR COUNT
                      0FAC         3535+DL2ICS EQU   *                      ENTRY POINT
     0FAC 34 01 102D               3536+      ST    DL2900+@OP1,@BR         SAVE OLD BASE
                      0FB0         3537+DL2000 EQU   *                      START PROCESSING
     0FB0 C2 01 0FB0               3538+      LA    DL2000,@BR              SET BASE ADDRESS
     0FB4 76 08 8A                 3539+      A     DL2C01(,@BR),@ARR       BUMP TO RIGHT BYTE OF ADDR
     0FB7 74 08 14                 3540+      ST    DL2001+@DOP2(,@BR),@ARR ADDR OF PARAM
     0FBA 76 08 8A                 3541+      A     DL2C01(,@BR),@ARR       BUMP TO RETURN ADDR
     0FBD 74 08 81                 3542+      ST    DL2910+@OP1(,@BR),@ARR  SAVE RETURN ADDR
                                   3543+*
     0FC0 4C 01 1D 0000            3544+DL2001 MVC   DL2002+@DOP2(@DADDR,@BR),*-* SETUP ADDR OF DPL
     0FC5 5E 01 1D 8C              3545+      ALC   DL2002+@DOP2(@CADDR,@BR),DL2C05(,@BR) DUMP TO RIGHT END
     0FC9 4C 05 92 0000            3546+DL2002 MVC   DL2DPL(@DPLNG,@BR),*-*   MOVE USER DPL TO WORK AREA
     0FCE 5F 00 8F 86              3547+DL2005 SLC   DL2LST+@DSAD(DL2E01,@BR),DL2C48(,@BR) ADJUST SCTR/CYL
     0FD2 F2 82 07                 3548+      JM    DL2006                   GO TO RESTORE TO CONTINUE
     0FD5 5E 00 8E 8A              3549+      ALC   DL2LST+@DCYL(DL2E01,@BR),DL2C01(,@BR) BUMP CYLINDER COUNT
     0FD9 D0 87 1E                 3550+      B     DL2005(,@BR)             BACK FOR NEXT CYLINDER
     0FDC 5E 00 8F 86              3551+DL2006 ALC   DL2LST+@DSAD(DL2E01,@BR),DL2C48(,@BR) RESTORE POSITIVE
```

```
                                  3552+*
                                  3553+*      GET THE LOGICAL SECTOR FROM THE DPL. THE NUMBER IS LEFT ADJUSTED
                                  3554+*      TO COMAE IT MTN THE POINTER ESTABLISHED PRIOR TO AN ENTRY.
   0FE0 5C 00 1D 8F               3555+       MVC   DL2SEC(DL2E01,@BR),DL2LST+@DSAD(,@BR) GET SECTOR NUMBER
   0FE4 7C 00 8F                  3556+       MVI   DL2LST+@DSAD(,@BR),@ZERO  CLEAR SECTOR BYTE
                                  3557+*
                                  3558+*      MOVE THE RELATIVE START TO THE DFL
                                  3559+*
   0FE7 5E 01 8F 94               3560+       ALC   DL2LST+@DSAD(DL2E02,@BR),DL2RAD(,@BR) DL2RAD TO DPL
   0FEB 7D 18 1D                  3561+       CLI   DL2SEC(,@BR),DL2E18         IS COUNT OVER A TRACK
   0FEE F2 82 08                  3562+       JL    DL2008                      NO GO CHANGE A PHYSICAL ADOR
   0FF1 5E 01 8F 85               3563+       ALC   DL2LST+@DSAD(DL2E02,@BR),DL2K80(,@BR) BUMP TRACK VALUE
   0FF5 5F 00 1D 88               3564+       SLC   DL2SEC(1,@BR),DL2K18(,@BR) DECR BY TRACK VALUE
   0FF9 5E 00 1D 1D               3565+DL2008 ALC   DL2SEC(1,@BR),DL2SEC(,@BR) SHIFT LEFT 1
   0FFD 5E 00 1D 1D               3566+       ALC   DL2SEC(1,@BR),DL2SEC(,@BR) SHIFT LEFT
   1001 5C 00 14 8F               3567+       MVC   DL2SAD(DL2E01,@BR),DL2LST+@DSAD(,@BR) GET SECTOR ADDRESS
                                  3568+*
                                  3569+*      ZERO OUT THE SECTOR COUNT AND LEAVE THE DISK. SPINDLE AND
                                  3570+*      TRACK BITS AS IS TO BE RE INSERTED AFTER THE SECTOR HAS BEEN
                                  3571+*      LOCATES.
                                  3572+*
   1005 7B 7C 8F                  3573+       SBF   DL2LST+@DSAD(,@BR),DL2E7C TURN OFF
   1008 7B 83 14                  3574+       SBF   DL2SAD(,@BR),DL2TSD       OFF TRACK SPINDLE DISK
   100B 5E 00 14 1D               3575+       ALC   DL2SAD(DL2E01,@BR),DL2SEC(,@BR) COMBINE SECTOR COUNTS
   100F 7D 60 14                  3576+DL2010 CLI   DL2SAD(,@BR),DL2E60       TEST IF TRACK CROSSED
   1012 F2 82 08                  3577+       JL    DL2100
                                  3578+*
                                  3579+*      INCREMENT TRACK BIT. OVERFLOW INTO THE CYLINDER COUNT.
                                  3580+*
   1015 5E 01 8F 85               3581+       ALC   DL2LST+@DSAD(DL2E02,@BR),DL2K80(,@BR)
   1019 5F 00 14 83               3582+       SLC   DL2SAD(1,@BR),DL2K60(,@BR) DECR BY TRACK VALUE
                                  3583+*
   101D 5E 00 8F 14               3584+DL2100 ALC   DL2LST+@DSAD(1,@BR),DL2SAD(,@BR) INSERT SECTOR COUNT
                                  3585+*
   1021 F2 80 06                  3586+DL2110 JC    DL2900,@NOP               CONVERSION SWITCH
              1022 3587+DL2SWH EQU   DL2110+@Q                 ADDR OF Q CODE FOR SWITCH
   1024 C0 87 0025               3588+       B     $DISKN                    GO PROCESS I/O
   1028 103D           1029 3589+       DC    AL2(DL2LST)               ADDRESS OF DPL
   102A C2 01 0000              3590+DL2900 LA    *-*,@BR                   RESTORE CALLERS BASE
   102E C0 87 0000              3591+DL2910 B     *-*
                                  3592+****************************************************************
                                  3593+*      CONSTANTS
                                  3594+****************************************************************
   1032 0060           1033 3595+DL2K60 DC    XL2'0060'                 SECTOR COUNT OF 24 LEFT ADJUSTD
   1034 0080           1035 3596+DL2K80 DC    XL2'0080'                 BIT FOR INCREMENTING TRACK
   1036 30             1036 3597+DL2C48 DC    IL1'48'                   CYLINDER VALUE FOR 1 DISK
   1037 0018           1038 3598+DL2K18 DC    XL2'18'                   HEX SECTORS PER TRACK
   1039 0001           103A 3599+DL2C01 DC    IL2'1'                    CONSTANT FOR REGISTER MODE
   103B 0005           103C 3600+DL2C05 DC    IL2'5'                    DISP TO RIGHT END OF DPL
                                  3601+****************************************************************
                                  3602+*      WORK AREA
                                  3603+****************************************************************
              103D 3604+DL2LST EQU   *                         LIST HIGH END
   103D               1042 3605+DL2DPL DS    CL(@DPLNG)                WORKING DPL
              103F 3606+DL2PHY EQU   DL2LST+@DSAD              POINTER TO PHYSICAL DADDR
              0FC4 3607+DL2SAD EQU   DL2001+@DOP2              SAVE SECTOR BYTE FROM DPI
```

```
                        0FCD 3608+DL2SEC EQU   DL2002+@DOP2            WORKING SECTOR ADDRESS FIELD
      1043              1044 3609+DL2RAD DS    CL(@DADDR)             USER RELATIVE STARTING ADDR.
                        1045 3610+DL2END EQU   *                      END OF DL2ICS
                             3611+***                       END OF DL2ICS                       ***
                             3612 *
      1051                   3613         ORG  X'1051'                ALLIGN SCANIT              1-3
                             3614 *       $CANI
```

```
                              3616+******************************************************************************
                              3617+*  5703-XM1    COPYRIGHT IBM CORP. 1970                                    *
                              3618+*                   REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083     *
                              3619+*                                                                          *
                              3620+******************************************************************************
                              3621+*STATUS                                                                    *
                              3622+*   VERSION 1 MODIFICATION 0                                                *
                              3623+*                                                                          *
                              3624+*FUNCTION                                                                   *
                              3625+*    THE FUNCTION OF SCANIT IS TO SCAN PAST VALID DELIMITERS AND           *
                              3626+*    RETURN A POINTER TO THE FIRST CHARACTER THAT'S NOT A DELIMITER.       *
                              3627+*                                                                          *
                              3628+*ENTRY POINTS                                                               *
                              3629+*    * THE ENTRY POINT IS SCANIT.                                           *
                              3630+*    * THE CALLING SEQUENCE IS AS FOLLOWS:                                  *
                              3631+*            B     SCANIT                                                   *
                              3632+*      WITH REGISTER 2 (@XR) POINTING TO THE FIRST CHARACTER TO BE         *
                              3633+*      EXAMINED.                                                            *
                              3634+*                                                                          *
                              3635+*INPUT                                                                      *
                              3636+*    NONE                                                                   *
                              3637+*                                                                          *
                              3638+*OUTPUT                                                                     *
                              3639+*    NONE                                                                   *
                              3640+*                                                                          *
                              3641+*EXTERNAL REFERENCES                                                        *
                              3642+*    $CAERR - ERROR CODE SAVE AREA                                          *
                              3643+*                                                                          *
                              3644+*EXITS, NORMAL                                                              *
                              3645+*    NORMAL EXIT FROM SCANIT IS TO THE BYTE FOLLOWING THE BRANCH TO        *
                              3646+*    SCANIT IN THE CALLING ROUTINE. THE PSR (REGISTER 4) WILL CONTAIN      *
                              3647+*    A ZERO IF NO DELIMITERS WERE FOUND OR A HIGH CONDITION IF ONE OR      *
                              3648+*    MORE DELIMITERS WERE SCANNED.                                          *
                              3649+*                                                                          *
                              3650+*EXITS, ERROR                                                               *
                              3651+*    ERROR EXIT FROM SCANIT IS TO THE BYTE FOLLOWING THE BRANCH TO         *
                              3652+*    SCANIT IN THE CALLING ROUTINE. THE PSR WILL CONTAIN A LOW             *
                              3653+*    CONDITION.                                                             *
                              3654+*                                                                          *
                              3655+*TABLES/WORKAREAS                                                           *
                              3656+*    * SCACNT - AREA CONTAINING NUMBERS OF DELIMITERS SCANNED              *
                              3657+*    * SCAMMA - LOC WHERE SCACOM MAY BE MOVED IF ONE COMMA IS ALSO         *
                              3658+*      TO BE CONSIDERED A DELIMITER. MOVING SCACOF BACK INTO SCAMMA        *
                              3659+*      INDICATES THAT ONLY BLANKS SHOULD BE CONSIDERED DELIMITERS.         *
                              3660+*                                                                          *
                              3661+*ATTRIBUTES                                                                 *
                              3662+*    RELOCATABLE AND RE-USABLE                                              *
                              3663+*                                                                          *
                              3664+*CHARACTER CODE DEPENDENCY                                                  *
                              3665+*    THE OPERATION OF THIS MODULE DOES NOT DEPEND UPON A PARTICULAR        *
                              3666+*    INTERNAL REPRESENTATION OF THE EXTERNAL CHARACTER SET.                *
                              3667+*                                                                          *
                              3668+*NOTES                                                                      *
                              3669+*    ERROR PROCEDURES                                                       *
                              3670+*       THE ONLY ERROR CONDITION DETECTED BY SCANIT IS THE CASE WHERE      *
                              3671+*       A CARRIAGE-RETURN CODE FOLLOWS A COMMA.  UPON RETURN TO THE        *
```

```
                                3672+*         CALLING ROUTINE, @PSR WILL BE SET TO A LOW CONDITION, THE      *
                                3673+*         ERROR CODE IS SET IN $CAERR, AND MG WILU BE POINTING TO THE    *
                                3674+*         CARRIAGE-RETURN CHARACTER.                                     *
                                3675+*                                                                        *
                                3676+*    REGISTER USAGE                                                      *
                                3677+*       REGISTER 2 (@XR) IS USED AS A POINTER ACROSS THE AREA BEING      *
                                3678+*       SCANNED FOR DELIMETERS.                                          *
                                3679+*                                                                        *
                                3680+*    SAVED/RESTORED AREAS                                                *
                                3681+*       UPON ENTRY TO SCANIT, REGISTER 8 (@ARR) IS SAVED AND USED AS     *
                                3682+*       THE RETURN ADDRESS.                                              *
                                3683+*                                                                        *
                                3684+*    MODIFICATION CONSIDERATIONS                                         *
                                3685+*       NONE                                                             *
                                3686+*                                                                        *
                                3687+*    REQUIRED MODULES                                                    *
                                3688+*       * @SYSEQ - COMMON SYSTEM EQUATES                                 *
                                3689+*       * @FXDEQ - FIXED NUCLEUS ADDRESSES EQUATES                       *
                                3690+*                                                                        *
                                3691+*    OTHER                                                               *
                                3692+*       SCANIT IS INITIALIZED TO BYPASS BLANKS ONLY. IF SCACOM IS        *
                                3693+*       MOVED TO SCAMMA, ONE COMMA WILL BE SCANNED ALONG WITH BLANKS.    *
                                3694+*       THE INSTRUCTION TO DO THIS IS AS FOLLOWS:                        *
                                3695+*              MVI   SCAMMA,SCACOM                                        *
                                3696+*                                                                        *
                                3697+*       TO DROP THE COMMA FROM ITS DELIMITER STATUS, SCACOF SHOULD BE    *
                                3698+*       MOVED TO SCAMMA, USING THE FOLLOWING INSTRUCTION:                *
                                3699+*              MVI   SCAMMA,SCACOF                                        *
                                3700+*                                                                        *
                                3701+************************************************************************

                                3703+*
                                3704+*                     EQUATES USED IN THIS SUBROUTINE
                                3705+*
                           0001 3706+SCAINC EQU    1                           TO INCREMENT POINTER
                           0001 3707+SCACOM EQU    @BNE                        SWITCH TO ALLOW SCANNING COMMA
                           0087 3708+SCACOF EQU    @UCB                        SWITCH TO SET OFF THE INDICATON
                                3709+*                                         * FOR SCANNING A COMMA
                           1051 3710+SCANIT EQU    *                           ENTRY POINT TO THIS SUBROUTINE
     1051 34 08 108D             3711+       ST     SCA500+@OP1,@ARR           SAVE RETURN ADDRESS
     1055 34 02 108F             3712+       ST     SCASVE,@XR                 SAVE POINTER VALUE
     1059 3C 04 03CD             3713+       MVI    $CAERR,@@E110              SET ERROR CODE
     105D F2 87 03               3714+       J      SCA200                     GO TO PROCESS
     1060 E2 02 01               3715+SCA100 LA     SCAINC(,@XR),@XR           INCREMENT POINTER TO NEXT CHAR
     1063 BD 40 00               3716+SCA200 CLI    0(,@XR),@BLANK             IS THIS CHAR BLANK ?
     1066 C0 81 1060             3717+       BE     SCA100                     YES, FETCH NEXT ONE
     106A BD 6B 00               3718+       CLI    0(,@XR),@COMMA             IS IT A COMMA ?
     106D F2 87 10               3719+SCA250 JC     SCA400,@UCB                UCS TO RETURN -- OR NOP IF
                                3720+*                                         * SCAMMA IS ACTIVE AND CHAR
     1070 E2 02 01               3721+SCA300 LA     SCAINC(,@XR),@XR           INCREMENT POINTER TO NEXT CHAR
     1073 BD 40 00               3722+       CLI    0(,@XR),@BLANK             IS THIS CHAR A BLANK ?
     1076 C0 81 1070             3723+       BE     SCA300                     YES, FETCH NEXT ONE
     107A BD 1F 00               3724+       CLI    0(,@XR),@EOS+1             IS THIS EOS ?
     107D F2 82 0A               3725+       JL     SCA500                     IF NOT, SKIP ERROR ROUTINE
     1080 34 02 1091             3726+SCA400 ST     SCACNT,@XR                 SAVE NEW POINTER VALUE
```

```
      SCANIT - DELIMETER SCAN MODULE

ERR LOC   OBJECT CODE    ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  13/01/22  PAGE  24

    1084 0F 01 1091 108F      3727+      SLC    SCACNT(2),SCASVE            SET PSR TO EQUAL IF POINTER
                              3728+*                                       * NOT ADVANCED
    108A C0 87 0000           3729+SCA500 B     *-*                        YES, RETURN
                         106E 3730+SCAMMA EQU   SCA250+@Q                   TO SET SCAN COMMA INDICATOR
                              3731+*
                              3732+*                 SAVE AREA
                              3733+*
                         108E 3734+SCASV1 EQU   *                          FIRST BYTE OF SCASVE
    108E                 108F 3735+SCASVE DS    CL2                        ORIGINAL POINTER VALUE SAVE
    1090                 1091 3736+SCACNT DS    CL2                        SAVE AREA FOR TOTAL CHAR SCAN
                              3737+***                       END OF SCANIT                        ***
```

```
      SCANIT - DELIMETER SCAN MODULE


ERR LOC  OBJECT CODE     ADDR STMT SOURCE STATEMENT           VER 15, MOD 00  13/01/22  PAGE  25

                          3739 ****************************************************************************
                          3740 *                                                                          *
                          3741 *   THIS SUBROUTINE WILL BUILD AND PASS PARAMETERS TO PROGRAMS TO          *
                          3742 *   START THIER EXECUTION WITH SPECIFIC DISK AND CORE ADDRESSES OR         *
                          3743 *   SPECIFIC VALUES.                                                       *
                          3744 *                                                                          *
                          3745 *   THIS SECTION HAS ONE ENTRY FOR EACH PROGRAM TO BE CALLLD.              *
                          3746 *   EACH ENTRY MUST DO THE FOLLOWING BEFORE A PARAMETER CAN BE BUILT:      *
                          3747 *      1- SET UP OPERATOR MESSAGES.                                        *
                          3748 *      2- SET UP EXIT TO POINT TO PROPER PROGRAM.                          *
                          3749 *      3- AND IN SOME CASES SET UP A SWITCH/ES FOR FOLLOWING PROGRAM.      *
                          3750 *                                                                          *
                          3751 *   A MESSAGE WILL PRINT AND THE KEYBOARD WILL ACCEPT THE OPERATOR'S       *
                          3752 *   ENTRY.                                                                 *
                          3753 *   THE FIRST PARAMETER WILL BE PLACED IN THE FIRST BUCKET AFTER THE       *
                          3754 *   BRANCH TO THE PROGRAM TO BE CALLED.                                    *
                          3755 *                                                                          *
                          3756 *   A SECOND MSG AND KEY ENTRY WILL FOLLOW                                 *
                          3757 *   THE CALL WILL THEN EXECUTE  B      PROGRAM                             *
                          3758 *                              DC    CL2(PARAM1)                           *
                          3759 *                              DC    CL2(PARAM2)                           *
                          3760 *                                                                          *
                          3761 ****************************************************************************
                          3762 *
                          3763 * DISK PATCH ENTRY
                          3764 *
                          3765 *ZCD005 ENTER BASE=ZCD020                ENTER FOR DISK PATCH
                     1114 3766        USING ZCD020,@BR                BASE ADDRESS SPECIFICATION
                     1092 3767 ZCD005 EQU   *                         MODULE ENTRY POINT
      1092 C2 01 1114      3768        LA    ZCD020,@BR               LOAD BASE REGISTER
                          3769 *** END OF EXPANSION ***

      1096 5C 0C 77 EC     3771        MVC   ZCDMV1(,@BR),ZCDWRT(ZCDMSG,@BR)  SET DADDR REQUEST MESSAGE
      109A 5C 0C 94 D2     3772        MVC   ZCDMV2(,@BR),ZCDDPL(ZCDMSG,@BR)  SET DISP REQUEST MESSAGE
      109E 7C 02 45        3773        MVI   ZCD062+@Q(,@BR),ZCDLL2    CHANGE LENGTH TO 2
      10A1 4C 01 5C 121A   3774        MVC   ZCD087+@OP1(@CADDR,@BR),ZCDCM4   MOVE IN BRANCH ADDR
      10A6 D0 87 00        3775        B     ZCD020(,@BR)              BRANCH TO COMMON ROUTINE
                          3776 *
                          3777 * CORE PATCH ENTRY
                          3778 *
                          3779 *ZCD006 ENTER BASE=ZCD020                ENTER FOR CORE PATCH
                     1114 3780        USING ZCD020,@BR                BASE ADDRESS SPECIFICATION
                     10A9 3781 ZCD006 EQU   *                         MODULE ENTRY POINT
      10A9 C2 01 1114      3782        LA    ZCD020,@BR               LOAD BASE REGISTER
                          3783 *** END OF EXPANSION ***

      10AD 5C 0C 94 AB     3785        MVC   ZCDMV2(,@BR),ZCDSTA(ZCDMSG,@BR)  SET START ADDR MESSAGE
      10B1 4C 01 5C 1218   3786        MVC   ZCD087+@OP1(@CADDR,@BR),ZCDCM3   MOVE IN BRANCH ADDR
      10B6 C0 87 1141      3787        B     ZCD060                   BRANCH TO SECOND ROUTINE
                          3788 *
                          3789 * DISK WRITE ENTRY
                          3790 *
                          3791 *ZUTCOP ENTER BASE=ZCD020                ENTRY FOR DISK WRITE
                     1114 3792        USING ZCD020,@BR                BASE ADDRESS SPECIFICATION
                     10BA 3793 ZUTCOP EQU   *                         MODULE ENTRY POINT
      10BA C2 01 1114      3794        LA    ZCD020,@BR               LOAD BASE REGISTER
```

```
                                3795 *** END OF EXPANSION ***

    10BE 5C 0C 77 DF            3797          MVC   ZCDMV1(,@BR),ZCDRDA(ZCDMSG,@BR)   SET READ DADDR MESSAGE
    10C2 5C 0C 94 EC            3798          MVC   ZCDMV2(,@BR),ZCDWRT(ZCDMSG,@BR)   SET WRITE DADDR MESSAGE
    10C6 4C 01 5C 1216          3799          MVC   ZCD087+@OP1(@CADDR,@BR),ZCDCOP    MOVE IN BRANCH ADDRESS
    10CB C0 87 1114             3800          B     ZCD020                            GO TO COMMON ROUTINE
                                3801 *
                                3802 * DISK COMPARE ENTRY
                                3803 *
                                3804 *ZDCOMP ENTER BASE=ZCD020               ENTRY FOR DISK COMPARE
         1114 3805              USING ZCD020,@BR                 BASE ADDRESS SPECIFICATION
         10CF 3806 ZDCOMP EQU   *                               MODULE ENTRY POINT
    10CF C2 01 1114             3807          LA    ZCD020,@BR                 LOAD BASE REGISTER
                                3808 *** END OF EXPANSION ***

    10D3 5C 0C 77 DF            3810          MVC   ZCDMV1(,@BR),ZCDRDA(ZCDMSG,@BR)   SET READ DADDR MESSAGE
    10D7 5C 0C 94 F9            3811          MVC   ZCDMV2(,@BR),ZCDCHK(ZCDMSG,@BR)   SET CHECK DADDR MESSAGE
    10DB 4C 01 5C 1214          3812          MVC   ZCD087+@OP1(@CADDR,@BR),ZCDCMP    MOVE IN BRANCH ADDRESS
    10E0 D0 87 00               3813          B     ZCD020(,@BR)
                                3814 *
                                3815 * CORE DUMP ENTRY
                                3816 *
                                3817 *ZCORED ENTER BASE=ZCD020               ENTRY FOR CORE DUMP
         1114 3818              USING ZCD020,@BR                 BASE ADDRESS SPECIFICATION
         10E3 3819 ZCORED EQU   *                               MODULE ENTRY POINT
    10E3 C2 01 1114             3820          LA    ZCD020,@BR                 LOAD BASE REGISTER
                                3821 *** END OF EXPANSION **

    10E7 5C 0C 77 AB            3823          MVC   ZCDMV1(,@BR),ZCDSTA(ZCDMSG,@BR)   SET START ADDR MESSAGE
    10EB 5C 0C 94 B8            3824          MVC   ZCDMV2(,@BR),ZCDEND(ZCDMSG,@BR)   SET END ADDR MESSAGE
    10EF 4C 01 5C 1210          3825          MVC   ZCD087+@OP1(@CADDR,@BR),ZCDCRD    MOVE IN BRANCH TO ADDR
    10F4 3B FF 120E             3826          SBF   ZCDDSK,ZCDFFF              TURN OFF DISK SWITCH
    10F8 D0 87 00               3827          B     ZCD020(,@BR)              GO TO MAIN PART
                                3828 *
                                3829 * DISK DUMP ENTRY
                                3830 *
                                3831 *ZDUMDK ENTER BASE=ZCD020               ENTRY FOR DISK DUMP
         1114 3832              USING ZCD020,@BR                 BASE ADDRESS SPECIFICATION
         10FB 3833 ZDUMDK EQU   *                               MODULE ENTRY POINT
    10FB C2 01 1114             3834          LA    ZCD020,@BR                 LOAD BASE REGISTER
                                3835 *** END OF EXPANSION ***

    10FF 5C 0C 77 C5            3837          MVC   ZCDMV1(,@BR),ZCDSEC(ZCDMSG,@BR)   SET SECTOR CNT MESSAGE
    1103 5C 0C 94 DF            3838          MVC   ZCDMV2(,@BR),ZCDRDA(ZCDMSG,@BR)   SET READ DADDR MESSAGE
    1107 4C 01 5C 1212          3839          MVC   ZCD087+@OP1(@CADDR,@BR),ZCDDMP    MOVE IN BRANCH TO ADDR
    110C 3C FF 120E             3840          MVI   ZCDDSK,ZCDFFF              TURN ON DISK SWITCH
    1110 C0 87 1141             3841          B     ZCD060                    GO TO MIDDLE OF ROUTINE
                                3842 *
                                3843 * PRINT 1ST MESSAGE FOR OPTION, GET AND VALIDATE KEYBOARD INPUT
                                3844 *
                                3845 *ZCD020 PRNT   ZCDM01                       PRINT MESSAGE FOR LOW LIMIT
    1114 C0 87 0707             3846 ZCD020 B     $$PRNT                       PRINT ON MATRIX PRINTER
    1118 121B          1119 3847          DC    AL2(ZCDM01)                  PPL ADDRESS
                                3848 *** END OF EXPANSION ***

    111A C0 87 0D31             3850          B     ZUTKEY                       GET KEY DATA
```

SCANIT - DELIMETER SCAN MODULE

```
    111E 38 FF 0D99          3851          TBN    ZUTKER,ZUT820           TEST FOR ERROR
    1122 D0 10 00            3852          BT     ZCD020(,@BR)            PRINT AGAIN IF ERROR
    1125 BD 1E 00            3853          CLI    0(,@XR),@EOS            TEST FOR NO ENTRY ?
    1128 C0 81 1114          3854          BE     ZCD020                  REPEAT MESSAGE IF NO ENTRY
                             3855 *
                             3856 * CONVERT INPUT DATA TO PACKED FORM
                             3857 *
    112C 7C 04 22            3858 ZCD030 MVI    ZCD040(,@BR),ZCDFOR     SET LENGTH ON CALL TO FOUR
    112F 74 02 24            3859          ST     ZCD050(,@BR),@XR        SET KEY DATA ADDRESS
    1132 C0 87 177D          3860          B      UCL900                  CALL PACK ROUTINE
    1136               1136 3861 ZCD040 DS     CL1                     LENGTH OF UNPACKED DATA IN HEX
    1137               1138 3862 ZCD050 DS     CL2                     ADDRESS OF DATA
    1139 C0 01 1114          3863          BNE    ZCD020                  ASK AGAIN IF IN ERROR
    113D 6C 01 5E 01         3864          MVC    ZCD090(,@BR),1(,@XR)    MOVE FIRST VALUE TO EXIT PARAM
                             3865 *
                             3866 * PRINT 2ND MESSAGE FOR OPTION, GET AND VALIDATE KEYBOARD INPUT
                             3867 *
                             3868 *ZCD060 PRNT   ZCDM02                   PRINT MESSAGE FOR HIGH LIMIT
    1141 C0 87 0707          3869 ZCD060 B      $$PRNT                  PRINT ON MATRIX PRINTER
    1145 121F          1146 3870          DC     AL2(ZCDM02)             PPL ADDRESS
                             3871 *** END OF EXPANSION ***

    1147 C0 87 0D31          3873          B      ZUTKEY                  GET KEY DATA
    114B 38 FF 0D99          3874          TBN    ZUTKER,ZUT820           TEST FOR ERROR
    114F D0 10 2D            3875          BT     ZCD060(,@BR)            PRINT AGAIN IF ERROR
    1152 BD 1E 00            3876          CLI    0(,@XR),@EOS            TEST FOR NO ENTRY ?
    1155 D0 81 2D            3877          BE     ZCD060(,@BR)            PRINT AGAIN ON AN ERROR ENTRY
                             3878 *
                             3879 * CONVERT INPUT DATA TO PACKED FORM
                             3880 *
    1158 7C 04 4E            3881 ZCD062 MVI    ZCD070(,@BR),ZCDFOR     SET LENGTH TO FOUR
    115B 74 02 50            3882          ST     ZCD080(,@BR),@XR        STORE DATA ADDRESS IN CALL
    115E C0 87 177D          3883          B      UCL900                  CALL PACK ROUTINE
    1162               1162 3884 ZCD070 DS     CL1                     LENGTH PARAM
    1163               1164 3885 ZCD080 DS     CL2                     ADDRESS PARAM
    1165 C0 01 1141          3886          BNE    ZCD060                  ASK AGAIN IF IN ERROR
    1169 6C 01 60 01         3887          MVC    ZCD100(,@BR),1(,@XR)    MOVE 2ND VALUE TO EXIT PARAM
                             3888 *
                             3889 * BRANCH TO SPECIFIC ROUTINE AS SET UPON INDIVIDUAL OPTION ENTRY
                             3890 *
    116D C0 87 0000          3891 ZCD087 B      *-*                     CALL PROGRAM
    1171               1172 3892 ZCD090 DS     CL2                     LOW LIMIT
    1173               1174 3893 ZCD100 DS     CL2                     HIGH LIMIT
                             3894 *
                             3895 * RETURN TO SELECTION ROUTINE TO REPEAT OPTION REQUEST
                             3896 *
    1175 C0 87 0C39          3897          B      ZUT020                  RETURN TO SELECTION ROUTINE

                             3899 *************************************************************************
                             3900 * PARAMETER PASSING ROUTINE CONSTANTS AND EQUATES                      *
                             3901 *************************************************************************
                             3902 *
                             3903 * EQUATES USED
                             3904 *
    001D 3905 ZCDL01 EQU    29                      MESSAGE LENGTH
    001D 3906 ZCDL02 EQU    29                      MESSAGE LENGTH
```

```
                         0004 3907 ZCDFOR EQU    X'04'                      CONVERSION LENGTH
                         0002 3908 ZCDLL2 EQU    X'02'                      CONVERSION LENGTH SPECIFICATION
                         00FF 3909 ZCDFFF EQU    X'FF'                      DISK DUMP MASK
                         000D 3910 ZCDMSG EQU    13                         VARIABLE MESSAGE LENGTH
                              3911 *
                         0006 3912 ZCDBN6 EQU    6                          LENGTH OF ENTER MESSAGE
                         000A 3913 ZCDB10 EQU    10                         LENGTH OF PERIOD MESSAGE
                              3914 *
                              3915 * MESSAGE CONSTANTS FOR PARAMETER PASSING ROUTINE
                              3916 *
                         1179 3917 ZCDMM1 EQU    *                          *
   1179 C5D5E3C5D940      117E 3918        DC     CL(ZCDBN6)'ENTER '
   117F                   118B 3919 ZCDMV1 DS     CL(ZCDMSG)                AREA FOR FIRST MESSAGE
   118C 4B4B4B4B4B4B4B4B  1195 3920        DC     CL(ZCDB10)'......... '

                         1196 3922 ZCDMM2 EQU    *                          *
   1196 C5D5E3C5D940      119B 3923        DC     CL(ZCDBN6)'ENTER '
   119C                   11A8 3924 ZCDMV2 DS     CL(ZCDMSG)                AREA FOR SECOND MESSAGE
   11A9 4B4B4B4B4B4B4B4B  11B2 3925        DC     CL(ZCDB10)'......... '

   11B3 E2E3C1D9E340C1C4  11BF 3927 ZCDSTA DC     CL(ZCDMSG)'START ADDRESS'
   11C0 C5D5C440C1C4C4D9  11CC 3928 ZCDEND DC     CL(ZCDMSG)'END ADDRESS..'
   11CD E2C5C3E3D6D940C3  11D9 3929 ZCDSEC DC     CL(ZCDMSG)'SECTOR COUNT.'
   11DA C4C9E2D7D3C1C3C5  11E6 3930 ZCDDPL DC     CL(ZCDMSG)'DISPLACEMENT.'
   11E7 D9C440C4C9E2D240  11F3 3931 ZCDRDA DC     CL(ZCDMSG)'RD DISK ADDR.'
   11F4 E6D940C4C9E2D240  1200 3932 ZCDWRT DC     CL(ZCDMSG)'WR DISK ADDR.'
   1201 C3C8D240C4C9E2D2  120D 3933 ZCDCHK DC     CL(ZCDMSG)'CHK DISK ADDR'
   120E                   120E 3934 ZCDDSK DS     CL1                       DISK DUMP SWITCH
                              3935 *
                              3936 * ADDRESS CONSTANTS FOR BRANCH TO ROUTINES
                              3937 *
   120F 12F4              1210 3938 ZCDCRD DC     AL2(UDUMPC)               *
   1211 1AB6              1212 3939 ZCDDMP DC     AL2(UDUMPD)               *
   1213 1B74              1214 3940 ZCDCMP DC     AL2(ZDCENT)               POINTER TO COMPARE
   1215 1B4A              1216 3941 ZCDCOP DC     AL2(ZDCCOP)               POINTER TO COPY
   1217 1842              1218 3942 ZCDCM3 DC     AL2(CPATCH)               POINTER TO CORE PATCH
   1219 184C              121A 3943 ZCDCM4 DC     AL2(DPATCH)               POINTER TO DISK PATCH
                              3944 *
                              3945 * PPL'S FOR PRINTING REQUESTS
                              3946 *
                              3947 *ZCDM01 PPL FUNC=APRINT,CNT=ZCD101,CADDR=ZCDMM1
                         121B 3948 ZCDM01 EQU    *                          PPL ADDRESS
   121B 40               121B 3949        DC     AL1(@PRINT)               FUNCTION REQUESTED
   121C 1D               121C 3950        DC     AL1(ZCDL01)               PRINT COUNT
   121D 1179             121E 3951        DC     AL2(ZCDMM1)               DATA ADDRESS
                              3952 *** END OF EXPANSION ***

                              3954 *ZCDM02 PPL FUNC=@PRINT,CNT=ZCDLO2,CADDR=ZCDMM2
                         121F 3955 ZCDM02 EQU    *                          PPL ADDRESS
   121F 40               121F 3956        DC     AL1(@PRINT)               FUNCTION REQUESTED
   1220 1D               1220 3957        DC     AL1(ZCDL02)               PRINT COUNT
   1221 1196             1222 3958        DC     AL2(ZCDMM2)               DATA ADDRESS
                              3959 *** END OF EXPANSION ***
                              3960 *************************************************************************
```

```
                              3962 ********************************************************************
                              3963 *   MEMORY REAL OR SAVED ROUTINE, WORKS AS FOLLOWS:                *
                              3964 *   EACH TIME CORE DUMP OR CORE PATCH CROSSES A BOUNDARY OF X'00' IT *
                              3965 *   MUST BE DETERMINED IF NEXT AREA OF CORE IS SAVED OR NOT.  IF SAVED *
                              3966 *   CORE IS USED THIS DATA MUST BE BROUGHT IN FROM THE SAVE AREA.    *
                              3967 *   THE FIRST PART OF THIS ROUTINE IS USED TO ESTABLISH THE LIMITS   *
                              3968 *   OF THE SAVED CORE AND TO DETERMINE UPPER LIMIT OF STORAGE TO BE  *
                              3969 *   DUMPED, THIS INFORMATION IS COMPUTED ON THE BASIS OF CORE SAVE DPL *
                              3970 *                                                                  *
                              3971 *   CALLING SEQUENCE IS AS FOLLOWS - - -                            *
                              3972 *                         MVC ZCSADD,XXXXXX       WHERE XXXXXX IS THE *
                              3973 *             --- OR -- ST  ZCSADD,REGIST      CURRENT ADDRESS IN USE *
                              3974 *             -- AND -- B   ZCSAVE                                   *
                              3975 *   UPON RETURN FROM ZCSAVE THE FOLLOWING CONDITIONS CAN EXIST:      *
                              3976 *                                                                  *
                              3977 *    1, IF ZCSADD IS LESS THAN START OF THE SAVED AREA, ZCSREL = X'FF' *
                              3978 *    2, IF ZCSADD IS GREATER THAN END OF CORE, ZCSREL = X'F0'        *
                              3979 *    3, IF ZCSADD IS IN CRT AREA BUT NOT ABOVE END, ZCSREL = X'FF'   *
                              3980 *    4, IF ZCSADD IS IN THE SAVED CORE, ZCSREL = X'00' AND 256 BYTES *
                              3981 *       OF SAVED DATA ARE IN AREA WITH LEFT END POINTER OF ZCSDAT.   *
                              3982 *                                                                  *
                              3983 ********************************************************************
                         1223 3984 ZCSAVE EQU   *                         ENTRY POINT
                              3985 *ZCS010 ENTER BASE-ZCS010,EXIT-ZCS0U,@BR,,@ARR
                         1223 3986        USING ZCS010,@BR             BASE ADDRESS SPECIFICATION
                         1223 3987 ZCS010 EQU   *                      MODULE ENTRY POINT
    1223 34 01 12B8           3988        ST    ZCSOU0+@OP1,@BR        SAVE @BR
    1227 C2 01 1223           3989        LA    ZCS010,@BR             LOAD BASE REGISTER
    122B 74 08 99             3990        ST    ZCSOU2+@OP1(,@BR),@ARR  SAVE RETURN ADDRESS
                              3991 *** END OF EXPANSION ***

    122E D0 80 44             3993 ZCS020 BC    ZCSTWO(,@BR),@NOP       FIRST TIME SWITCH
    1231 7C 87 0C             3994        MVI   ZCS020+@Q(,@BR),@UCB    FLIP SWITCH ON
    1234 4C 01 9B 0513        3995        MVC   ZCSLOW(@CADDR,@BR),$CSDPL+@DBFR2  MOVE IN LOW CADDR
    1239 5C 01 9D 9B          3996        MVC   ZCSHGH(,@BR),ZCSLOW(@CADDR,@BR)  MOVE LOW LIMIT TO HIGH
    123D 4C 00 9E 0511        3997        MVC   ZCSSCT(,@BR),$SWPCR(1)  MOVE SECTOR COUNT TO COUNTER
    1242 5E 01 9D A7          3998 ZCS030 ALC   ZCSHGH(,@BR),ZCSHUN(@CADDR,@BR)  ADD 256 TO HIGH LIMIT
    1246 5F 00 9E AB          3999        SLC   ZCSSCT(,@BR),ZCSONE(1,@BR)  REDUCE SECTOR COUNT BY ONE
    124A D0 01 1F             4000        BNZ   ZCS030(,@BR)            GO BACK TO BUILD HIGH LIMIT
                              4001 *                                   * UNTIL SECTOR CNT IS GONE WHEN
                              4002 *                                   * ZCSSCT = ZERO, ZCSHGH = SAVE+1
    124D 5C 01 A5 9D          4003        MVC   ZCSEND(,@BR),ZCSHGH(@CADDR,@BR) MOVE HIGH TO END, MAY CHNG
    1251 79 FF 9D             4004        TBF   ZCSHGH(,@BR),ZCSFFF     TEST LOW ORDER BYTE FOR X'00'
    1254 F2 90 06             4005        JF    ZCS040                  GO COMPUTE ACTUAL END
    1257 79 0F 9C             4006        TBF   ZCSHGH-1(,@BR),ZCSOFF   TEST HIGH ORDER FOR X'XO'
    125A F2 10 0A             4007        JT    ZCSTWO                  CONTINUE WITH STORAGE TEST
    125D 7B FF A5             4008 ZCS040 SBF   ZCSEND(,@BR),ZCSFFF     TURN OFF LOW ORDER BITS
    1260 7B 0F A4             4009        SBF   ZCSEND-1(,@BR),ZCSOFF   TURN OFF LOW ORDER
                              4010 *                                   * HIGH ORDER BYTE
    1263 5E 01 A5 A9          4011        ALC   ZCSEND(,@BR),ZCSTHO(@CADDR,@BR)  ADD HEX 1000 TO HIGH LIM
                         1267 4012 ZCSTWO EQU   *                      START HERE IF LOW AND HIGH AND
                              4013 *                                   END ADDRESS ARE COMPUTED
                              4014 *   AT THIS POINT:
                              4015 *           LOW LIMIT IS IN ZCSLOW
                              4016 *           HIGH LIMIT IS IN ZCSHGH
                              4017 *           DUMP END ADDR IS IN ZCSEND
```

```
      SCANIT - DELIMETER SCAN MODULE


ERR LOC   OBJECT CODE     ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  13/01/22  PAGE  30

                          4018 *
     1267 7A FF A1         4019       SBN   ZCSREL(,@BR),ZCSFFF      TURN ON REAL CORE SWITCH
     126A 5D 01 A0 9B      4020       CLC   ZCSADD(,@BR),ZCSLOW(@CADDR,@BR) TEST ADDR FOR LESS THAN
     126E D0 82 92         4021       BL    ZCSOUT(,@BR)            EXIT IF LOWER CORE
     1271 5D 01 A0 A5      4022       CLC   ZCSADD(@CADDR,@BR),ZCSEND(,@BR)  TEST IF ABOVE END OF CORE
     1275 D0 82 58         4023       BL    ZCS050(,@BR)            CONTINUE TO NEXT TEST
     1278 7B 0F A1         4024       SBF   ZCSREL(,@BR),ZCSOFF     TURN HALF OFF TO INDICATE ERROR
     127B 5D 01 A0 9D      4025 ZCS050 CLC  ZCSADD(,@BR),ZCSHGH(,@BR) TEST FOR SAVE AREA END
     127F D0 02 92         4026       BNL   ZCSOUT(,@BR)            EXIT IF GREATER
     1282 7B FF A1         4027       SBF   ZCSREL(,@BR),@DWAIT     ELSE INDICATE SAVED CORE
     1285 0C 01 1044 0510  4028       MVC   DL2RAD,$CSDPL+@DSAD(@DADDR)  MOVE IN SAVE CORE DADDR
     128B 5C 01 A3 A0      4029       MVC   ZCSCTR(,@BR),ZCSADD(@CADDR,@BR)  MOVE ADDR TO COUNTER
     128F 5F 01 A3 9B      4030       SLC   ZCSCTR(,@BR),ZCSLOW(@CADDR,@BR)  SUBTRACT LOW LIMIT
     1293 5F 00 9E 9E      4031       SLC   ZCSSCT(1,@BR),ZCSSCT(,@BR)  ZERO SECTOR COUNT
     1297 5F 01 A3 A7      4032 ZCS060 SLC  ZCSCTR(,@BR),ZCSHUN(@CADDR,@BR)  TAKE AWAY HEX'100'
     129B F2 82 07         4033       JM    ZCS070                  EXIT WHEN DISP = ZERO
     129E 5E 00 9E AB      4034       ALC   ZCSSCT(,@BR),ZCSONE(1,@BR)  ADD ONE TO SECTOR DISPL
     12A2 D0 87 74         4035       B     ZCS060(,@BR)            RETURN FOR MORE REDUCTION

     12A5 5C 00 AE 9E      4037 ZCS070 MVC  ZCSDPD(,@BR),ZCSSCT(1,@BR)  MOVE COUNT TO DPL
                          4038 *        DSKL2 ZCSDPL,WAIT          GET DISK
     12A9 C0 87 0FAC       4039       B     DL2ICS                  PERFORM RELATIVE DISK OP
     12AD 12CF        12AE 4040       DC    AL2(ZCSDPL)             DPL ADDRESS
     12AF C0 87 0025       4041       B     $DISKN                  WAIT AND CHECK DISK ERRORS
     12B3 057F        12B4 4042       DC    AL2($WAITF)             WAIT DPL ADDRESS
                          4043 *** END OF EXPANSION ***
                     12B5 4044 ZCSOUT EQU  *                        *
                          4045 *ZCSOU  EXIT  @BR,,RETURN            RETURN TO CALLER
     12B5 C2 01 0000       4046 ZCSOU0  LA   *-*,@BR                RESTORE @BR
     12B9 C0 87 0000       4047 ZCSOU2  B    *-*                    RETURN TO CALLER
                          4048 *** END OF EXPANSION ***
                          4049 *   END OF SECTION CODE


                          4051 ***********************************************************************
                          4052 *   WORKING STORAGE FOR MEMORY REAL OR SAVE ROUTINE
                          4053 ***********************************************************************
     12BD            12BE 4054 ZCSLOW DS   CL2                      BUCKET FOR LOWEST CORE SAVED
     12BF            12C0 4055 ZCSHGH DS   CL2                      BUCKET FOR HIGH LIMIT OF CORE
     12C1            12C1 4056 ZCSSCT DS   CL1                      SECTOR COUNTER
     12C2            12C3 4057 ZCSADD DS   CL2                      CORE ADDRESS IN USE ON DUMP
                          4058 *                                    OR PATCH CURRENTLY IN USE
     12C4            12C4 4059 ZCSREL DS   CL1                      SWITCH BYTE TO DETERMINE STATUS
     12C5            12C6 4060 ZCSCTR DS   CL2                      *
     12C7            12C8 4061 ZCSEND DS   CL2                      *
                          4062 ***********************************************************************1
                          4063 *   SECTION EQUATES
                          4064 ***********************************************************************1
     000F            4065 ZCSOFF EQU   X'0F'                        *
                          4066 ***********************************************************************O
                          4067 *   CONSTANTS FOR MEMORY REAL OR SAVED ROUTINE
                          4068 ***********************************************************************1
                          4069 *
                          4070 * EQUATES
                          4071 *
     0001            4072 ZCSSEC EQU   X'01'                        SECTOR COUNT
     1F00            4073 ZCSDAT EQU   X'1F00'                      BUFFER LOCATION
```

```
      SCANIT - DELIMETER SCAN MODULE

ERR LOC  OBJECT CODE      ADDR STMT SOURCE STATEMENT            VER 15, MOD 00  13/01/22  PAGE  31

                          00FF 4074 ZCSFFF EQU   X'FF'                     REAL/SAVED CORE MASK
                               4075 *
                               4076 * CONSTANTS
                               4077 *
     12C9 0100               12CA 4078 ZCSHUN DC    XL2'0100'
     12CB 1000               12CC 4079 ZCSTHO DC    XL2'1000'                  THOUSAND HEX FOR 4K IN DEC
     12CD 0001               12CE 4080 ZCSONE DC    XL2'0001'
                               4081 *ZCSDPL DPL    FUNC=@DGET,DADDR=*-*,CNT=ZCSSEC,CADDR=ZCSDAT
                          12CF 4082 ZCSDPL EQU   *                         DISK PARAMETER LIST
     12CF 01               12CF 4083        DC    AL1(@DGET)                REQUESTED FUNCTION
     12D0 0000             12D1 4084        DC    AL2(*-*)                  DISK ADDRESS
     12D2 01               12D2 4085        DC    AL1(ZCSSEC)               SECTOR COUNT
     12D3 1F00             12D4 4086        DC    AL2(ZCSDAT)               BUFFER ADDRESS
                               4087 *** END OF EXPANSION ***
                          12D1 4088 ZCSDPD EQU   ZCSDPL+@DSAD              DISK ADDRESS
```

UDUMPC - CORE-DISK DUMPS

```
                              4090 *************************************************************************
                              4091 *   UDUMPC WILL DUMP THE CONTENTS OF CORE SPECIFIED IN A HEX-CHARACTER *
                              4092 *   FORMAT, AND WILL PRINT THE DUMP ON THE PRINTER.   IT CAN BE        *
                              4093 *   CALLED WITH TWO PARAMETERS WHICH SPECIFY THE BEGINNING ADDRESS AND *
                              4094 *   THE ENDING ADDRESS OF THE DUMP.   UDUMPC CALLS TWO OTHER ROUTINES. *
                              4095 *   ONE IS CVBHEX WHICH CONVERTS THE INTERNAL BINARY CONFIGURATION     *
                              4096 *   INTO HEXADECIMAL CHARACTERS.   THE OTHER IS ZCSAVE WHICH DETERMINES *
                              4097 *   IF REAL OR SAVED CORE WILL BE USED TO DUMP THE AREA IN QUESTION.   *
                              4098 *   UDUMPC WILL ALSO PRINT THE CONTENTS OF THE XR, BR, AND THE PSR.    *
                              4099 *   THE CALLING SEQUENCE IS:                                           *
                              4100 *                                                                      *
                              4101 *        B     UDUMPC                                                  *
                              4102 *        DC    AL2(PARAM1)                                             *
                              4103 *        DC    AL2(PARAM2)                                             *
                              4104 *                                                                      *
                              4105 *   WHERE PARAM1 IS THE LOW ADDRESS AND PARAM2 IS THE HIGH ADDR OF     *
                              4106 *   THE DUMP.                                                          *
                              4107 *************************************************************************
                              4108 *
                              4109 * EQUATES FOR UDUMPC
                              4110 *
      0020 4111 UDC050 EQU    32
      005C 4112 UDC060 EQU    C'*'                           BOUNDARY FOR INTERPRETIVE FIELD
      00FF 4113 UDC089 EQU    X'FF'                          MASK FOR ALL BITS
      001F 4114 UDC090 EQU    X'1F'                          MASK FOR START ADDR
      0000 4115 UDC091 EQU    X'0000'                        TO SET INDICATORS TO ZERO
      0007 4116 UDC093 EQU    X'07'                          INITIALIZE PTR DISP TO OUTPUT
      0037 4117 UDC094 EQU    X'37'                          INITIALIZE PTR DISP TO INPUT
      0040 4118 UDC095 EQU    X'40'                          BLANK TO PROPOGATE THROUGH FLD
      0006 4119 UDC101 EQU    6                              LNG OF REG CONSTANTS FOR FLD
      0008 4120 UDC103 EQU    8                              LNG OF PL1IAR AND XR1 CONSTANT
      0033 4121 UDCIML EQU    51                             IDENTICAL LINE MESSAGE LENGTH
      0003 4122 UDCDP3 EQU    3                              DISP TO END ADDRESS
      0046 4123 UDC106 EQU    70                             DISP IN OUTPUT LINE FOR XR1
      004B 4124 UDC107 EQU    75                             DISP IN OUTPUT FLD FOR XR2
      0050 4125 UDC108 EQU    80                             DISP IN OUTPUT FLD FOR PSR
      0045 4126 UDC111 EQU    69
      0004 4127 UDC112 EQU    4                              SPACE BEFORE HEX FIELD
      0007 4128 UDC120 EQU    7                              DISP TO START OF HEX PRINT FLD
      0077 4129 UDC127 EQU    119                            2ND * COLUMN
      0007 4130 UDC130 EQU    7                              DISP FOR XR1 CON
      0011 4131 UDC131 EQU    17                             DISP FOR XR2 CON
      001B 4132 UDC132 EQU    27                             DISP FOR PSR CON
      0038 4133 UDC137 EQU    56                             DISP FOR CONVERTED HEX FIELD
      0057 4134 UDC143 EQU    87
      04F2 4135 UDC144 EQU    $PSDXR                         POINT TO SAVED XR
      04FA 4136 UDC145 EQU    $PSDBR                         POINT TO SAVED BR
```

```
                               4138 *
                               4139 *   SET THE BASE REGISTER FOR THE FIRST TWO SECTIONS OF UDUMPC
                               4140 *
                          13F2 4141       USING UDC453,@BR
                               4142 *
                               4143 *   MOVE THE CONTENTS OF THE PERTINENT REGISTERS INTO THE RIGHT HALF OF
                               4144 *   THE OUTPUT FIELD SPACED AT THE PROPER INTERVALS SO THAT ONE CALL TO
                               4145 *   CVBHEX, THE ROUTINE CHANGING HEX TO CHARS, WILL PT THEM IN THE
                               4146 *   PRE-PLANNED SPACES IN THE OUTPUT FIELD.
                               4147 *
    12D5 3C 80 1360            4148 UDC200 MVI   UDC300+@Q,@NOP            YES, PRINT HEADER
    12D9 D0 87 A7             4149        B     UDC470(,@BR)             CONTINUE W/O HEADER

    12DC 4F 01 00 162B         4151 UDC210 SLC   UDC453(2,@BR),UDC818     POINT TO LAST LINE
    12E1 3D 01 14FA            4152        CLI   UDC740,1                 TEST SKIP INDICATOR
    12E5 C0 81 13D2            4153        BE    UDC451                   PRINT LAST LINE
    12E9 C0 84 13A2            4154        BH    UDC410                   PRINT IDENT MSG AND LAST LINE
    12ED D0 87 AB             4155        B     UDC497(,@BR)             RETURN

    12F0 3C 87 1360            4157 UDUMP1 MVI   UDC300+@Q,@UCB           NO PAGE HEADER
    12F4 0C 01 154B 04F2       4158 UDUMPC MVC   UDC749+UDC107,UDC144(@CADDR)   PUT SAVED XR IN OUTPUT
    12FA 34 08 1319            4159        ST    UDC215+@OP1,@ARR         SAVE ARR FOR LATER
    12FE 34 02 14A0            4160        ST    UDC497+@OP1,@XR          SAVE INDEX REGISTER
    1302 C2 02 1500            4161        LA    UDC749,@XR               LOAD LEFT ADDR REG OUTPUT FLD
    1306 8C 01 46 04FA         4162        MVC   UDC106(@CADDR,@XR),UDC145 LOAD SAVED BR IN OUTPUT
    130B 34 01 14A4            4163        ST    UDC498+@OP1,@BR          SAVE BASE REGISTER
    130F C2 01 13F2            4164        LA    UDC453,@BR               LOAD BASE REG ADDRESS
    1313 B4 04 50             4165        ST    UDC108(,@XR),@PSR        PUT PSR CONTENTS IN OUTPUT FLD
                               4166 *
                               4167 *   PUT THE ADDRS OF THE START AND END ADDR INTO WORK AREAS.  THE START
                               4168 *   AND END ADDR MUST BE SPECIFIED FOLLOWING THE BRANCH TO THIS ROUTINE.
                               4169 *
    1316 C2 02 0000            4170 UDC215 LA    *-*,@XR                  SAVE RETURN ADDR
    131A 6C 01 00 01           4171        MVC   UDC453(,@BR),1(@CADDR,@XR)  MOVE START DDDR
    131E 2C 01 14F9 03         4172        MVC   UDC720,UDCDP3(@CADDR,@XR) MOVE END ADDR
    1323 E2 02 04             4173        LA    UDC112(,@XR),@XR         POINT TO RETURN
    1326 74 02 B6             4174        ST    UDC499+@OP1(,@BR),@XR    SET RETURN ADDR
                               4175 *
                               4176 *   SET THE STARTING ADDR OF THE DUMP TO THE NEXT LOWER MULTIPLE OF 32
                               4177 *   SO THAT THE FAR LEFT BYTE OF EACH LINE IS A MULTIPLE OF X'20'
                               4178 *
    1329 7B 1F 00             4179        SBF   UDC453(,@BR),UDC090      SET ADDR TO 32-BYTE BOUNDARY
    132C 3C FF 171E            4180        MVI   ZCTFTS,ZCTFFF            SET ON FIRST TIME IN CORE TEST
                               4181 *
                               4182 *   CONVERT THE REGISTERS CONTENTS TO HEX, MOVE THE CONSTNTS INTO THE
                               4183 *   OUTPUT FIELD AND PRINT.
                               4184 *
    1330 C0 87 0DB9            4185 UDC230 B     CVBHEX                   CNURT REG OF OUTPUT FLD TO CHA
    1334 0C            1334    4186        DC    XL1'0C'
    1335 1545          1336    4187        DC    AL2(UDC749+UDC111)       LEFT ADDRESS OF INPUT FIELD
    1337 1584          1338    4188        DC    AL2(UDC751+UDC103)       LEFT ADDR OF OUTPUT FIELD
    1339 C2 02 157C            4189        LA    UDC751,@XR               RESET XR
    133D 8C 07 07 1639         4190        MVC   UDC130(UDC103,@XR),UDC830 MOVE XR1 CONSTANT INTO OUTPUT
    1342 8C 05 11 163F         4191        MVC   UDC131(UDC101,@XR),UDC840 MOVE XR2 CON INTO OUTPUT FIELD
    1347 8C 05 1B 1645         4192        MVC   UDC132(UDC101,@XR),UDC850 MOVE PSR CON INTO OUTPUT FIELD
    134C 3C 40 15F3            4193        MVI   UDC752,C' '              PUT BLANK IN HIGH ORDER BYTE
```

```
     1350 0C 56 15F2 15F3        4194          MVC   UDC752-@B1(UDC143),UDC752 CLEAR TO BLANKS
                                 4195 *
                                 4196 * ON FHE FIRST PASS SKIP THIS SECTION COMPARING THE HEX FIELD WITH
                                 4197 * THE LAST BYTE OF THE PREVIOUS LINE.
                                 4198 *
     1356 3B FF 14FA             4199          SBF   UDC740,UDC089              INDICATE LAST LINE NOT PRINTED
     135A 4C 01 AA 16C7          4200          MVC   UDC470+@OP1(@CADDR,@BR),UDC900  SET BRANCH ADDR
     135F C0 80 12D5             4201 UDC300 BC    UDC200,X'80'                CONDT SKIP HEADER - 0 INITL'D
     1363 D0 87 89               4202          B     UDC461(,@BR)                *
```

```
     UDUMPC - CORE-DISK DUMPS


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  13/01/22  PAGE  35
                           4204 *                                                            *
                           4205 *   THIS SECTION OF CODE IS TO PERFORM A CHECK ON EACH LINE TO      *
                           4206 *   DETERMINE WHETHER OR NOT ALL BYTES OF THIS LINE OF 64 HEX CHAR-   *
                           4207 *   ACTERS ARE THE SAME AS THE LAST BYTE OF THE LAST LINE PRINTED.    *
                           4208 *                                                            *
                           4209 *                                                            *
                           4210 *   IF THE STARTING ADDR IS EQUAL TO THE END ADDR THEN RETURN TO      *
                           4211 *   CALLING PROGRAM.                                         *
                           4212 *                                                            *
     1366 1D 01 14F9 00    4213 UDC400 CLC   UDC720(@CADDR),UDC453(,@BR) COMPARE LOW AND HIGH ADDRS
     136B C0 04 12DC       4214        BNH   UDC210                      END OF DUMP
                           4215 *
                           4216 *   CHECK EACH BYTE OF THE CURRENT LINE IF ONE BYTE IS NOT THE SAME
                           4217 *   THEN PROCEED TO UDC410 TO DETERMINE WHAT TO PRINT.
                           4218 *
     136F C0 87 170A       4219        B     ZCTEST
     1373 75 02 00         4220        L     UDC453(,@BR),@XR            ADDR OF LEFT BYTE OF LINE
     1376 3C 00 137C       4221        MVI   UDC402+@D1,0               ZERO LINE COUNTER
     137A BD 00 00         4222 UDC402 CLI   *-*(,@XR),*-*              LAST BYTE SAME AS THIS LINE ?
     137D C0 01 13A2       4223        BNE   UDC410                     NO, THEN FORM PRINT LINE
     1381 0E 00 137C 1629  4224        ALC   UDC402+@D1(1),UDC817       INCREMENT BYTE COMPARED BY 1
     1387 3D 20 137C       4225        CLI   UDC402+@D1,UDC050          IF BYTE IS ON 32-BYTE BOUNDARY
     138B C0 82 137A       4226        BL    UDC402                     LOOP TO NEXT BYTE
     138F C0 87 175D       4227        B     ZCTRST                     LINK TO RESTORE CORE ADDR
                           4228 *
                           4229 *   IF THE BYTES OF THE LINE ARE THE SAME AS THE LAST BYTE OF THE LAST
                           4230 *   LINE THEN LOOP TO PROCESS THE NEXT LINE OR 32 BYTES IN CORE.
                           4231 *
     1393 0E 00 14FA 1629  4232 UDC405 ALC   UDC740(1),UDC817           INCR IDENT LINE INDICATOR
     1399 4E 01 00 162B    4233        ALC   UDC453(@CADDR,@BR),UDC818 INCREMENT ADDRESS
     139E C0 87 1366       4234        B     UDC400                     PROCESS THE NEXT LINE
                           4235 *
                           4236 *   IF A BYTE IS NOT THE SAME AND THE LAST LINE WAS PRINTED, THAT IS
                           4237 *   THE PREVIOUS 32 BYTES WERE NOT THE SAME AS THE 'LASTBYTE', THEN
                           4238 *   PROCEED TO PRINT THE CONTENTS OF THESE 32 BYTES.
                           4239 *
     13A2 C0 87 175D       4240 UDC410 B     ZCTRST                     LINK TO RESTORE CORE ADDR
     13A6 3D 00 14FA       4241        CLI   UDC740,@ZERO               IS SKIP INDICATOR SET TO ZERO
     13AA C0 81 13D2       4242        BE    UDC451                     YES, FORM A PRINT LINE
                           4243 *
                           4244 *   IF A BYTE IS NOT THE SAME AND THE LAST LINE WAS NOT PRINTED, THAT
                           4245 *   IS THE PREVIOUS 32 BYTES WERE THE SAME AS THE 'LASTBYTE', THEN PRINT
                           4246 *   THAT THE LAST LINE OR LINES WERE THE SAME BEFORE PROCEEDING TO PRINT
                           4247 *   THE CONTENTS OF THIS LINE.
                           4248 *
     13AE 3C 40 1577       4249        MVI   UDC750,UDC095              MOVE BLANK INTO RIGHT BYTE-FLD
     13B2 0C 76 1576 1577  4250        MVC   UDC750-@B1(UDC127),UDC750 PROPOGATE BLANKS THROUGH FIELD
     13B8 0C 32 1545 1626  4251        MVC   UDC749+UDC111(UDCIML),UDC810  SET IDENTICAL LENGTH MSG
                           4252 *     PRNT  UDC746                     CALL PRINTER
     13BE C0 87 0707       4253        B     $$PRNT                     PRINT ON MATRIX PRINTER
     13C2 14FC         13C3 4254        DC    AL2(UDC746)                PPL ADDRESS
                           4255 *** END OF EXPANSION ***

                           4257 *     PRNT  $WAITF                     CALL PRINT WAIT
     13C4 C0 87 0707       4258        B     $$PRNT                     PRINT ON MATRIX PRINTER
     13C8 057F         13C9 4259        DC    AL2($WAITF)                PPL ADDRESS
```

```
     UDUMPC - CORE-DISK DUMPS

ERR LOC   OBJECT CODE    ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  13/01/22  PAGE  36

                            4260 *** END OF EXPANSION ***
     13CA 3B FF 14FA        4261       SBF   UDC740,UDC089              ZERO SKIP INDICATOR
     13CE C0 87 0DA5        4262       B     ZUTIRI                     * GO TEST FOR INTERRUPTS
```

```
                              4264 *
                              4265 *   THIS SECTION PROCESSES 32 BYTES INTO AN OUTPUT FIELD
                              4266 *
     13D2 C2 02 1500          4267 UDC451 LA    UDC749,@XR                POINT TO OUTPUT FIELD
     13D6 4C 01 AA 16C9       4268        MVC   UDC470+@OP1(@CADDR,@BR),UDC901  SET BRANCH TO CHECK LINE
     13DB 3C 00 14FA          4269        MVI   UDC740,@ZERO              RESET SKIP INDICATOR
                              4270 *
                              4271 *   CONVERT HEX FIELD TO HEX CHARACTERS
                              4272 *
     13DF C0 87 0DB9          4273 UDC452 B     CVBHEX                    CONVERT TU HEX THE 2 BYTE ADDR*
     13E3 02            13E3  4274        DC    XL1'02'                   LENGTH IN BYTES
     13E4 13F1          13E5  4275        DC    AL2(UDC453-1)             ADDR OF INPUT FIELD
     13E6 1500          13E7  4276        DC    AL2(UDC749)               LEFT ADDR OF OUTPUT FIELD
                              4277 *
                              4278 *   PLEASE NOTE --- THE CORE TEST IS ENTERED JUST PRIOR TO PRINTING
                              4279 *   THE DATA LIE, THIS IS DONE SO THAT THE FIELD UDC453 ONLY HAS
                              4280 *   A PSEUDO ADDRESS WHILE IT HAS TO POINT TO THE SAVED DATA ON DISK.
                              4281 *   AT ALL OTHER TINES UDC453 POINTS TO AN ACTUAL CORE ADDRESS.
                              4282 *
     13E8 C0 87 170A          4283        B     ZCTEST                    GO TO CORE TEST
     13EC C0 87 0DB9          4284        B     CVBHEX                    BYTES TO 2 HEX CHARACTERS
     13F0 20            13F0  4285        DC    XL1'20'                   LENGTH IN BYTES
     13F1               13F2  4286 UDC453 DS    CL2                       ADDR OF LEFT BYTE OF INPUT
     13F3 1538          13F4  4287        DC    AL2(UDC749+UDC137)        LEFT ADDR OUTPUT FLD
     13F5 C0 87 175D          4288        B     ZCTRST                    GO TO CORE TEST RESTORE
     13F9 8C 03 07 1631       4289        MVC   UDC120(UDC112,@XR),UDC820  FILL 4 BLANKS BETWEEN ADDR-HEX
                              4290 *
                              4291 *   THIS NEXT SECTION MOVES THE 64 BYTE FIELD WHICH HAS BEEN CONVERTED,
                              4292 *   FROM 32 BYTES OF CORE INTO THE PROPER PLACES IN THE PRINT FIELD
                              4293 *   WITH BLANKS BETWEEN 8 GROUPS OF 8 HEX CHARACTERS.
                              4294 *
     13FE 7C 07 14            4295        MVI   UDC455+@D1(,@BR),UDC093   INITIALIZE TO ADDR
     1401 E2 02 08            4296 UDC454 LA    UDC103(,@XR),@XR          UPDATE FIELDS POINTER.
     1404 AC 07 00 37         4297 UDC455 MVC   *-*(UDC103,@XR),UDC094(,@XR)  MOVE 8-BYTE FIELD
     1408 7C 02 1F            4298        MVI   UDC456+@D1(,@BR),UDCBN2   SET DISP FOR EDITING BLANKS
     140B 5E 00 1F 14         4299        ALC   UDC456+@D1(1,@BR),UDC455+@D1(,@BR)
     140F 8C 01 00 1631       4300 UDC456 MVC   *-*(@CADDR,@XR),UDC820    MOVE BLANKS
     1414 1E 00 14FB 1E       4301        ALC   UDC741(1),UDC456+@Q(,@BR)
     1419 4E 00 14 14FB       4302        ALC   UDC455+@D1(1,@BR),UDC741
     141E 3B FE 14FB          4303        SBF   UDC741,X'FE'
     1422 7D 12 14            4304        CLI   UDC455+@D1(,@BR),X'12'       *
     1425 D0 82 0F            4305        BL    UDC454(,@BR)              NO, GO BACK FOR NEXT GROUP
                              4306 *
                              4307 *   MOVE 6 BLANKS BEFORE CHARACTER FIELD
                              4308 *
     1428 8C 01 15 1631       4309        MVC   UDCB21(@CADDR,@XR),UDC820 EDIT BLANKS FOLLOWING DATA
     142D BC 5C 16            4310        MVI   UDCB22(,@XR),UDC060       MOVE '*' BEFORE INTERPRETATION
     1430 C0 87 170A          4311        B     ZCTEST                    CALL TEST
     1434 5C 01 51 00         4312        MVC   UDC458+@DOP2(,@BR),UDC453(@CADDR,@BR)  MOVE CHAR FLD ADDR
     1438 C0 87 175D          4313        B     ZCTRST                    CALL RESTORE
     143C 7A 1F 51            4314        SBN   UDC458+@DOP2(,@BR),X'1F'  POINT TO END OF FIELD
     143F 8C 1F 36 0000       4315 UDC458 MVC   UDCB54(UDCB32,@XR),*-*    MOVE INTERPRETATION
     1444 BC 5C 37            4316        MVI   UDCB55(,@XR),UDC060       MOVE '*' FOLLOWING INTERP
     1447 C0 87 14A9          4317        B     UCLCHG                    GO REMOVE NON-PRINTABLES
                         144B 4318 UDC459 EQU   *
                              4319 *
```

```
                                    4320 *   PRINT FIELD IS NOW COMPLETE, PRINT IT ON PAPER
                                    4321 *
                                    4322 *          PRNT  UDC746                      CALL PRINT LINE
     144B C0 87 0707                4323          B     $$PRNT                  PRINT ON MATRIX PRINTER
     144F 14FC              1450    4324          DC    AL2(UDC746)             PPL ADDRESS
                                    4325 *** END OF EXPANSION ***
                                    4326 *          PRNT  $WAITF                      CALL PRINT WAIT
     1451 C0 87 0707                4327          B     $$PRNT                  PRINT ON MATRIX PRINTER
     1455 057F              1456    4328          DC    AL2($WAITF)             PPL ADDRESS
                                    4329 *** END OF EXPANSION ***
     1457 C0 87 0DA5                4330          B     ZUTIRI                  *  GO TEST FOR INTERRUPTS
     145B C0 87 170A                4331          B     ZCTEST                  LINK TO TEST FOR SAVED CORE
     145F 5C 01 7F 00               4332          MVC   UDC460+@OP2(@CADDR,@BR),UDC453(,@BR)
     1463 C0 87 175D                4333          B     ZCTRST                  CALL RESTORE
     1467 4E 01 7F 1628             4334          ALC   UDC460+@OP2(@CADDR,@BR),UDC814   ADD 31 TO ADDR
     146C 0C 00 137B 0000           4335 UDC460 MVC    UDC402+@Q(1),*-*        MOVE LAST BYTE INTO WORK AREA
     1472 4E 01 00 162B             4336          ALC   UDC453(@CADDR,@BR),UDC818   INCR ADDR BY 32
     1477 C0 87 1366                4337          B     UDC400                  RETURN FOR NEW LINE
                                    4338 *UDC461 PRNT  UDC754
     147B C0 87 0707                4339 UDC461 B      $$PRNT                  PRINT ON MATRIX PRINTER
     147F 1578              1480    4340          DC    AL2(UDC754)             PPL ADDRESS
                                    4341 *** END OF EXPANSION ***

     1481 C0 87 0DA5                4343          B     ZUTIRI                  * GO TEST FOR INTERRUPTS
                                    4344 *        PRNT  UDC880                   PRINT SECOND HEAD LINE
     1485 C0 87 0707                4345          B     $$PRNT                  PRINT ON MATRIX PRINTER
     1489 1646              148A    4346          DC    AL2(UDC880)             PPL ADDRESS
                                    4347 *** END OF EXPANSION ***

     148B C0 87 0DA5                4349          B     ZUTIRI                  * GO TEST FOR INTERRUPTS
                                    4350 *         PRNT  UDC886                  SPACE
     148F C0 87 0707                4351          B     $$PRNT                  PRINT ON MATRIX PRINTER
     1493 16C2              1494    4352          DC    AL2(UDC886)             PPL ADDRESS
                                    4353 *** END OF EXPANSION ***

     1495 C0 87 0DA5                4355          B     ZUTIRI                  * GO TEST FOR INTERRUPTS
     1499 C0 87 0000                4356 UDC470 B      *-*                     BRANCH TO PROCESS OR CHECK LINE
                                    4357 *
                                    4358 *   RESTORE REGISTERS AND RETURN
                                    4359 *
     149D C2 02 0000                4360 UDC497 LA     *-*,@XR                 RETRUN XR2 TO CALLING PROGRAM
     14A1 C2 01 0000                4361 UDC498 LA     *-*,@BR                 RETURN CALLING PROG BASE REG.
     14A5 C0 87 0000                4362 UDC499 B      *-*                     RETURN - CALLING PROGRAM

                                    4364 *********************************************************************
                                    4365 *                                                                   *
                                    4366 *   INSERT PERIODS IN INTERPRETED FIELD FOR NON-PRINTABLE CHARACTERS *
                                    4367 *                                                                   *
                                    4368 *********************************************************************
                              14A9  4369 UCLCHG EQU    *                       ROUTINE TO SET IN PERIODS
     14A9 3C 1F 14C8                4370          MVI   UDC551+@DOP2,UDC090     SET DISPLACEMENT IN MOVE
     14AD 3C 1F 14DB                4371          MVI   UDC553+@D1,UDC090       SET DISPLACEMENT IN MOVE
     14B1 34 02 14EF                4372          ST    UDC560+@OP1,@XR         SAVE XR
     14B5 34 01 14F3                4373          ST    UDC565+@OP1,@BR         SAVE BR
     14B9 E2 01 17                  4374          LA    UDC570(,@XR),@BR        SET BR TO START OF FIELD
     14BC C2 02 16CA                4375          LA    UDC902,@XR              SET XR TO TABLE START ADD
```

```
      UDUMPC - CORE-DISK DUMPS


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT              VER 15, MOD 00  13/01/22  PAGE  39

    14C0 3C 3F 14CB           4376 UDC540 MVI    UDC550+@D1,UDC571        SET DISPL FOR TABLE IN COMP
    14C4 1C 00 14CA 00        4377 UDC551 MVC    UDC550+@Q,*-*(1,@BR)     MOVE CHAR INTO COMP INSTR
    14C9 BD 00 00             4378 UDC550 CLI    *-*(,@XR),*-*            COMPARE FOR PRINTABLE CHAR
    14CC F2 81 0D             4379        JE     UDC555                   SKIP PERIOD INSERTION ON
    14CF 0F 00 14CB 1629      4380        SLC    UDC550+@D1,UDC817(1)     MOVE TABLE POINT BY ONE
    14D5 C0 02 14C9           4381        BNM    UDC550                   GO TEST FOR NEXT TABLE ENTRY
    14D9 7C 4B 00             4382 UDC553 MVI    *-*(,@BR),C'.'           PLACE IN PERIOD FOR NON-PRINT
    14DC 0F 00 14DB 1629      4383 UDC555 SLC    UDC553+@D1(1),UDC817     REDUCE DISP BY ONE
    14E2 0F 00 14C8 1629      4384        SLC    UDC551+@DOP2(1),UDC817   REDUCE FIELD BY ONE
    14E8 C0 02 14C0           4385        BNM    UDC540                   GO CHECK NEXT CHARACTER
    14EC C2 02 0000           4386 UDC560 LA     *-*,@XR                  RESTORE XR
    14F0 C2 01 0000           4387 UDC565 LA     *-*,@BR                  RESTORE BR
    14F4 C0 87 144B           4388        B      UDC459                   RETURN TO CALLER
                              4389 *
                              4390 * WORK AREA UDUMPC
                              4391 *
                         0078 4392 UDCPLL EQU    120                      PRINT LINE LENGTH
    14F8                 14F9 4393 UDC720 DS     CL(@CADDR)               ENDING ADDR OF CORE DUMP
    14FA                 14FA 4394 UDC740 DS     CL1                      INDICATOR IF LAST LINE PRINTED
    14FB 00              14FB 4395 UDC741 DC     XL1'0'                   DISPLACEMENT ADDER
                              4396 *UDC746 PPL   FUNC=@PRETR,CNT=UDC747,CADDR=UDC749
                         14FC 4397 UDC746 EQU    *                        PPL ADDRESS
    14FC C0              14FC 4398        DC     AL1(@PRETR)              FUNCTION REQUESTED
    14FD 78              14FD 4399        DC     AL1(UDC747)              PRINT COUNT
    14FE 1500            14FF 4400        DC     AL2(UDC749)              DATA ADDRESS
                              4401 *** END OF EXPANSION ***

                         0078 4403 UDC747 EQU    120
                         1500 4404 UDC749 EQU    *                        LEFT MOST BYTE OF OUTPUT LINE *
    1500                 1577 4405 UDC750 DS     CL(UDCPLL)               OUTPUT FIELD FOR PRINTING
                              4406 *UDC754 PPL   FUNC=@PRETR,CNT=UDC753,CADDR=UDC751
                         1578 4407 UDC754 EQU    *                        PPL ADDRESS
    1578 C0              1578 4408        DC     AL1(@PRETR)              FUNCTION REQUESTED
    1579 59              1579 4409        DC     AL1(UDC753)              PRINT COUNT
    157A 157C            157B 4410        DC     AL2(UDC751)              DATA ADDRESS
                              4411 *** END OF EXPANSION ***
                         157C 4412 UDC751 EQU    *
    157C                 15F3 4413 UDC752 DS     CL(UDCPLL)               SET REG PAGE HEADER HERE
    157C                      4414        ORG    *-UDCPLL
    157C 4040404040404040 15F3 4415        DC     CL(UDCPLL)' '           INITIALIZE DATA AREA

                              4417 ***********************************************************************
                              4418 * CORE/DISK DUMP ROUTINE CONSTANTS AND EQUATES
                              4419 ***********************************************************************
                              4420 *
                              4421 * EQUATES
                              4422 *
                         0002 4423 UDCBN2 EQU    2                        BINARY 2
                         0015 4424 UDCB21 EQU    21                       21
                         0016 4425 UDCB22 EQU    22                       22
                         0036 4426 UDCB54 EQU    54                       54
                         0020 4427 UDCB32 EQU    32                       32
                         0037 4428 UDCB55 EQU    55                       55
                         0032 4429 UDCB50 EQU    50                       50
                         0014 4430 UDCB20 EQU    20                       20
                         0059 4431 UDC753 EQU    89                       89
```

```
                             0017 4432 UDC570 EQU    23                          23
                             003F 4433 UDC571 EQU    63                          PRINTABLE TABLE SIZE
                             001C 4434 UDCB28 EQU    28                          28
                             0018 4435 UDCB24 EQU    24                          24
                                  4436 *
                                  4437 * CONSTANTS
                                  4438 *
    15F4                     1626 4439 UDC810 DS     CL(UDCIML)
    15F4                          4440        ORG    *-UDCIML
    15F4 D5C5E7E340D3C9D5 160A 4441        DC     CL(UDC570)'NEXT LINE/S IDENTICAL T'
    160B D640D3C1E2E340C2 1626 4442        DC     CL(UDCB28)'O LAST BYTE OF PREVIOUS LINE'
    1627 001F              1628 4443 UDC814 DC     XL(@CADDR)'001F'            31
    1629 01                1629 4444 UDC817 DC     XL1'01'                     BINARY 1
    162A 0020              162B 4445 UDC818 DC     XL(@CADDR)'0020'            BINARY 32 = BYTE INCREMENT/LINE
    162C 404040404040      1631 4446 UDC820 DC     CL(UDC101)'        '        CONSTANT FOR SPACING IN OUTPUT
    1632 4040404040C2D97E  1639 4447 UDC830 DC     CL8'      BR='
    163A 404040E7D97E      163F 4448 UDC840 DC     CL6'   XR='
    1640 4040D7E2D97E      1645 4449 UDC850 DC     CL6'  PSR='                 PRINT FLD CON PSR
                                  4450 *UDC880 PPL    FUNC=@PRETR,CNT=UDCPLL,CADDR=UDC884
                             1646 4451 UDC880 EQU    *                          PPL ADDRESS
    1646 C0                1646 4452        DC     AL1(@PRETR)                FUNCTION REQUESTED
    1647 78                1647 4453        DC     AL1(UDCPLL)                PRINT COUNT
    1648 164A              1649 4454        DC     AL2(UDC884)                DATA ADDRESS
                                  4455 *** END OF EXPANSION ***


                             164A 4457 UDC884 EQU    *
    164A C1C4C4D94040404E  1669 4458        DC     CL(UDCB32)'ADDR    +00 1 2 3   4 5 6 7    8 9 '
    166A C140C24040C340C4  1689 4459        DC     CL(UDCB32)'A B   C D E F +10 1 2 3   4 5 6 7 '
    168A 4040F840F940C140  16A9 4460        DC     CL(UDCB32)'  8 9 A B   C D E F     ******** '
    16AA C9D5E3C5D9D7D9C5  16C1 4461 UDC885 DC     CL(UDCB24)'INTERPRETATION  ********'
                                  4462 *UDC886 PPL    FUNC=@RETRN,CNT=@RTRNC
                             16C2 4463 UDC886 EQU    *                          PPL ADDRESS
    16C2 80                16C2 4464        DC     AL1(@RETRN)                FUNCTION REQUESTED
    16C3 80                16C3 4465        DC     AL1(@RTRNC)                PRINT COUNT
    16C4 0000              16C5 4466        DC     AL2(*-*)                   DATA ADDRESS
                                  4467 *** END OF EXPANSION ***


    16C6 13D2              16C7 4469 UDC900 DC     AL2(UDC451)                *
    16C8 1366              16C9 4470 UDC901 DC     AL2(UDC400)                *
                             16CA 4471 UDC902 EQU    *                          *
    16CA 7C6C5C4C7B6B5B4B  16E9 4472        DC     CL(UDCB32)'@%*<#,$.-/+ ) 1&'' ;:>?"^ (VWXYZ'
    16EA C1C2C3C4C5C6C7C8  1709 4473 ZUTTAB DC     CL(UDCB32)'ABCDEFGHIJKLMNOPQRSTU 0123456789'
                                  4474 ****************************************************************************
```

```
                              4476 *********************************************************************
                              4477 *  CORE TEST ROUTINE * PROVIDES INTERFACE TO ZCSAVE FROM CORE DUMP.   *
                              4478 *  IF FIRST TIME SWITCH IS SET, CALL TO ZCSAVE IS FORCED ON THE      *
                              4479 *   SUBSEQUENT CALL...  ELSE...  ZCSAVE IS CALLED ON HEX 100          *
                              4480 *  STORAGE BOUNDARIES ONLY...                                         *
                              4481 *********************************************************************
                              4482 *ZCTEST ENTER EXIT=ZCTX1,@BR,@XR,@ARR
                         170A 4483 ZCTEST EQU    *                          MODULE ENTRY POINT
     170A 34 01 1754          4484        ST     ZCTX10+@OP1,@BR            SAVE @BR
     170E 34 02 1758          4485        ST     ZOTX11+@OP1,@XR            SAVE @XR
     1712 34 08 175C          4486        ST     ZCTX12+@OP1,@ARR           SAVE RETURN ADDRESS
                              4487 *** END OF EXPANSION ***

     1716 3D FF 120E          4489        CLI    ZCDDSK,ZCDFFF              TEST FOR DISK DUMP
     171A F2 81 2E            4490        JE     ZCTDSK                     *
     171D C0 80 1728          4491 ZCT010 BC     ZCT020,@NOP               FIRST TIME SWITCH
     1721 39 FF 13F2          4492        TBF    UDC453,ZCTFFF             TEST FOR HEX 100 BOUNDARY
     1725 F2 90 0E            4493        JF     ZCT030                    SKIP CALL ON ODD BOUNDARY
     1728 0C 01 12C3 13F2     4494 ZCT020 MVC    ZCSADD,UDC453(@CADDR)     LOAD DUMP ADDRESS
     172E 3C 80 171E          4495        MVI    ZCTFTS,@NOP               RESET FIRST TIME SWITCH
     1732 C0 87 1223          4496        B      ZCSAVE                    CALL ZCSAVE

     1736 38 FF 12C4          4498 ZCT030 TBN    ZCSREL,ZCTFFF             TEST FOR REAL CORE
     173A 0C 00 177A 13F1     4499        MVC    ZCTSVA,UDC453-1(1)        SAVE REAL ADDRESS FOR LATER
     1740 F2 10 0E            4500        JT     ZCTX10                    RETURN TO CALLING PROGRAM
     1743 38 F0 12C4          4501        TBN    ZCSREL,ZCTF00             TEST FOR EXCEEDED CORE
     1747 C0 10 12DC          4502        BT     UDC210                    SHUT DOWN DUMP
     174B 0C 00 13F1 177B     4503 ZCTDSK MVC    UDC453-1(1),ZCTADD-1      SET IN DISK I/O ADDRESS
                              4504 *ZCTX1  EXIT  @BR,@XR,RETURN
     1751 C2 01 0000          4505 ZCTX10 LA     *-*,@BR                   RESTORE @BR
     1755 C2 02 0000          4506 ZOTX11 LA     *-*,@XR                   RESTORE @XR
     1759 C0 87 0000          4507 ZCTX12 B      *-*                       RETURN TO CALLING PROGRAM
                              4508 *** END OF EXPANSION ***

                              4510 *
                              4511 * ROUTINE TO RESTORE THE REAL ADDRESS AS LOW ADDRESS TO BE DUMPED
                              4512 *
                              4513 *ZCTRST ENTER EXIT-ZCTX2,,,@ARR
                         175D 4514 ZCTRST EQU    *                          MODULE ENTRY POINT
     175D 34 08 1771          4515        ST     ZCTX22+@OP1,@ARR           SAVE RETURN ADDRESS
                              4516 *** END OF EXPANSION **
     1761 3D FF 120E          4517        CLI    ZCDDSK,ZCTFFF             IS THIS DISK DUMP ?
     1765 F2 81 0A            4518        JE     ZCTRDK
     1768 0C 00 13F1 177A     4519        MVC    UDC453-1(1),ZCTSVA        RESTORE CORRECT ADDRESS
                              4520 *ICTX2  EXIT  ,,RETURN
     176E C0 87 0000          4521 ZCTX22 B      *-*                       RETURN TO CALLING PROGRAM
                              4522 *** END OF EXPANSION ***

     1772 3C 00 13F1          4524 ZCTRDK MVI    UDC453-1,0                ZERO OUT HIGH ORDER
     1776 C0 87 176E          4525        B      ZCTX22
                              4526 *  END OF ROUTINE
                    00FF      4527 ZCTFFF EQU    X'FF'
                    00F0      4528 ZCTF00 EQU    X'F0'
                    171E      4529 ZCTFTS EQU    ZCT010+@Q                 POINTER TO 1ST TINE SWITCH
     177A              177A   4530 ZCTSVA DS     CL1                       *
     177B 1F00         177C   4531 ZCTADD DC     XL2'1F00'
```

```
                           4532 *   END OF STORAGE FOR THIS ROUTINE
                           4533 *
                           4534 ***********************************************************************
```

```
                          4536 ****************************************************************************
                          4537 *                         PACK ROUTINE                                     *
                          4538 *                                                                          *
                          4539 *  SUBROUTINE TO CONVERT EBCDIC DIGITS 0-F TO PACKED HEXADECIMAL DATA.*
                          4540 *     LINKAGE TO THIS SUBROUTINE IS AS FOLLOWS:                          *
                          4541 *           B    UCL900                      ENTRY POINT                   *
                          4542 *           DS   1,LENGTH OF SOURCE FIELD                                  *
                          4543 *           DS   2,SOURCE ADDR (LEFT MOST BYTE OF SOURCE FIELD)            *
                          4544 *                                                                          *
                          4545 *  THE RESULTING PACKED HEXADECIMAL DATA IS PLACED INTO THE LEFTMOST  *
                          4546 *     POSITIONS OF THE SOURCE FIELD.  THE LENGTH OF THE PACKED DATA     *
                          4547 *     REPLACES THE LENGTH SPECIFIED FOR THE SOURCE FIELD.               *
                          4548 *                                                                          *
                          4549 *  PACK TERMINATES ON X'40' IN THE SOURCE FIELD OR WHEN THE LENGTH      *
                          4550 *     SPECIFICATION IS MET, WHICHEVER OCCURS FIRST.                      *
                          4551 ****************************************************************************
                          4552 *
                          4553 *                         INITIALIZATION ROUTINE
                          4554 *
                     177D 4555          USING UCL900,@BR                  ESTABLISH LOCAL BASE
     177D 34 01 183C      4556 UCL900 ST    UCL970,@BR                   SAVE CONTENTS OF BASE REGISTER
     1781 C2 01 177D      4557        LA    UCL900,@BR                   LOAD LOCAL BASE
     1785 7B FF C4        4558        SBF   UCL974(,@BR),UCL954          TURN OFF NO-HEX SWITCH
     1788 74 02 C1        4559        ST    UCL971(,@BR),@XR             SAVE CONTENTS OF INDEX REGISTER
     178B 74 08 15        4560        ST    UCL905+@DOP2(,@BR),@ARR      LENGTH PARAM ADDR TO MOVE INST
     178E 4C 00 C2 0000   4561 UCL905 MVC   UCL972(1,@BR),*-*            MOVE LENGTH TO SOURCE COUNTER
     1793 74 08 AA        4562        ST    UCL949+@OP1(,@BR),@ARR       SAVE LNG ADDR FOR PACKED LNG
     1796 76 08 BA        4563        A     UCL962(,@BR),@ARR            ADJUST @ARR FOR SOURCE ADDR
     1799 74 08 23        4564        ST    UCL907+@DOP2(,@BR),@ARR      SET FOR RET. OF PACKED DATA ZONE
     179C 4C 01 5F 0000   4565 UCL907 MVC   UCL930+@OP1(@CADDR,@BR),*-*  *
     17A1 74 08 2B        4566        ST    UCL908+@DOP2(,@BR),@ARR      SET FOR RET. OF PACKED DATA NUM.
     17A4 4C 01 8D 0000   4567 UCL908 MVC   UCL935+@OP1(@CADDR,@BR),*-*  *
     17A9 75 02 5F        4568        L     UCL930+@OP1(,@BR),@XR        PUT SOURCE ADDR IN INDEX REG.
     17AC 5F 00 C3 C3     4569        SLC   UCL973(1,@BR),UCL973(,@BR)   CLEAR PACKED DATA LNG. CTR.
     17B0 76 08 B8        4570        A     UCL961(,@BR),@ARR            ADJUST @ARR FOR RET. ADDR.
     17B3 74 08 B6        4571        ST    UCL950+@OP1(,@BR),@ARR       PUT RET. ADDR. IN EXIT INST.
                          4572 *
                          4573 *                         PROCESSING LOOP
                          4574 *
     17B6 7D 40 00        4575 UCL910 CLI   UCL951(,@BR),UCL952          IS SOURCE BYTE A BLANK ?
     17B9 D0 81 A4        4576        BE    UCL945(,@BR)                 YES, EXIT
     17BC 5E 00 C3 BD     4577        ALC   UCL973(1,@BR),UCL964(,@BR)   ADD 1 TO PACKED COUNTER
     17C0 B8 F0 00        4578        TBN   UCL951(,@XR),UCL953          IS SOURCE BYTE EBCDIC 0-9 ?
     17C3 D0 10 5C        4579        BT    UCL930(,@BR)                 YES, SKIP CORRECTION
     17C6 BD C1 00        4580        CLI   UCL951(,@XR),C'A'            TEST FOR AN A
     17C9 F2 82 06        4581        JL    UCL915                       GO SET NO-HEX SWITCH
     17CC BD C6 00        4582        CLI   UCL951(,@XR),C'F'            TEST FOR AN F
     17CF F2 04 03        4583        JNH   UCL920                       JUMP SWITCH TURN ON IF VALID
     17D2 7A FF C4        4584 UCL915 SBN   UCL974(,@BR),UCL954          TURN ON NO-HEX SWITCH
                     17D5 4585 UCL920 EQU   *
     17D5 9E 00 00 BB     4586        ALC   UCL951(1,@XR),UCL963(,@BR)   CORRECT NUM. BITS BY ADDING 9
     17D9 28 01 0000 00   4587 UCL930 MZN   *-*,UCL951(,@XR)             MOVE NUM. TO OUTPUT FIELD-ZONE
     17DE 5F 00 C2 BD     4588        SLC   UCL972(1,@BR),UCL964(,@BR)   SUB 1 FROM SOURCE CTR
     17E2 D0 81 A4        4589        BZ    UCL945(,@BR)                 IF ZERO. EXIT
     17E5 76 02 BD        4590        A     UCL964(,@BR),@XR             POINT TO NEXT SOURCE BYTE
     17E8 BD 40 00        4591        CLI   UCL951(,@XR),UCL952          IS SOURCE BYTE A BLANK ?
```

```
    17EB D0 81 A4              4592          BE    UCL945(,@BR)              YES, EXIT
    17EE B8 F0 00             4593          TBN   UCL951(,@XR),UCL953       IS SOURCE BYTE EBCDIC 0-9 ?
    17F1 D0 10 8A             4594          BT    UCL935(,@BR)              YES, SKIP CORRECTION
    17F4 BD C1 00             4595          CLI   UCL951(,@XR),C'A'         TEST FOR AN A
    17F7 F2 82 06             4596          JL    UCL932                    GO SET NO-HEX SNITCH
    17FA BD C6 00             4597          CLI   UCL951(,@XR),C'F'         TEST FOR AN F
    17FD F2 04 03             4598          JNH   UCL933                    JUMP SWITCH TURN ON IF VALID
    1800 7A FF C4             4599 UCL932 SBN    UCL974(,@BR),UCL954       TURN ON NO-HEX SWITCH
    1803            4600 UCL933 EQU   *
    1803 9E 00 00 BB          4601          ALC   UCL951(1,@XR),UCL963(,@BR)  CORRECT NUM. BITS BY ADDING 9
    1807 28 03 0000 00        4602 UCL935 MNN    *-*,UCL951(,@XR)          MOVE NUM. TO OUTPUT FIELD - NUM.
    180C 5F 00 C2 BD          4603          SLC   UCL972(1,@BR),UCL964(,@BR)  SUB 1 FROM SOURCE COUNTER
    1810 D0 81 A4             4604          BZ    UCL945(,@BR)              IF ZERO, EXIT.
    1813 5E 01 5F BD          4605          ALC   UCL930+@OP1(@CADDR,@BR),UCL964(,@BR)  ADJ RET. PACKED ZONE
    1817 5E 01 8D BD          4606          ALC   UCL935+@OP1(@CADDR,@BR),UCL964(,@BR)  ADJ RET. PACKED NUM.
    181B 76 02 BD             4607          A     UCL964(,@BR),@XR          POINT TO NEXT SOURCE BYTE
    181E D0 87 39             4608          B     UCL910(,@BR)              CONTINUE PROCESSING LOOP.
                    4609 *
                    4610 *                     DONE - RESTORE REGISTERS AND LENGTH - EXIT
                    4611 *
    1821 75 02 C1             4612 UCL945 L      UCL971(,@BR),@XR          RESTORE INDEX REG.
    1824 1C 00 0000 C3        4613 UCL949 MVC    *-*,UCL973(1,@BR)         RETURN LNG. OF PACKED DATA
    1829 75 01 BF             4614          L     UCL970(,@BR),@BR          RESTORE BASE REG.
    182C 3D 00 1841           4615          CLI   UCL974,UCL951            SET TO EQUAL IF OK
    1830 C0 87 0000           4616 UCL950 B      *-*                       EXIT
                    4617 *
                    4618 *                     CONSTANTS, STORAGE AREAS AND EQUATES
                    4619 *
    0000 4620 UCL951 EQU   0                         ZERO DISP. - @XR USED AS POINTER
    0040 4621 UCL952 EQU   X'40'                     BLANK - FOR CHECKING SOURCE FLD
    00F0 4622 UCL953 EQU   X'F0'                     CHECK CHAR FOR EBCDIC 0-9
    00FF 4623 UCL954 EQU   X'FF'
    1834 0001       1835 4624 UCL961 DC    XL2'01'                   ADJMT FACTOR TO GET RET. ADDR.
    1836 0002       1837 4625 UCL962 DC    XL2'02'                   ADJMT FACTOR TO GET SOURCE ADDR
    1838 09         1838 4626 UCL963 DC    XL1'09'                   CORRECTION FACTOR - EBCDIC A-F
    1839 0001       183A 4627 UCL964 DC    XL2'01'                   CTR & POSITION ADJUSTMENT FACTR
    183B            183C 4628 UCL970 DS    CL2                       SAVE BASE REGISTER
    183D            183E 4629 UCL971 DS    CL2                       SAVE INDEX REGISTER
    183F            183F 4630 UCL972 DS    CL1                       SOURCE FIELD LENGTH COUNTER
    1840            1840 4631 UCL973 DS    CL1                       PACKED FIELD LENGTH COUNTER
    1841 00         1841 4632 UCL974 DC    XL1'00'
                    4633 *
                    4634 ***********************************************************************
                    4635 ***********************************************************************
                    4636 * UPATCH IS CALLED BY THE UTILITY MONITOR AND IT PATCHES EITHER CORE  *
                    4637 * OR DISK ONE HEX CHARACTER AT A TIME.  ALL INPUT IS FROM THE         *
                    4638 * KEYBOARD, AND ALL OUTPUT IS TO THE MATRIX PRINTER                   *
                    4639 ***********************************************************************
                    4640 *
                    4641 *         UPATCH EQUATES
                    4642 *
    0009 4643 UPGOTC EQU   X'09'                     RESTORE AND LOCK KEYBOARD CNTR
    0001 4644 UPT001 EQU   1                         CONSTANT
    0001 4645 UPQNZ  EQU   X'01'                     NUMERIC TO ZONE
    0002 4646 UPQZN  EQU   X'02'                     ZONE TO NUMERIC
    0003 4647 UPQNN  EQU   X'03'                     NUMERIC TO NUMERIC
```

```
                        0011 4648 UPREAD EQU   X'11'                  SIO Q CODE FOR KEYBOARD
                        0007 4649 UPKEY  EQU   X'07'                  READY TO READ CONTROL
                        0040 4650 UPBLNK EQU   X'40'                  BLANK CHARACTER
                        004E 4651 UPRTNC EQU   X'4E'                  PROG START KEY IS RETURN KEY
```

UDUMPC - CORE-DISK DUMPS

```
                      0011 4653 UPSNSQ EQU   X'11'                   SNS 0 CODE FOR KEYBOARD
                           4654 *
                      1842 4655 CPATCH EQU   *                       START HERE FOR CORE PATCH
    1842 0C 01 19B8 1A39   4656        MVC   ZCP020+@OP1(@CADDR),ZCPZCS  SET IN BRANCH FOR CORE TEST
    1848 C0 87 1860        4657        B     UPATCH                  START PATCH
                      184C 4658 DPATCH EQU   *                       DISK PATCH START
    184C 0C 00 1174 1173   4659        MVC   ZCD100(1),ZCD100-1      SET DISPLACEMENT TO LOW ORDER
    1852 3C 04 1159        4660        MVI   ZCD062+@Q,ZCDFOR        CHANGE LENGTH BACK
    1856 0C 01 19B8 1A39   4661        MVC   ZCP020+@OP1(@CADDR),ZCPZCS  SET IN BRANCH FOR DISK TEST
    185C 3C 87 1A04        4662        MVI   ZCPTST+@Q,@UCB          SET ON FIRST TIME SWITCH
                           4663 *UPATCH ENTER BASE=UPGDTA
                      1874 4664        USING UPGDTA,@BR              BASE ADDRESS SPECIFICATION
                      1860 4665 UPATCH EQU   *                       MODULE ENTRY POINT
    1860 C2 01 1874        4666        LA    UPGDTA,@BR              LOAD BASE REGISTER
                           4667 *** END OF EXPANSION ***
    1864 3C 87 19A7        4668        MVI   ZCPFTS,@UCB             TURN ON FIRST TIME SWITCH
                           4669 *
                           4670 * SET UP PATCH ADDRESS AND BEGIN PATCH OPERATION
    1868 0C 01 1948 1174   4671        MVC   UPADDR,ZCD100(@CADDR)   MOVE IN CORE PATCH ADDR
                           4672 *
                           4673 *     PRNT  UPTMO2                   ASK FOR DATA
    186E C0 87 0707        4674        B     $$PRNT                  PRINT ON MATRIX PRINTER
    1872 198D         1873 4675        DC    AL2(UPTM02)             PPL ADDRESS
                           4676 *** END OF EXPANSION ***
                           4677 * UPGDTA - READ HEX PATCH CHARACTERS FROM KEYBOARD.
                           4678 *
                           4679 * INITIALIZE COUNTERS, MVX Q CODES, INITIATE READ
                           4680 *
    1874 5F 01 D8 D8       4681 UPGDTA SLC   UPTCNT(,@BR),UPTCNT(2,@BR)  ZERO COUNTER
    1878 C0 87 0890        4682        B     $$PRES                  GET KEYBOARD LINE
    187C 38 10 03C3        4683 UPCHCK TBN   $KEYCD,$KYBSY           WAIT FOR COMPLETION
    1880 C0 10 187C        4684        BT    UPCHCK                  WAIT...
    1884 C0 87 0DA5        4685        B     ZUTIRI                  * GO TEST FOR INTERRUPTS
                           4686 ***********************************************************************
                           4687 *     THIS ROUTINE CHECKS FOR VALID PATCH DATA
                           4688 *     IF INVALID A QUESTION MARK WILL PRINT AND DATA MUST
                           4689 *     BE ENTERED ALL OVER AGAIN
                           4690 ***********************************************************************
    1888 C2 02 0607        4691        LA    $$INLN,@XR              SET XR TO LINE BUFFER VALUE
    188C BD 1E 00          4692 UPTLP0 CLI   0(,@XR),@EOS            CHECK FOR END OF LINE
    188F C0 81 18B9        4693        BE    UPTCON                  GO CONTINUE THE OPERATION
    1893 BD F9 00          4694        CLI   0(,@XR),C'9'            CHECK FOR 9
    1896 D0 84 E2          4695        BH    UPBADC(,@BR)            GO TO ERROR IF > 9
    1899 BD 40 00          4696        CLI   0(,@XR),C' '            TEST FOR A BLANK
    189C C0 81 18B3        4697        BE    UPTLP1                  GO CHECK NEXT CHARACTER
    18A0 BD F0 00          4698        CLI   0(,@XR),C'0'            TEST FOR ZERO
    18A3 C0 02 18B3        4699        BNL   UPTLP1                  CHECK NEXT CHARACTER IF NUMERIC
    18A7 BD C1 00          4700        CLI   0(,@XR),C'A'            TEST FOR AN A
    18AA D0 82 E2          4701        BL    UPBADC(,@BR)            CHARACTER ILLEGAL IF < A
    18AD BD C6 00          4702        CLI   0(,@XR),C'F'            TEST FOR AN F
    18B0 D0 84 E2          4703        BH    UPBADC(,@BR)            IN ERROR IF MORE THAN F
    18B3 E2 02 01          4704 UPTLP1 LA    UPT001(,@XR),@XR        ADD ONE TO THE XR
    18B6 D0 87 18          4705        B     UPTLP0(,@BR)            GO CHECK NEXT CHARACTER
    18B9 75 02 D4          4706 UPTCON L     UPADDR(,@BR),@XR        SET XR TO PATCH ADDRESS
    18BC 5C 01 ED EF       4707        MVC   UPBUF(,@BR),UPADRR(@CADDR,@BR)  SET UP LINE ADDRESS
    18C0 F2 87 10          4708        J     UPT200                  DONT UPDATE ADDRESS INITIALLY
```

```
      UDUMPC - CORE-DISK DUMPS


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT           VER 15, MOD 00  13/01/22  PAGE  47

    18C3 E2 02 01          4710 UPEVEN LA    UPT001(,@XR),@XR          UPDATE PATCH ADDR
    18C6 74 02 75          4711        ST    UPT210+@OP1(,@BR),@XR     SAVE XR FOR CARRY TEST
    18C9 79 FF 75          4712        TBF   UPT210+@OP1(,@BR),ZCPFFF  TEST FOR CARRY OF SECTOR
    18CC F2 90 04          4713        JF    UPT200                   SKIP SWITH TURN OFF
    18CF 3C 80 1A04        4714        MVI   ZCPTST+@Q,@NOP           CLEAR SWITCH TO CAUSE SECTOR CH
    18D3 7C 01 91          4715 UPT200 MVI   UPT220+@Q(,@BR),UPQNZ    PATCH CHAR IS ZONE
    18D6             4716 UPLOOP EQU   *
    18D6 74 02 75          4717        ST    UPT210+@OP1(,@BR),@XR     SAVE @XR
    18D9 75 02 ED          4718        L     UPBUF(,@BR),@XR          CURRENT ADDRESS IN BUFFER
    18DC E2 02 01          4719        LA    UPT001(,@XR),@XR         INCREMENT TO NEXT POSTION
    18DF 6C 00 D5 00       4720        MVC   UPDATA(1,@BR),0(,@XR)    MOVE CURRENT BYTE TO TEST CELL
    18E3 74 02 ED          4721        ST    UPBUF(,@BR),@XR          STORE CURRENT ADDRESS
    18E6 C2 02 0000        4722 UPT210 LA    *-*,@XR                  RESTORE @XR
                           4723 *
                           4724 * DECODE KEYED CHARACTER, PATCH CORE
                           4725 *
    18EA 7D 1E D5          4726        CLI   UPDATA(,@BR),@EOS        IS CHAR A RETURN ?
    18ED D0 81 B4          4727        BE    UPRETN(,@BR)
    18F0 7D 40 D5          4728        CLI   UPDATA(,@BR),UPBLNK      IS CHAR A BLANK ?
    18F3 D0 81 9C          4729        BE    UPT225(,@BR)             GO AVOID DATA MOVE
    18F6 7D F0 D5          4730        CLI   UPDATA(,@BR),X'F0'       NUMERIC ?
    18F9 D0 02 8C          4731        BNL   UPT215(,@BR)             JUMP CORRECTION IF NUMERIC
    18FC 5E 00 D5 E1       4732        ALC   UPDATA(,@BR),UPTD9(1,@BR) ADD 9 TO NUMERIC BITS OF A-F
    1900 C0 87 199A        4733 UPT215 B     ZCPEST                   CALL CORE TEST
    1904 98 00 00 D5       4734 UPT220 MVX   0(,@XR),UPDATA(*-*,@BR)  MOVE PATCH DATA TO CORE
    1908 C0 87 19E7        4735        B     ZCPRST                   CALL RESTORE
    190C C0 87 1918        4736        B     UPT230                   GO AVOID TEST WITHOUT MOVE
    1910 C0 87 199A        4737 UPT225 B     ZCPEST                   CALL TEST
    1914 C0 87 19E7        4738        B     ZCPRST                   CALL RESTORE
                           4739 *
    1918 5E 01 D8 DC       4740 UPT230 ALC   UPTCNT(,@BR),UPTD1(2,@BR) UPDATE COUNT
    191C 78 01 D8          4741        TBN   UPTCNT(,@BR),X'01'       IS COUNT ODD ?
    191F D0 90 4F          4742        BF    UPEVEN(,@BR)             GO UPDATE BYTE COUNT
    1922 7C 03 91          4743        MVI   UPT220+@Q(,@BR),UPQNN    PATCH CHAR IS NUMERIC
    1925 D0 87 62          4744        B     UPLOOP(,@BR)             CONTINUE WITH PATCH OPERATION
                           4745 *
                           4746 *      RETURN CARRIAGE AND RETURN FROM GETDATA
                           4747 *
                           4748 *UPRETN PRNT  UPTM04
    1928 C0 87 0707        4749 UPRETN B     $$PRNT                   PRINT ON MATRIX PRINTER
    192C 1996         192D 4750        DC    AL2(UPTM04)              PPL ADDRESS
                           4751 *** END OF EXPANSION **
    192E 38 FF 12C4        4752        TBN   ZCSREL,ZCPFFF            TEST IF CORE IS SAVED
    1932 F2 10 0C          4753        JT    UPT250                   SKIP CORE SAVE WRITE OP
                           4754 *      DSKL2 ICPDPL,WAIT              WRITE PATCH
    1935 C0 87 0FAC        4755        B     DL2ICS                   PERFORM RELATIVE DISK OP
    1939 1A2D         193A 4756        DC    AL2(ZCPDPL)              DPL ADDRESS
    193B C0 87 0025        4757        B     $DISKN                   WAIT AND CHECK DISK ERRORS
    193F 057F         1940 4758        DC    AL2($WAITF)             WAIT DPI ADDRESS
                           4759 *** END OF EXPANSION ***
    1941 C0 87 0C39        4760 UPT250 B     ZUT020                   RETURN TO MONITOR
                           4761 *
    0029             4762 UPTL02 EQU   41                       LENGTH OF MESSAGE
    0001             4763 UPTL03 EQU   X'01'                    LENGTH OF MESSAGE
                           4764 *
                           4765 *   UPATCH WORK AREA
```

```
      UDUMPC - CORE-DISK DUMPS


ERR LOC   OBJECT CODE       ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  13/01/22  PAGE  48

                            4766 *
      1945              1946 4767 UPT810 DS    CL2                      IAR SAVE LOCATION
      1947              1948 4768 UPADDR DS    CL2                      ADDRESS DATA AREA
      1949              1949 4769 UPDATA DS    CL1                      DATA CHARACTER
      194A              194A 4770 UPSTS  DS    CL1                      KEYBOARD STATUS BYTE
      194B              194C 4771 UPTCNT DS    CL2                      CHARACTER COUNT
                            4772 *
                            4773 *   UPATCH CONSTANT AREA
                            4774 *
      194D 39           194D 4775 UPTH39 DC    XL1'39'                  ADD TO ALPHAS PLUS C0 TO GET E
      194E C0           194E 4776 UPTHC0 DC    XL1'C0'                  ADD TO NUMERICS TO OBTAIN EBCD
      194F 0001         1950 4777 UPTD1  DC    XL2'0001'                CONSTANT
      1951 0000         1952 4778 UPTD2  DC    XL2'0000'                CONSTANT
      1953 0004         1954 4779 UPTD4  DC    XL2'0004'                CONSTANT
      1955 09           1955 4780 UPTD9  DC    XL1'09'                  CONSTANT
                            4781 *
                            4782 *   ROUTINE TO HANDLE ILLEGAL CHARACTERS
                            4783 *
                            4784 *UPBADC PRNT  UPTM03
      1956 C0 87 0707    4785 UPBADC B      $$PRNT                PRINT ON MATRIX PRINTER
      195A 1991         195B 4786        DC    AL2(UPTM03)              PPL ADDRESS
                            4787 *** END OF EXPANSION ***
      195C C0 87 1860    4788        B      UPATCH                START OVER
      1960              1961 4789 UPBUF  DS    CL2                      ADDRESS INTO LINE BUFFER
      1962 0606         1963 4790 UPADRR DC    AL2($$INLN-1)            POINTER TO LINE BUFFER MINUS 1
                            4791 *
                            4792 *   MESSAGE AREA - CONSTANTS
                            4793 *
                       1964 4794 UPT920 EQU   *                        BEGINNING OF DATA
                       1964 4795 UPT930 EQU   *                        BEGINNING OF DATA
      1964 C5D5E3C5D940D7C1 198C 4796        DC    CL(UPTL02)'ENTER PATCH DATA, USE SPACE FOR NO CHANGE'
                            4797 *UPTM02 PPL    FUNC=@PRETR,CNT=UPTL02,CADDR=UPT930
                       198D 4798 UPTM02 EQU   *                        PPL ADDRESS
      198D C0           198D 4799        DC    AL1(@PRETR)             FUNCTION REQUESTED
      198E 29           198E 4800        DC    AL1(UPTL02)             PRINT COUNT
      198F 1964         1990 4801        DC    AL2(UPT930)             DATA ADDRESS
                            4802 *** END OF EXPANSION ***
                            4803 *UPTM03 PPL    FUNC=@PRETR,CNT=UPTL03,CADDR=UPT940
                       1991 4804 UPTM03 EQU   *                        PPL ADDRESS
      1991 C0           1991 4805        DC    AL1(@PRETR)             FUNCTION REQUESTED
      1992 01           1992 4806        DC    AL1(UPTL03)             PRINT COUNT
      1993 1995         1994 4807        DC    AL2(UPT940)             DATA ADDRESS
                            4808 *** END OF EXPANSION **
      1995 6F           1995 4809 UPT940 DC    CL1'?'                   KEH ?
                            4810 *UPTM04 PPL FUNC=@RETRN,CNT=@RTRNC
                       1996 4811 UPTM04 EQU   *                        PPL ADDRESS
      1996 80           1996 4812        DC    AL1(@RETRN)             FUNCTION REQUESTED
      1997 80           1997 4813        DC    AL1(@RTRNC)             PRINT COUNT
      1998 0000         1999 4814        DC    AL2(*-*)                DATA ADDRESS
                            4815 *** END OF EXPANSION ***

                            4817 ***********************************************************************
                            4818 *   PATCH INTERFACE ROUTINE                                          *
                            4819 ***********************************************************************
                            4820 *ZCPEST ENTER EXIT=ZCPX1,,,@ARR
                       199A 4821 ZCPEST EQU   *                        MODULE ENTRY POINT
```

```
      UDUMPC - CORE-DISK DUMPS


ERR LOC   OBJECT CODE     ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  13/01/22  PAGE  49

      199A 34 08 19E6          4822          ST    ZCPX12+@OP1,@ARR        SAVE RETURN ADDRESS
                               4823 *** END OF EXPANSION ***
      199E 34 02 12C3          4824          ST    ZCSADD,@XR              PUT XR IN TEST CELL
      19A2 34 02 1A34          4825          ST    ZCPSVA,@XR              SAVE REAL ADDRESS
      19A6 C0 80 19B5          4826 ZCP010 BC     ZCP020,@NOP             FIRST TIME SWITCH
      19AA 39 FF 12C3          4827          TBF   ZCSADD,ZCPFFF           TEST FOR HEX 100 BOUNDARY
      19AE F2 90 12            4828          JF    ZCP030                  SKIP CALL IF ODD
      19B1 C0 87 1A1D          4829          B     ZCPWRT                  GO WRITE OLD SECTOR
      19B5 C0 87 0000          4830 ZCP020 B     *-*                     CALL CORRECT TEST
      19B9 3C 80 19A7          4831 ZCP025 MVI    ZCPFTS,@NOP             RESET SWITCH
      19BD 0C 01 1A2F 12D1     4832          MVC   ZCPDPL+@DSAD(@DADDR),ZCSDPL+@DSAD  MODIFY DISK ADDR
      19C3 38 FF 12C4          4833 ZCP030 TBN    ZCSREL,ZCPFFF           TEST FOR REA.. CORE
      19C7 F2 10 19            4834          JT    ZCPX12                  GO TO EXIT
      19CA 38 F0 12C4          4835          TBN   ZCSREL,ZCPF00           TEST FOR EXCEEDED CORE
      19CE F2 90 08            4836          JF    ZCP040                  GO SET ADDR EQUAL TO BUFFER
      19D1 3A FF 12C4          4837          SBN   ZCSREL,ZCSFFF           SET INDICATOR FOR REAL CORE
      19D5 C0 87 1141          4838          B     ZCD060                  GO REPEAT MESSAGE
      19D9 0C 00 12C2 1A35     4839 ZCP040 MVC    ZCSADD-1(1),ZCPADD-1    SET ADDR FOR SAVED CORE = BUFF
      19DF 35 02 12C3          4840          L     ZCSADD,@XR              PLACE IN XR
                               4841 *ZCPX1  EXIT  ,,RETURN
      19E3 C0 87 0000          4842 ZCPX12 B     *-*                     RETURN TO CALLING PROGRAM
                               4843 *** END OF EXPANSION ***
                               4844 *ZCPRST ENTER EXIT=ZCPX2,,,@ARR
      19E7 4845 ZCPRST EQU    *                       MODULE ENTRY POINT
      19E7 34 08 19F2          4846          ST    ZCPX22+@OP1,@ARR        SAVE RETURN ADDRESS
                               4847 *** END OF EXPANSION ***
      19EB 35 02 1A34          4848          L     ZCPSVA,@XR              RESTORE XR
                               4849 *ZCPX2  EXIT ,,RETURN
      19EF C0 87 0000          4850 ZCPX22 B     *-*                     RETURN TO CALLING PROGRAM
                               4851 *** END OF EXPANSION ***
      19F3 4852 ZCPDSK EQU    *                       DISK TEST ROUTINE
      19F3 3B FF 12C4          4853          SBF   ZCSREL,ZCPFFF           SET CORE IND TO SAVED
      19F7 0C 01 1044 1172     4854          MVC   DL2RAD,ZCD090(@DADDR)   SET UP DISK PATCH ADDR
      19FD 0C 00 12D1 1952     4855          MVC   ZCSDPD,UPTD2            PUT ZEROS IN ZCSDPL
      1A03 C0 87 1A0D          4856 ZCPTST BC     ZCPDS1,@UCB             SECTOR CARRY SWITCH
      1A07 0E 01 12D1 1950     4857          ALC   ZCSDPD,UPTD1(@DADDR)    ADD ONE TO DADOR
                               4858 *ZCPDS1 DSKL2 ZCSIPL,WAIT             GET DISK DATA
      1A0D C0 87 0FAC          4859 ZCPDS1 B     DL2ICS                  PERFORM RELATIVE DISK OP
      1A11 12CF         1A12 4860          DC    AL2(ZCSDPL)             DPL ADDRESS
      1A13 C0 87 0025          4861          B     $DISKN                  WAIT AND CHECK DISK ERRORS
      1A17 057F         1A18 4862          DC    AL2($WAITF)             WAIT DPL ADDRESS
                               4863 *** END OF EXPANSION ***
      1A19 C0 87 19B9          4864          B     ZCP025                  RETURN TO TEST
                               4865 *   END OF ROUTINE
                               4866 *ZCPWRT DSKL2 ZCPDPL,WAIT
      1A1D C0 87 0FAC          4867 ZCPWRT B     DL2ICS                  PERFORM RELATIVE DISK OP
      1A21 1A2D         1A22 4868          DC    AL2(ZCPDPL)             DPL ADDRESS
      1A23 C0 87 0025          4869          B     $DISKN                  WAIT AND CHECK DISK ERRORS
      1A27 057F         1A28 4870          DC    AL2($WAITF)             WAIT DPL ADDRESS
                               4871 *** END OF EXPANSION ***
      1A29 C0 87 19B5          4872          B     ZCP020
                               4873 *ZCPDPL DPI   FUNC=@DPUT,CNT=ZCPCNT,CADDR=ZCSDAT
      1A2D 4874 ZCPDPL EQU    *                       DISK PARAMETER LIST
      1A2D 02           1A2D 4875          DC    AL1(@DPUT)              REQUESTED FUNCTION
      1A2E 00           1A2E 4876          DC    AL1(*-*)                CYLINDER ADDRESS
      1A2F 00           1A2F 4877          DC    AL1(*-*)                HEAD/SECTOR/DRIVE/DISK SPEC
```

```
    1A30 01               1A30 4878          DC    AL1(ZCPCNT)              SECTOR COUNT
    1A31 1F00             1A32 4879          DC    AL2(ZCSDAT)              BUFFER ADDRESS
                               4880 *** END OF EXPANSION ***
                          19A7 4881 ZCPFTS EQU   ZCP010+@Q              POINTER TO 1ST TIME SWITCH
                          0001 4882 ZCPCNT EQU   1
                          00FF 4883 ZCPFFF EQU   X'FF'
                          00F0 4884 ZCPF00 EQU   X'F0'
    1A33                  1A34 4885 ZCPSVA DS    CL(@DADDR)
    1A35 1F00             1A36 4886 ZCPADD DC    XL(@CADDR)'1F00'
    1A37                  1A37 4887 UPTDSK DS    CL1
                               4888 *  END OF STORAGE
    1A38 1223             1A39 4889 ZCPZCS DC    AL(@CADDR)(ZCSAVE)
    1A3A 19F3             1A3B 4890 ZCPUCS DC    AL(@CADDR)(ZCPDSK)
```

```
                                4892 ****************************************************************************
                                4893 *   UDUMPD DUMPS FROM DISK STARTING AT ADDRESS SPECIFIED IN CALL AND    *
                                4894 *   WILL CONTINUE UNTIL END ADDRESS SPECIFIED IN CALL IS REACHED        *
                                4895 *                                                                       *
                                4896 *         CALLING SEQUENCE IS:                                          *
                                4897 *                 B    UDUMPD                                           *
                                4898 *                 DC   2X'START ADDRESS'                                *
                                4899 *                 DC   2X'END ADDRESS'                                  *
                                4900 ****************************************************************************
                                4901 *
                                4902 * DISK INPUT/DUMP OUTPUT AREA
                                4903 *
                    0011 4904 UDDB17 EQU    17                           17
                    004C 4905 UDDB76 EQU    76                           76
                    001B 4906 UDDB27 EQU    27                           27
                    0004 4907 UDDB04 EQU    4                            4
  1A3C 0001         1A3D 4908 UDD901 DC     XL(@CADDR)'0001'
                                4909 *
                                4910 * PAGE HEADER
                                4911 *
                    1A3E 4912 UDD919 EQU    *
  1A3E E2C5C3E3D6D940C1 1A4E 4913 UDD910 DC     CL(UDDB17)'SECTOR ADDR=      '
  1A4F 4040404040404040 1A9A 4914        DC     CL(UDDB76)' '
  1A9B D9C5D3C1E3C9E5C5 1AB5 4915 UDD920 DC     CL(UDDB27)'RELATIVE SECTOR NUMBER=0000'

                    1AD4 4917        USING UDBASE,@BR
                                4918 *
                                4919 * DISK DUMP ENTRY POINT
                                4920 *
                                4921 *UDUMPD PRNT   ZCDM01                      ASK FOR SECTOR COUNT
  1AB6 C0 87 0707         4922 UDUMPD B      $$PRNT                     PRINT ON MATRIX PRINTER
  1ABA 121B          1ABB 4923        DC     AL2(ZCDM01)                PPL ADDRESS
                                4924 *** END OF EXPANSION ***

  1ABC C0 87 0D31         4926        B      ZUTKEY                     GET KEY DATA
  1AC0 C0 87 1CE2         4927        B      C4BIN2                     CONVERT TO HEX
  1AC4 C0 04 1AB6         4928        BNH    UDUMPD                     ASK AGAIN IF IN ERROR
  1AC8 0C 01 1B45 1D4C    4929        MVC    UDD820,C4BVAL(@DADDR)      MOVE IN HEX VALUE
  1ACE 0C 01 1B40 1174    4930        MVC    UDD800,ZCD100(@DADDR)      MOVE IN ADDRESS
  1AD4 C2 01 1AD4         4931 UDD100 LA     UDBASE,@BR                 SET BASE REGISTER
  1AD8 07 03 1AB5 1AB5    4932        SZ     UDD920(UDDB04),UDD920(UDDB04)  ZERO REL SECTOR NO.
                                4933 *
                                4934 * CONVERT SECTOR ADDRESS & BEGIN PRINTING, READ AND DUMP SECTOR.
                                4935 *
                    1ADE 4936 UDD110 EQU    *
  1ADE C0 87 0DB9         4937        B      CVBHEX                     CONVERT DISK ADOR
  1AE2 02            1AE2 4938        DC     XL1'02'                    LENGTH
  1AE3 1B3F          1AE4 4939        DC     AL2(UDD800-1)              INPUT ADDR
  1AE5 1A4B          1AE6 4940        DC     AL2(UDD910-3)              OUTPUT FIELD
                    1AE7 4941 ZAPP06 EQU    *                          PRINTER PATCH
                                4942 *        PRNT   UDDM01                       PRINT HEADER
  1AE7 C0 87 0707         4943        B      $$PRNT                     PRINT ON MATRIX PRINTER
  1AEB 1B46          1AEC 4944        DC     AL2(UDDM01)                PPL ADDRESS
                                4945 *** END OF EXPANSION ***
                    1AED 4946 ZAPP07 EQU    *                          PRINTER PATCH
                                4947 *        PRNT   UDC886                       PRINTER SPACE
```

```
    1AED C0 87 0707            4948          B     $$PRNT              PRINT ON MATRIX PRINTER
    1AF1 16C2            1AF2 4949          DC    AL2(UDC886)         PPL ADDRESS
                             4950 *** END OF EXPANSION ***
                             4951 *        DISK UDDDPL,WAIT              GET DISK DATA
    1AF3 C0 87 0025            4952          B     $DISKN              PERFORM PHYSICAL DISK OP
    1AF7 1B3E            1AF8 4953          DC    AL2(UDDDPL)         DPL ADDRESS
    1AF9 C0 87 0025            4954          B     $DISKN              WAIT AND CHECK DISK ERRORS
    1AFD 057F            1AFE 4955          DC    AL2($WAITF)         WAIT DPL ADDRESS
                             4956 *** END OF EXPANSION ***
                        1AFF 4957 ZAPP08 EQU   *                   PRINTER PATCH
                             4958 *        PRNT  UDC880               PRINT HEADER
    1AFF C0 87 0707            4959          B     $$PRNT              PRINT ON MATRIX PRINTER
    1B03 1646            1B04 4960          DC    AL2(UDC880)         PPL ADDRESS
                             4961 *** END OF EXPANSION ***
                             4962 *        PRNT  $WAITF                DO WAIT
    1B05 C0 87 0707            4963          B     $$PRNT              PRINT ON MATRIX PRINTER
    1B09 057F            1B0A 4964          DC    AL2($WAITF)         PPL ADDRESS
                             4965 *** END OF EXPANSION ***
    1B0B C0 87 0DA5            4966          B     ZUTIRI              * GO TEST FOR INTERRUPTS
    1B0F C0 87 12F0            4967          B     UDUMP1              DUMP SECTOR CORE BUFFER
    1B13 0000            1B14 4968          DC    XL2'0000'           START ADDRESS
    1B15 00FF            1B16 4969          DC    XL2'00FF'           END ADDRESS
                        1B17 4970 ZAPP09 EQU   *                   PRINTER PATCH
                        1B17 4971 ZAPP10 EQU   *                   PRINTER PATCH
                             4972 *        PRNT  UDC886                SPACE
    1B17 C0 87 0707            4973          B     $$PRNT              PRINT ON MATRIX PRINTER
    1B1B 16C2            1B1C 4974          DC    AL2(UDC886)         PPL ADDRESS
                             4975 *** END OF EXPANSION ***
                             4976 *
                             4977 * TEST FOR END OF DUMP
                             4978 *
    1B1D 4F 01 71 1A3D         4979          SLC   UDD820(@CADDR,@BR),UDD901 REDUCE SECTOR COUNT
    1B22 F2 84 04             4980          JH    UDD400              CONTINUE IF SECTOR COUNT PLUS
    1B25 C0 87 0C39            4981          B     ZUT020              RETURN TO MONITOR
                             4982 *
                             4983 * UPDATE DISK ADDRESS AND CONTINUE
                             4984 *
    1B29 16 30 1AB5 67         4985 UDD400 AZ    UDD920(UDDB04),UDD700(1,@BR)  UPDATE REL SECTR ADDR
    1B2E 5E 01 6C 69          4986 UDD410 ALC   UDD800(@DADDR,@BR),UDD710(,@BR)   UPDATE SECTOR NUMBER
    1B32 78 60 6C            4987          TBN   UDD800(,@BR),X'60'
    1B35 D0 10 5A            4988          BT    UDD410(,@BR)        LOOP TO SKIP ILLEGAL DADDRS
    1B38 D0 87 0A            4989          B     UDD110(,@BR)        CONTINUE
                             4990 *
                             4991 * CONSTANT/WORK AREA
                             4992 *
    1B3B F1              1B3B 4993 UDD700 DC    CL1'1'              CONSTANT
    1B3C 0004            1B3D 4994 UDD710 DC    XL2'0004'           CONSTANT
                             4995 *
                             4996 * DISK PARAMETER LIST
                             4997 *
                             4998 *UDDDPL DPL   FUNC=@DGET,DADDR=*=*,CNT=UDD801,CADDR=ZCSDAT
                        1B3E 4999 UDDDPL EQU   *                   DISK PARAMETER LIS
    1B3E 01             1B3E 5000          DC    AL1(@DGET)          REQUESTED FUNCTION
    1B3F 0000           1B40 5001          DC    AL2(*-*)            DISK ADDRESS
    1B41 01             1B41 5002          DC    AL1(UDD801)         SECTOR COUNT
    1B42 1F00           1B43 5003          DC    AL2(ZCSDAT)         BUFFER ADDRESS
```

```
      UDUMPD - F.E. UTILITY DISK DUMP

ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT            VER 15, MOD 00  13/01/22  PAGE  53

                            5004 *** END OF EXPANSION ***

                    0001 5006 UDD801 EQU   X'01'                           *
                    1B40 5007 UDD800 EQU   UDDDPL+@DSAD            DISK ADDRESS TO BE READ
                         5008 *
                    1AD4 5009 UDBASE EQU   UDD100
   1B44             1B45 5010 UDD820 DS    CL2                             *
                         5011 *UDDM01 PPL   FUNC=@PRETR,CNT=UDDL01,CADDR=UDD919
                    1B46 5012 UDDM01 EQU   *                         PPL ADDRESS
   1B46 C0          1B46 5013        DC    AL1(@PRETR)               FUNCTION REQUESTED
   1B47 78          1B47 5014        DC    AL1(UDDL01)               PRINT COUNT
   1B48 1A3E        1B49 5015        DC    AL2(UDD919)               DATA ADDRESS
                         5016 *** END OF EXPANSION ***

                    0078 5018 UDDL01 EQU   120                             *
                         5019 ***********************************************************************
```

```
                           5021 ******************************************************************************
                           5022 *                                                                            *
                           5023 *                       OPTION 'DC' - DISK COMPARE                           *
                           5024 *                                                                            *
                           5025 *    SELECTED DISK AREAS WILL BE READ INTO BUFFERS AT HEX 1F00 TO           *
                           5026 *    1FFF AND HEX 1E00 TO 1EFF.  THE DATA WILL BE COMPARED BYTE BY          *
                           5027 *    BYTE UNTIL ALL DATA IS COMPARED.  THEN THE DISK ADDRESSES WILL         *
                           5028 *    BE INCREASED BY ONE SECTOR AND THE OPERATION WILL REPEAT.  THIS        *
                           5029 *    PROCESS WILL CONTINUE UNTIL THE NUMBER OF SECTORS SPECIFIED            *
                           5030 *    HAVE BEEN COMPARED.                                                    *
                           5031 *                                                                            *
                           5032 *        WHEN AN UNEQUAL CONDITION IS DETECTED, THE FOLLOWING INFOR-        *
                           5033 *        NATION WILL PRINT.                                                 *
                           5034 *                                                                            *
                           5035 *          DADDR1 DADDR2 DISPL DK1 DK2                                      *
                           5036 *                                                                            *
                           5037 *           XXXX   XXXX   XX  XX  XX                                        *
                           5038 *                                                                            *
                           5039 *        DADDRL= DISK ADDRESS OF FIRST DSK, DADDR2= DISK ADDR 2ND DSK      *
                           5040 *        DISPL = DISPLACEMENT WITHIN SECTOR.                                *
                           5041 *        DK1 = DATA ON DISK1 AND DK2 = DATA ON DISK2.                       *
                           5042 *                                                                            *
                           5043 ******************************************************************************

                           5045 ******************************************************************************
                           5046 *                                                                            *
                           5047 *              DISK COPY ROUTINE                                             *
                           5048 *                                                                            *
                           5049 ******************************************************************************
                           5050 *ZDCCOP ENTER BASE=ZDCERR
                      1BF1 5051        USING ZDCERR,@BR              BASE ADDRESS SPECIFICATION
                      1B4A 5052 ZDCCOP EQU   *                      MODULE ENTRY POINT
     1B4A C2 01 1BF1       5053        LA    ZDCERR,@BR             LOAD BASE REGISTER
                           5054 *** END OF EXPANSION ***
     1B4E 4C 01 C4 1172    5055        MVC   ZDCDP2+@DSAD(@DADDR,@BR),ZCD090  MOVE IN READ ADDR
     1B53 4C 01 CA 1174    5056        MVC   ZDCDP3+@DSAD(@DADDR,@BR),ZCD100  MOVE IN WRITE ADDR
                           5057 *       DISK  ZOCDP2,WAIT              GET DISK DATA
     1B58 C0 87 0025       5058        B     $DISKN                 PERFORM PHYSICAL DISK OP
     1B5C 1CB3        1B5D 5059        DC    AL2(ZDCDP2)            DPL ADDRESS
     1B5E C0 87 0025       5060        B     $DISKN                 WAIT AND CHECK DISK ERRORS
     1B62 057F        1B63 5061        DC    AL2($WAITF)            WAIT DPL ADDRESS
                           5062 *** END OF EXPANSION ***
                           5063 *       DISK  ZDCDP3,WAIT             WRITE ONE SECTOR
     1B64 C0 87 0025       5064        B     $DISKN                 PERFORM PHYSICAL DISK OP
     1B68 1CB9        1B69 5065        DC    AL2(ZDCDP3)            DPL ADDRESS
     1B6A C0 87 0025       5066        B     $DISKN                 WAIT AND CHECK DISK ERRORS
     1B6E 057F        1B6F 5067        DC    AL2($WAITF)            WAIT DPI ADDRESS
                           5068 *** END OF EXPANSION ***
     1B70 C0 87 0C39       5069        B     ZUT020                 RETURN TO MONITOR

                           5071 *ZDCENT ENTER BASE=ZDCERR
                      1BF1 5072        USING ZDCERR,@BR              BASE ADDRESS SPECIFICATION
                      1B74 5073 ZDCENT EQU   *                      MODULE ENTRY POINT
     1B74 C2 01 1BF1       5074        LA    ZDCERR,@BR             LOAD BASE REGISTER
                           5075 *** END OF EXPANSION ***
     1B78 7C 80 01         5076        MVI   ZDC040+@Q(,@BR),@NOP   TURN OFF SWITCH
```

```
      ZDCOMP - DISK COMPARE PROGRAM


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT           VER 15, MOD 00  13/01/22  PAGE  55

    1B7B 4C 01 BE 1172       5077         MVC    ZDCDPL+@DSAD(@DADDR,@BR),ZCD090  MOVE IN DISK1 DADDR
    1B80 4C 01 C4 1174       5078         MVC    ZDCDP2+@DSAD(@DADDR,@BR),ZCD100  MOVE IN DISK2 DADDR
    1B85 0C 0C 118B 11D9     5079         MVC    ZCDMV1,ZCDSEC                MOVE IN SECTOR COUNT MSG
                             5080 *ZDCEN2 PRNT   ZUM01                       PRINT REQUEST
    1B8B C0 87 0707          5081 ZDCEN2 B       $$PRNT                      PRINT ON MATRIX PRINTER
    1B8F 121B           1B90 5082         DC      AL2(ZCDM01)                 PPL ADDRESS
                             5083 *** END OF EXPANSION ***
    1B91 C0 87 0DA5          5084         B       ZUTIRI                      GO CHECK FOR INTERRUPTS
    1B95 C0 87 0D31          5085         B       ZUTKEY                      GET KEY DATA
    1B99 C0 87 1CE2          5086         B       C4BIN2                      CONVERT TO HEX
    1B9D C0 04 1B8B          5087         BNH     ZDCEN2                      ASK AGAIN IF IN ERROR
    1BA1 0C 01 1CAC 1D4C     5088         MVC     ZDCCNT,C4BVAL(@DADDR)       MOVE IN VALUE IN HEX
                             5089 *ZDC010 DISK   ZDCDP1,WAIT                 GET DISK AREA1
    1BA7 C0 87 0025          5090 ZDC010 B       $DISKN                      PERFORM PHYSICAL DISK OP
    1BAB 1CAD           1BAC 5091         DC      AL2(ZDCDPL)                 DPI ADDRESS
    1BAD C0 87 0025          5092         B       $DISKN                      WAIT AND CHECK DISK ERRORS
    1BB1 057F           1BB2 5093         DC      AL2($WAITF)                 WAIT DPL ADDRESS
                             5094 *** END OF EXPANSION ***
                             5095 *      DISK   ZDCDP2,WAIT                 GET DISK AREA2
    1BB3 C0 87 0025          5096         B       $DISKN                      PERFORM PHYSICAL DISK OP
    1BB7 1CB3           1BB8 5097         DC      AL2(ZDCDP2)                 DPL ADDRESS
    1BB9 C0 87 0025          5098         B       $DISKN                      WAIT AND CHECK DISK ERRORS
    1BBD 057F           1BBE 5099         DC      AL2($WAITF)                 WAIT DPL ADDRESS
                             5100 *** END OF EXPANSION ***
    1BBF 0D FF 1EFF 1FFF     5101         CLC     ZDCDT1(ZDCCPL),ZDCDT2       COMPARE TWO DISK AREAS
    1BC5 C0 01 1BF1          5102         BNE     ZDCERR                      GO FIND UNEQUAL BYTES
    1BC9 C0 87 0DA5          5103         B       ZUTIRI                      GO CHECK FOR INTERRUPTS
                        1BCD 5104 ZDCADV EQU     *                           POINTER TO ADDRESS UPDATE
    1BCD 5F 01 BB 83         5105         SLC     ZDCCNT(,@BR),ZDCONE(@DADDR,@BR)  REDUCE SECTOR COUNT BY
    1BD1 C0 04 0F08          5106         BNH     UVMEND                      EXIT FROM TEST
    1BD5 4E 01 BE 1C82       5107 ZDC020 ALC     ZDCDPL+@DSAD(@DADDR,@BR),ZDCFOR  INCREASE ADDR BY HEX 4
    1BDA 78 60 BE            5108         TBN     ZDCDPL+@DSAD(,@BR),ZDCSIX   TEST FOR ILLEGAL ADDRESS
    1BDD C0 10 1BD5          5109         BT      ZDC020                      BUMP UP TO MISS ILLEGAL DADDR
    1BE1 4E 01 C4 1C82       5110 ZDC030 ALC     ZDCDP2+@DSAD(@DADDR,@BR),ZDCFOR  INCREASE ADDR BY HEX 4
    1BE6 78 60 C4            5111         TBN     ZDCDP2+@DSAD(,@BR),ZDCSIX   TEST FOR ILLEGAL ADDRESS
    1BE9 C0 10 1BE1          5112         BT      ZDC030                      BUMP UP TO MISS ILLEGAL DADDR
    1BED C0 87 1BA7          5113         B       ZDC010                      GO TO READ MORE
                        1BF1 5114 ZDCERR EQU     *                           ERROR DIFINITION ENTRY POINT
    1BF1 D0 80 0C            5115 ZDC040 BC      ZDC050(,@BR),@NOP           GO PRINT ERROR
    1BF4 7C 87 01            5116         MVI     ZDC040+@Q(,@BR),@UCB        TURN ON SWITCH
                             5117 *      PRNT   ZDCPL1
    1BF7 C0 87 0707          5118         B       $$PRNT                      PRINT ON MATRIX PRINTER
    1BFB 1CBF           1BFC 5119         DC      AL2(ZDCPL1)                 PPL ADDRESS
                             5120 *** END OF EXPANSION ***
                        1BFD 5121 ZDC050 EQU     *
    1BFD 5C 03 15 8B         5122         MVC     ZDCCMP+@OP2(ZDCBN4,@BR),ZDCPTR(,@BR)  SET COMP ADDRESS
    1C01 0D 00 0000 0000     5123 ZDCCMP CLC     *-*(ZDCDL1),*-*             COMPARE A BYTE OF EACH AREA
    1C07 C0 01 1C1E          5124         BNE     ZDCPRT                      PRINT OUT DIFFERENCE
    1C0B C0 87 0DA5          5125         B       ZUTIRI                      GO CHECK FOR INTERRUPTS
    1C0F 5E 03 15 8F         5126 ZDC055 ALC     ZDCCMP+@OP2(@INST4,@BR),ZDCBUM(,@BR)  INCREASE POINTERS
    1C13 78 20 14            5127         TBN     ZDCCMP+@DOP2(,@BR),ZDCTWO   CHECK FOR END OF COMPARE
    1C16 C0 10 1BCD          5128         BT      ZDCADV                      GO ADVANCE ADDRESS
    1C1A C0 87 1C01          5129         B       ZDCCMP                      ELSE CHECK NEXT BYTES
                        1C1E 5130 ZDCPRT EQU *                               ERROR PRINT OUT IN DETAIL
                             5131 *      PRNT   $WAITF                      PRINTER WAIT
    1C1E C0 87 0707          5132         B       $$PRNT                      PRINT ON MATRIX PRINTER
```

```
     ZDCOMP - DISK COMPARE PROGRAM


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  13/01/22  PAGE  56

    1C22 057F             1C23 5133          DC    AL2($WAITF)              PPL ADDRESS
                               5134 *** END OF EXPANSION ***

    1C24 5C 01 AF BE           5136          MVC   ZDCWK1(@CADDR,@BR),ZDCDPL+@DSAD(,@BR)  MOVE DADDR1 TO WKA
    1C28 5C 01 B2 C4           5137          MVC   ZDCWK2(@CADDR,@BR),ZDCDP2+@DSAD(,@BR)  MOVE DADDR2 TO WKA
    1C2C 5C 00 B5 15           5138          MVC   ZDCWK3(1,@BR),ZDCCMP+@OP2(,@BR)  MOVE DISPL TO WORK AREA
    1C30 5C 01 4B 13           5139          MVC   ZDC060+@DOP2(@CADDR,@BR),ZDCCMP+@OP1(,@BR)  MODIFY MOVE
    1C34 5C 01 50 15           5140          MVC   ZDC070+@DOP2(@CADDR,@BR),ZDCCMP+@OP2(,@BR)  MODIFY MOVE
    1C38 4C 00 B7 0000         5141 ZDC060 MVC   ZDCWK4(1,@BR),*-*        MOVE DATA1 TO WORK AREA
    1C3D 4C 00 B9 0000         5142 ZDC070 MVC   ZDCWK5(1,@BR),*-*        MOVE DATA2 TO WORK AREA
    1C42 C0 87 0DB9            5143          B     CVBHEX                  GO TO CONVERT ROUTINE
    1C46 0D              1C46 5144          DC    XL1'0D'                 LENGTH OF CONVERSION
    1C47 1C9E            1C48 5145          DC    AL2(ZDCWKA)             INPUT AREA FOR CONVERT
    1C49 1C83            1C4A 5146          DC    AL2(ZDCPRA)             OUTPUT AREA FOR CONVERT
    1C4B 5C 01 93 87           5147          MVC   ZDCPR1(@CADDR,@BR),ZDCBLA(,@BR)  MOVE BLANKS
    1C4F 5C 01 99 87           5148          MVC   ZDCPR2(@CADDR,@BR),ZDCBLA(,@BR)  MOVE BLANKS
    1C53 5C 03 A1 87           5149          MVC   ZDCPR3(ZDCBN4,@BR),ZDCBLA(,@BR)  MOVE BLANKS
    1C57 5C 01 A5 87           5150          MVC   ZDCPR4(@CADDR,@BR),ZDCBLA(,@BR)  MOVE BLANKS
    1C5B 5C 01 A9 87           5151          MVC   ZDCPR5(@CADDR,@BR),ZDCBLA(,@BR)  MOVE BLANKS
    1C5F 5C 00 AC 87           5152          MVC   ZDCPR6(1,@BR),ZDCBLA(,@BR)  MOVE BLANKS
                               5153 *        PRNT  ZDCPP2
    1C63 C0 87 0707            5154          B     $$PRNT                  PRINT ON MATRIX PRINTER
    1C67 1CC3            1C68 5155          DC    AL2(ZDCPP2)             PPL ADDRESS
                               5156 *** END OF EXPANSION ***
                               5157 *        PRNT  $WAITF                 WAIT FOR COMPARE TO PRINT
    1C69 C0 87 0707            5158          B     $$PRNT                  PRINT ON MATRIX PRINTER
    1C6D 057F            1C6E 5159          DC    AL2($WAITF)             PPL ADDRESS
                               5160 *** END OF EXPANSION ***
    1C6F C0 87 1BCD            5161          B     ZDCADV                  GO TO NEXT SECTORS

                               5163 ********************************************************************
                               5164 *
                               5165 *        SECTION STORAGE BLOCKS
                               5166 *
                               5167 ********************************************************************
    1C73 0001            1C74 5168 ZDCONE DC    XL2'0001'
    1C75 40404040        1C78 5169 ZDCBLA DC    XL4'40404040'           BLANKS FOR OUTPUT LINE
    1C79 1E001F00        1C7C 5170 ZDCPTR DC    XL4'1E001F00'
    1C7D 00010001        1C80 5171 ZDCBUM DC    XL4'00010001'
    1C81 0004            1C82 5172 ZDCFOR DC    XL2'0004'
                        1C83 5173 ZDCPRA EQU   *
    1C83                1C9D 5174          DS    CL27                    PRINT AREA
                        1C9E 5175 ZDCWKA EQU   *
    1C9E                1CAA 5176          DS    CL13                    WORK AREA
    1CAB 0000            1CAC 5177 ZDCCNT DC    XL2'0000'               SECTOR COUNTER
                        0004 5178 ZDCBN4 EQU   4                       4
                               5179 ********************************************************************
                               5180 *                                                                  *
                               5181 *        SECTION PARAMETER LISTS                                    *
                               5182 *                                                                  *
                               5183 ********************************************************************
                               5184 *ZDCDPL CIAL FUNC,IDGET,DADDR,ZDCDALCNT+ZDCDLLCADDR?ZOCCO1
                        1CAD 5185 ZDCDPL EQU   *                       DISK PARAMETER LIST
    1CAD 01              1CAD 5186          DC    AL1(@DGET)              REQUESTED FUNCTION
    1CAE 0000            1CAF 5187          DC    AL2(ZDCDA1)             DISK ADDRESS
    1CB0 01              1CB0 5188          DC    AL1(ZDCDL1)             SECTOR COUNT
```

```
      ZDCOMP - DISK COMPARE PROGRAM


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT           VER 15, MOD 00  13/01/22  PAGE  57

   1CB1 1E00              1CB2 5189          DC    AL2(ZDCC01)              BUFFER ADDRESS
                               5190 *** END OF EXPANSION ***
                               5191 *ZDCDP2 DPL FUNC=@DGET,DADDR=ZDCDA2,CNT=ZDCDLL,CADDR=ZDCCO2
                          1CB3 5192 ZDCDP2 EQU   *                         DISK PARAMETER LIST
   1CB3 01               1CB3 5193          DC    AL1(@DGET)               REQUESTED FUNCTION
   1CB4 0000             1CB5 5194          DC    AL2(ZDCDA2)              DISK ADDRESS
   1CB6 01               1CB6 5195          DC    AL1(ZDCDL1)              SECTOR COUNT
   1CB7 1F00             1CB8 5196          DC    AL2(ZDCC02)              BUFFER ADDRESS
                               5197 *** END OF EXPANSION ***
                               5198 *ZDCDP3 DPL   FUNC=@DPUT,DADDR=ZDCDA2,CNT=ZDCDL1,CADDR=ZDCCO2
                          1CB9 5199 ZDCDP3 EQU   *                         DISK PARAMETER LIST
   1CB9 02               1CB9 5200          DC    AL1(@DPUT)               REQUESTED FUNCTION
   1CBA 0000             1CBB 5201          DC    AL2(ZDCDA2)              DISK ADDRESS
   1CBC 01               1CBC 5202          DC    AL1(ZDCDL1)              SECTOR COUNT
   1CBD 1F00             1CBE 5203          DC    AL2(ZDCC02)              BUFFER ADDRESS
                               5204 *** END OF EXPANSION ***
                               5205 *ZDCPL1 PPL   FUNC=@PRETR,CNT=ZDCL01,CADDR=ZDCM01
                          1CBF 5206 ZDCPL1 EQU   *                         PPL ADDRESS
   1CBF C0               1CBF 5207          DC    AL1(@PRETR)              FUNCTION REQUESTED
   1CC0 1B               1CC0 5208          DC    AL1(ZDCL01)              PRINT COUNT
   1CC1 1CC7             1CC2 5209          DC    AL2(ZDCM01)              DATA ADDRESS
                               5210 *** END OF EXPANSION ***
                               5211 *ZDCPP2 PPL   FUNC=@PRETR,CNT=ZDCL01,CADDR=ZDCPRA
                          1CC3 5212 ZDCPP2 EQU   *                         PPL ADDRESS
   1CC3 C0               1CC3 5213          DC    AL1(@PRETR)              FUNCTION REQUESTED
   1CC4 1B               1CC4 5214          DC    AL1(ZDCL01)              PRINT COUNT
   1CC5 1C83             1CC6 5215          DC    AL2(ZDCPRA)              DATA ADDRESS
                               5216 *** END OF EXPANSION ***
                               5217 ******************************************************************
                               5218 *                                                                *
                               5219 *        SECTION PRINT MESSAGES                                   *
                               5220 *                                                                *
                               5221 ******************************************************************
                          001B 5222 ZDCL01 EQU   27                        HEADER MESSAGE LENGTH
                          1CC7 5223 ZDCM01 EQU   *
   1CC7 C4C1C4C4D9F140C4 1CE1 5224          DC    CL(ZDCL01)'DADDR1 DADDR2 DISPL DK1 DK2'
                               5225 ******************************************************************
                               5226 *                                                                *
                               5227 *        SECTION EQUATES                                          *
                               5228 *                                                                *
                               5229 ******************************************************************
                          0000 5230 ZDCDA1 EQU   X'0000'
                          0000 5231 ZDCDA2 EQU   X'0000'
                          0001 5232 ZDCDL1 EQU   1
                          1E00 5233 ZDCC01 EQU   X'1E00'                   POINTER TO READ AREA
                          1F00 5234 ZDCC02 EQU   X'1F00'                   POINTER TO READ AREA
                          1EFF 5235 ZDCDT1 EQU   X'1EFF'
                          1FFF 5236 ZDCDT2 EQU   X'1FFF'
                          0100 5237 ZDCCPL EQU   256                       LENGTH FOR COMPARE
                          0060 5238 ZDCSIX EQU   X'60'
                          0020 5239 ZDCTWO EQU   X'20'
                          1C84 5240 ZDCPR1 EQU   ZDCPRA+1                  POINTER TO 2 BYTE BLANK
                          1C8A 5241 ZDCPR2 EQU   ZDCPRA+7                  POINTER TO 2 BYTE BLANK
                          1C92 5242 ZDCPR3 EQU   ZDCPRA+15                 POINTER TO 4 BYTE BLANK
                          1C96 5243 ZDCPR4 EQU   ZDCPRA+19                 POINTER TO 2 BYTE BLANK
                          1C9A 5244 ZDCPR5 EQU   ZDCPRA+23                 POINTER TO 2 BYTE BLANK
```

```
                           1C9D 5245 ZDCPR6 EQU    ZDCPRA+26                  POINTER TO 1 BYTE BLANK
                           1CA0 5246 ZDCWK1 EQU    ZDCWKA+2                   POINTER TO DADDR1
                           1CA3 5247 ZDCWK2 EQU    ZDCWKA+5                   POINTER TO DADDR2
                           1CA6 5248 ZDCWK3 EQU    ZDCWKA+8                   POINTER TO DISPL
                           1CA8 5249 ZDCWK4 EQU    ZDCWKA+10                  POINTER TO DK DATA 1
                           1CAA 5250 ZDCWK5 EQU    ZDCWKA+12                  POINTER TO DK DATA 2
                                5251 **********   END OF SECTION **********
                                5252 *       $C4BD
```

```
                                5254+******************************************************************
                                5255+*FUNCTION -                                                     *
                                5256+*   SERIALLY REUSABLE SUBROUTINE TO CONVERT A 4 BYTE POSITIVE DECIMAL *
                                5257+*   NUMBER A 2 BYTE BINARY VALUE.                                *
                                5258+*   A 5 BYTE POSITIVE DECIMAL NUMBER.                            *
                                5259+*   ON ENTRY @XR POINTS TO THE LEFT BYTE OF THE DECIMAL VALUE.   *
                                5260+*   ON RETURN C4BVAL IS THE RIGHT BYTE OF THE 2 BYTES BINARY VALUE *
                                5261+*   WHICH MAY BE MODIFIED BY THE USER IN ANY WAY IN IT'S LOCATION. *
                                5262+*   THE 4 BYTES DECIMAL VALUE IS NOT ALTERED.                    *
                                5263+*   @XR IS NOT ALTERED.                                          *
                                5264+*   @BR IS SAVED AND RESTORED AT EXIT.                           *
                                5265+******************************************************************

                                5267+*                                                               *
                                5268+*                    INITIALIZATION                             *
                                5269+*                                                               *
                           1CE2 5270+C4BIN2 EQU   *                        ENTRY POINT
                           1CE2 5271+       USING C4BIN2,@BR               BASE VALUE
                                5272+*
    1CE2 34 01 1D44              5273+       ST    C4B800+@OP1,@BR          SAVE CALLERS BASE REGISTER
    1CE6 C2 01 1CE2              5274+       LA    C4BIN2,@BR               LOAD BASE VALUE
                                5275+*
    1CEA 74 08 66                5276+       ST    C4B850+@OP1(,@BR),@ARR   SAVE RETURN ADDRESS
                                5277+*
    1CED 74 02 6E                5278+       ST    C4BSAV(,@BR),@XR         SAVE VALUE OF POINTER
    1CF0 3C 0C 03CD              5279+       MVI   $CAERR,@@E122            SET ERROR CODE IN CASE
    1CF4 5C 01 6A 6B             5280+       MVC   C4BVAL(C4BLVL,@BR),C4BINI(,@BR) INIT VALUE TO ZERO
    1CF8 3C 04 1D51              5281+C4B100 MVI   C4B900,4                 INITLZ CHAR. COUNT
                                5282+*
                                5283+***    DETERMINE IF CHAR NUMERIC AND DECR CHAR COUNT

                                5284+*
    1CFC F2 80 32                5285+C4B200 JC    C4B600,@NOP              SET TO UCB IF IMBEDDED BLANKS
                                5286+*                                     * ALLOWED
    1CFF BD F0 00                5287+C4B300 CLI   0(,@XR),C4BLOW           THIS CHAR NUMERIC ?
    1D02 F2 82 35                5288+       JL    C4B700                   NO, GOTO RETURN
                                5289+*
    1D05 5F 00 6F 4E             5290+       SLC   C4B900(1,@BR),C4B590+@D1(,@BR) DECR CHAR COUNT
    1D09 F2 82 35                5291+       JL    C4B800                   BR TO ERROR EXIT IF TOO MANY
                                5292+*
                                5293+***    MULTIPLY PREVIOUS VALUE BY TEN
                                5294+*
    1D0C 5E 01 6A 6A             5295+       ALC   C4BVAL(C4BLVL,@BR),C4BVAL(,@BR) DOUBLE PREVIOUS VALUE
    1D10 5C 01 68 6A             5296+       MVC   C4BWRK(C4BLVL,@BR),C4BVAL(,@BR) SAVE DOUBLE VALUE
    1D14 5E 01 6A 6A             5297+       ALC   C4BVAL(C4BLVL,@BR),C4BVAL(,@BR) QUADRUPLE PREVIOUS VALUE
    1D18 5E 01 6A 6A             5298+       ALC   C4BVAL(C4BLVL,@BR),C4BVAL(,@BR) OCTUPLE PREVIOUS VALUE
    1D1C 5E 01 6A 68             5299+       ALC   C4BVAL(C4BLVL,@BR),C4BWRK(,@BR) ADD IN SAVED DOUBLE
                                5300+*
                                5301+***    ADD IN VALUE OF THIS CHAR AND INCR POINTER
                                5302+*
    1D20 68 03 6C 00             5303+       MNN   C4BCHR(,@BR),0(,@XR)     FETCH NEMERIC VALUE OF NEW CHAR
    1D24 5E 01 6A 6C             5304+       ALC   C4BVAL(C4BLVL,@BR),C4BCHR(,@BR) INCR VALU BY THIS CHAR
                                5305+*
    1D28 E2 02 01                5306+       LA    @B1(,@XR),@XR            INCR POINTER TO NEXT CHAR
    1D2B D0 87 1A                5307+       B     C4B200(,@BR)             GOTO DO IT AGAIN
                                5308+*                                                               *
                                5309+*                    ROUTINE TO SCAN BLANKS                     *
```

```
     C4BIN2 - CONVERT DECIMAL TO BINARY ROUTINE
```

```
                            5310+*                                                 *
    1D2E E2 02 01           5311+C4B590 LA    @B1(,@XR),@XR            INCR POINTER TO NEXT CHAR
    1D31 BD 40 00           5312+C4B600 CLI   0(,@XR),@BLANK           IS THIS CHAR A BLANK ?
    1D34 D0 01 1D           5313+       BNE   C4B300(,@BR)             RETURN IF NOT
    1D37 D0 87 4C           5314+       B     C4B590(,@BR)             GET NEXT CHAR IF YES
```

```
                                   5316+*
                                   5317+***       ENDING ROUTINE
                                   5318+*
      1D3A 74 02 68                5319+C4B700 ST    C4BLEN(,@BR),@XR           PLACE VALUE OF POINTER
      1D3D 5F 01 68 6E             5320+        SLC   C4BLEN(2,@BR),C4BSAV(,@BR) SUBRACT ENTERING VALUE
                                   5321+*
      1D41 C2 01 0000              5322+C4B800 LA    *-*,@BR                    RESTORE CALLERS BR
                                   5323+*
      1D45 C0 87 0000              5324+C4B850 B     *-*                        RETURN TO CALLING ROUTINE
                                   5325+*                                                              *
                                   5326+*                        WORK AREA AND CONSTANT               *
                                   5327+*                                                              *
      1D49             1D4A        5328+C4BWRK DS    CL2                        SAVE AREA FOR DOUBLED VALUE
                                   5329+*
                       1D4B        5330+C4BYT1 EQU   *                          FIRST BYTE OF BINARY VALUE
      1D4B             1D4C        5331+C4BVAL DS    CL2                        SAVE AREA FOR BINARY VALUE
                                   5332+*
      1D4D 00          1D4D        5333+C4BINI DC    XL1'00'                    INITIALIZE WA TO ZERO
                                   5334+*
      1D4E             1D4E        5335+C4BCHR DS    CL1                        SAVE AREA FOR EACH NEW CHAR
      1D4E                         5336+        ORG   *-1                        INITIALIZE
      1D4E 00          1D4E        5337+        DC    XL1'00'                    * TO ZERO
                                   5338+*
      1D4F             1D50        5339+C4BSAV DS    CL2                        SAVE AREA FOR XR
                                   5340+*
      1D51             1D51        5341+C4B900 DS    CL1                        SAVE AREA FOR CHAR COUNTER
                                   5342+*                                                              *
                                   5343+*                        EQUATES FOR C4BIN2                    *
                                   5344+*                                                              *
                       1D4A        5345+C4BLEN EQU   C4BWRK                     ON RETURN WILL CONTAIN COUNT
                                   5346+*                                       * @XR INCREMENTED BY
                       0004        5347+C4BCHC EQU   4                          NUMBER OF CHAR TO CONVERT
                                   5348+*
                       00F0        5349+C4BLOW EQU   C'0'                       LOWEST NUMERIC CHARACTER
                                   5350+*
                       0002        5351+C4BLVL EQU   C4BVAL-C4BWRK              LENGTH OF BINARY VALUE
                                   5352+*
                       1CFD        5353+C4BLNK EQU   C4B200+@Q                  LOCATION OF IMBEDDED BLANK IND
                                   5354+*
                       0087        5355+C4BSPC EQU   @UCB                       MOVED TO C4BLNK TO ALLOW BLANKS
                                   5356+*
                       1CF9        5357+C4BNMC EQU   C4B100+@Q                  LOCATION OF CONVERSION COUNT
                                   5358+*
                       0080        5359+C4BNOP EQU   @NOP                       CHANGED IF IMBEDDED BLANK OK
                       1D52        5360+C4END  EQU   *                          DEFINE END OF CODE
                                   5361+***                        END OF C4BIN2                      ***
                                   5362 ****   MOD1  VER3      LINE PRINTER REQM'T   ****
      1D52 38 01 03E4              5363 ZUT900 TBN   $LPRP3,@INDEX              TEST DUMMY PRINT POSITION   1-3
      1D56 C0 90 1D60              5364        BF    ZUT950                     BRANCH NO
      1D5A 0C 00 03C2 03E5         5365        MVC   $PRPOS(1),$LPROS           RESTORE TRUE PRINT POSITION
      1D60 3C 00 03C1              5366 ZUT950 MVI   $LMRGN,0                   SET LEFT MARGIN TO ZERO
      1D64 0F 01 03E4 03E4         5367        SLC   $LPRP3(2),$LPRP3           RESET LINE PTR. BUFFER&FLAGS 1-3
      1D6A C0 87 0C15              5368        B     ZUT012                     GO DO CARRAGE RETURN
                                   5369        PRINT ON
                       FFFF        5370        END
```

TOTAL STATEMENTS IN ERROR IN THIS ASSEMBLY =   0

SYMBOL	LEN	VALUE	DEFN	REFERENCES											
$$$$$$	001	0C00	3008												
$$$$$L	023	0D30	3154												
$$$CMD	001	0020	1746												
$$$DAT	001	0040	1745												
$$$EPL	001	0091	1742												
$$$ERN	001	0080	1796												
$$$FUN	001	0010	1747												
$$$NLN	001	00A0	1792												
$$$STD	001	0081	1741												
$$BNLN	001	0605	1722	1724											
$$CDBS	001	08C0	1772												
$$CDND	001	0666	1731												
$$CDRD	001	0890	1770	1772											
$$CKEY	001	0603	1720												
$$CKFF	001	0B3D	1752												
$$COFF	001	0B44	1751												
$$CSNS	001	209C	1781												
$$DATB	001	0BBF	1753												
$$EOSA	001	0AFE	1750												
$$ERSK	001	1C00	1791												
$$FITS	001	1D00	1799												
$$FLIB	001	06FF	1798												
$$ILEN	001	0601	1716	1718	1722										
$$ILHD	001	0600	1714	1716											
$$INLN	001	0607	1729	1731	1733	3168	4691	4790							
$$INND	001	06FA	1733												
$$KBDT	001	09E1	1740	1744											
$$KBSN	001	09E2	1744	1749											
$$KLD1	001	0600	1804												
$$KLD2	001	0700	1806												
$$KLD3	001	0C00	1808												
$$LPOS	001	09EB	1749												
$$PCNT	001	07E9	1765												
$$PLYN	001	2004	1779												
$$PRES	001	0890	1738	1740	1750	1751	1752	1753	1770	3169	4682				
$$PRFL	001	2143	1783												
$$PRNT	001	0707	1759	1760	1764	1765	3035	3049	3054	3059	3106	3110	3125	3135	3233
				3361	3370	3846	3869	4253	4258	4323	4327	4339	4345	4351	4674
				4749	4785	4922	4943	4948	4959	4963	4973	5081	5118	5132	5154
				5158											
$$PRTN	001	0782	1760												
$$PSIO	001	07CE	1764												
$$PYCD	001	2200	1785												
$$PYMP	001	2000	1777	1779	1781	1783	1785								
$$SLIB	001	1C00	1794												
$$TPCD	001	0606	1724	1729											
$$UPAR	001	0602	1718	1720											
$$WSPB	001	1E00	1797												
$$XIND	001	06FF	1795	1798											
$$ZERO	001	0000	0223	0224	0226	0227	0228	0232	1777						
$ABORT	001	0010	0336												
$BASIC	001	0080	0394												
$BIGCD	001	0080	0470												
$BLDPL	001	0579	0603	0605											
$BLNOE	001	0569	0593												
$BLOAD	001	0522	0584	0586	0589	0602	0603								

```
$BLRTN  001  0550 0592    0593
$BRSAV  001  03C5 0281    0282
$BSADR  001  0587 0608    0610   3346   3383
$BUFPT  001  03E3 0489    0490
$CABLD  001  04B4 0562    0563
$CAERK  001  0469 0539    0542
$CAERR  001  03CD 0287    0289   3713*  5279*
$CAIPL  001  049D 0558    0560   3239
$CALLI  001  0008 0479
$CARDI  001  0001 0250
$CARPL  001  04A1 0560    0562
$CIENT  001  0483 0549    0550
$CIEXT  001  0480 0548    0549
$CIMSK  001  0476 0545    0548   3031*
$CISUS  001  0496 0553    0558   3254   3255*
$CLBFR  001  0010 0437
$CMDKY  001  0008 0349    3044
$CMODE  001  0002 0399
$CONFG  001  03DD 0462    0472
$CRPOS  001  03E2 0488    0489
$CRTAD  001  044D 0527    0528
$CRTAV  001  0002 0343
$CRTDN  001  0002 0367
$CRTIN  001  03D3 0364    0371
$CRTNO  001  0004 0346
$CRTPU  001  0004 0368
$CRTSP  001  0008 0369
$CRTUP  001  0001 0366
$CRUSH  001  0080 0475
$CSDPL  001  050E 0574    0575   3995   4028
$C0001  001  0464 0531    0537
$DATE   001  043A 0512    0513
$DBGUF  001  03E0 0474    0483
$DBLOK  001  0001 0424
$DFDET  001  03E8 0495    0496
$DISKN  001  0025 0226    3350   3387   3588   4041   4757   4861   4869   4952   4954   5058   5060   5064
                          5066   5090   5092   5096   5098
$DKERR  001  0008 0405
$DKSIZ  001  03D7 0449    0457   0498
$DK100  001  0001 0451
$DK200  001  0002 0452
$DK400  001  0004 0453
$DK600  001  0008 0454
$DK800  001  0010 0455
$DPLSV  001  0449 0523    0525
$DTNMB  001  0040 0270
$DTRDR  001  0040 0358
$ENDNU  001  0600 0617    1714   1738   1759   1795   1804   1806   1808   1819
$ERDPL  001  046F 0542    0544
$ERFIL  001  0040 0297
$ERHRD  001  0004 0429
$ERKEY  001  0080 0301
$ERLOG  001  0345 0231
$ERMAD  001  0472 0544    0545
$ERPND  001  0004 0402
$ERRCT  001  03CF 0303
```

```
$ERRPG  001  03CE 0291
$ERSFL  001  0035 0296
$ERSTK  001  0030 0294
$ER050  001  0363 0232
$ER1N2  001  0050 0299
$EXADR  001  0517 0577   0579
$EXCMD  001  0001 0331   3143
$EXFTR  001  043B 0513   0518
$FCIND  001  0010 0409
$FDIND  001  0040 0416
$FEARR  001  0004 0224
$FEMAP  001  0588 0610   0611
$FILIB  001  03DA 0460   0461
$FITIN  001  0010 0385
$FUIND  001  0020 0414
$GUFIO  001  0583 0607   0608
$GUFIR  001  0008 0259
$HISTE  001  042E 0510   0511
$HIST1  001  0435 0511   0512
$HRDER  001  0020 0355
$INDR1  001  03D4 0371   0397
$INDR2  001  03D5 0397   0422
$INDR3  001  03D6 0422   0449
$INLNO  001  03CF 0289   0291  0303  0310
$INRPT  001  0020 0267
$IOIND  001  03D2 0338   0364  3044*
$IOPGS  001  0010 0478
$IOYES  001  0002 0253
$IPLDV  001  05FF 0614   0617
$IRKEY  001  0020 0477
$KEYBD  001  03E1 0483   0488
$KEYCD  001  03C3 0247   0281  3170  4683
$KEYDT  001  0040 0391
$KE090  001  00DE 0227
$KE130  001  01D5 0228
$KYBSY  001  0010 0264   3170  4683
$LDRTN  001  0571 0602
$LEVEL  001  03DF 0472   0474
$LIST   001  0002 0426
$LMRGN  001  03C1 0242   0244  3032  3140* 5366*
$LNPTR  001  0080 0361
$LOADB  001  054A 0586
$LOADR  001  051A 0579   0582
$LPRIO  001  03EA 0496
$LPROS  001  03E5 0491   0493  5365
$LPRP3  001  03E4 0490   0491  5363  5367  5367*
$MOUNT  001  0020 0440
$MPDWN  001  0001 0340
$NEXTB  001  03E6 0493   0494
$NEXTL  001  03E7 0494   0495
$NOENB  001  0008 0432
$NOLST  001  0004 0256
$NUCBS  001  03C0 0239   0240
$NWRKF  001  0080 0445
$NWRKR  001  0040 0442
$PASWD  001  042D 0509   0510
```

```
$PAUSD  001  04BA 0563   0565
$PAUSE  001  0002 0333   3143
$PGMDT  001  0020 0388
$PGMST  001  0010 0352
$PKERT  001  0419 0507   0509
$PLST1  001  0454 0528   0529
$PLST2  001  045B 0529   0530
$PLST3  001  0462 0530   0531
$PRDEV  001  044B 0525   0527   3042   3043*  3142*
$PRESN  001  0002 0376
$PROCI  001  0001 0373
$PRPOS  001  03C2 0244   0247   3039*  5365*
$PSDBR  001  04FA 0568   4136
$PSDXR  001  04F2 0567   0568   4135
$PSTEP  001  0004 0334
$PSTMT  001  0008 0335
$PTCH1  001  03F5 0498   0502
$READY  001  0080 0418
$REORD  001  0040 0476
$RLOAD  001  051E 0582   0584   3314
$RMRGN  001  03C0 0240   0242   3040   3041*  3141*
$RSTR   001  04D6 0565   0567   0569   0574   3147
$RUNIT  001  0001 0312
$SFAID  001  050D 0570
$SPRNT  001  0465 0537   0539
$SRTRN  001  04FE 0569   0570   3123*
$STEPT  001  0002 0313
$SWPCR  001  0511 0575   0577   3997
$TABLN  001  03CB 0284   0287
$TFLOW  001  0008 0319
$TRACE  001  0004 0314
$TRALL  001  0010 0320
$TROVR  001  054E 0589   0592   3117   3119*  3122*
$TRUNK  001  0080 0272
$TRVAR  001  0020 0321
$UNMSK  001  048D 0550   0553
$USRDR  001  03DC 0461   0462
$VMDEF  001  0080 0325
$VOLF1  001  03FE 0504   0505
$VOLF2  001  040E 0506
$VOLID  001  03F6 0502   0503   0507
$VOLR1  001  03F6 0503   0504
$VOLR2  001  0406 0505   0506
$WAITF  001  057F 0605   0607   3060   3111   3351   3388   4042   4259   4328   4758   4862   4870   4955
                         4964   5061   5067   5093   5099   5133   5159
$WFDEF  001  0040 0519
$WFLOK  001  0008 0382
$WFNME  001  0443 0518   0523
$WSIND  001  0004 0379
$XIND1  001  03D0 0310   0329
$XIND2  001  03D1 0329   0338   3143*
$XIND3  001  03D8 0457   0460
$XPREC  001  0040 0322
$XRSAV  001  03C7 0282   0284
$ZTRAD  001  05A2 0611
$12K    001  0004 0466
```

```
SYMBOL  LEN VALUE DEFN   REFERENCES                            VER 15, MOD 00  13/01/22  PAGE  66

$16CKY  001  0008 0468
$16K    001  0002 0465
$22IMP  001  0001 0463
#$$#BL  001  0000 0945
#$$#CK  001  0000 1073
#$$#CN  001  0000 1041
#$$#CO  001  0000 0833
#$$#CS  001  0000 0893
#$$#DR  001  0000 0637
#$$#ER  001  0000 0837
#$$#FS  001  0000 0933
#$$#IN  001  0000 1077
#$$#PW  001  0000 1081
#$$#RS  001  0000 0913
#$$#SA  001  0000 0901
#$$#SS  001  0000 0897
#$$#VU  001  0600 0857
#$$#0T  001  0700 0629
#$$#1T  001  0000 0633
#$$BCO  001  0600 0645
#$$BOV  001  0800 0917
#$$DPR  001  0700 0653
#$$DRE  001  0889 0669
#$$DSP  001  2800 0689
#$$ECM  001  0C00 0949
#$$EFK  001  0C00 0969
#$$ERR  001  0C00 0941
#$$EXM  001  0C00 0829
#$$FIL  001  0E00 0909
#$$FIS  001  0E00 0905
#$$FML  001  0200 1037
#$$FMS  001  0200 0877
#$$GRA  001  0889 0801
#$$GUF  001  0C00 0937
#$$INL  001  0600 1017
#$$INS  001  0600 0641
#$$KAL  001  0C00 0805
#$$KCA  001  0C00 1021
#$$KCH  001  0C00 0773
#$$KCN  001  0C00 0889
#$$KCT  001  0C00 0741
#$$KDE  001  0C00 0737
#$$KDI  001  0D00 0817
#$$KDN  001  0C00 0725
#$$KDO  001  0E00 0821
#$$KED  001  0C00 0661
#$$KEN  001  0C00 0665
#$$KEX  001  0C00 0685
#$$KGO  001  0C00 0657
#$$KHE  001  0C00 0841
#$$KKE  001  0C00 1069
#$$KLI  001  0C00 0745
#$$KLL  001  0920 1045
#$$KLO  001  0C00 0749
#$$KME  001  0D00 0729
#$$KMO  001  0C00 0673
```

```
#$$KNA   001   0C00  0785
#$$KOV   001   0E00  0705
#$$KPA   001   0C00  0681
#$$KPO   001   0C00  0769
#$$KPR   001   0C00  0793
#$$KRE   001   0C00  0713
#$$KRL   001   0700  0809
#$$KRM   001   0C00  0677
#$$KRN   001   0700  0697
#$$KRO   001   0D00  0701
#$$KRS   001   0C00  1025
#$$KRU   001   0C00  0721
#$$KRV   001   0800  0813
#$$KSA   001   0C00  0757
#$$KSE   001   0E00  0797
#$$KSO   001   0C20  0849
#$$KSS   001   0C00  0781
#$$KSV   001   0980  0777
#$$KSY   001   0C00  0789
#$$KWI   001   0C00  0717
#$$KWR   001   0C00  0709
#$$LOA   001   0600  0649
#$$MIP   001   0C00  0845
#$$SDS   001   0C00  0957
#$$SFF   001   0E00  0961
#$$SFL   001   0F00  0953
#$$SFO   001   1500  0925
#$$SFS   001   0C00  0921
#$$SPA   001   0C00  0761
#$$SPO   001   0806  0765
#$$SPS   001   0C00  0753
#$$STR   001   1600  0929
#$$TDC   001   1000  0733
#$$TSY   001   1000  0693
#$$TVK   001   0FC0  0869
#$$UAL   001   0C00  0885
#$$UAT   001   0900  0981
#$$UCD   001   0900  0989
#$$UCN   001   0C00  0973
#$$UCP   001   0700  0977
#$$UDE   001   0C00  0993
#$$UDI   001   0C00  0997
#$$UEX   001   0C00  0881
#$$UIN   001   0C00  0985
#$$UPA   001   0C00  0965
#$$UPO   001   0C00  1033
#$$UPT   001   0C00  1029
#$$VCR   001   2000  0825
#$$VLO   001   0600  0861
#$$VOD   001   0600  0865
#$$VVM   001   0000  0873
#$$VXI   001   0600  0853
#$$ZDU   001   1100  1005
#$$ZLB   001   1100  1049   3326
#$$ZLO   001   1100  1009
#$$ZLV   001   0F00  1065
```

```
SYMBOL   LEN VALUE DEFN   REFERENCES                              VER 15, MOD 00  13/01/22  PAGE  68

#$$ZL1   001  0F00 1053
#$$ZL2   001  0F00 1057
#$$ZL3   001  0C00 1061
#$$ZTR   001  1000 1001
#$$ZUT   001  0C00 1013   3007
#$#BLN   001  18D4 0944
#$#CKT   001  2118 1072
#$#CNF   001  2000 1040
#$#COR   001  0800 0832
#$#CSA   001  1000 0892
#$#DRT   001  0000 0636
#$#ERM   001  0928 0836
#$#FSP   001  1880 0932
#$#INV   001  212C 1076
#$#PWR   001  2300 1080
#$#RSP   001  1780 0912
#$#SAV   001  1180 0900
#$#SSA   001  1128 0896
#$#VUF   001  0B08 0856
#$#0TR   001  0000 0628
#$#1TR   001  0080 0632
#$@#BL   001  0001 0946
#$@#CK   001  0004 1074
#$@#CN   001  0001 1042
#$@#CO   001  003A 0834
#$@#CS   001  003A 0894
#$@#DR   001  0008 0638
#$@#ER   001  0032 0838
#$@#FS   001  0030 0934
#$@#IN   001  003A 1078
#$@#PW   001  00C0 1082
#$@#RS   001  0030 0914
#$@#SA   001  0108 0902
#$@#SS   001  0001 0898
#$@#VU   001  0002 0858
#$@#0T   001  0018 0630
#$@#1T   001  0018 0634
#$@BCO   001  0018 0646
#$@BOV   001  0018 0918
#$@DPR   001  0005 0654
#$@DRE   001  0001 0670
#$@DSP   001  0004 0690
#$@ECM   001  0006 0950
#$@EFK   001  0002 0970
#$@ERR   001  0003 0942
#$@EXM   001  0003 0830
#$@FIL   001  0009 0910
#$@FIS   001  0009 0906
#$@FML   001  0052 1038
#$@FMS   001  0052 0878
#$@GRA   001  0003 0802
#$@GUF   001  0010 0938
#$@INL   001  0010 1018
#$@INS   001  0010 0642
#$@KAL   001  000F 0806
#$@KCA   001  000C 1022
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES	VER 15, MOD 00	13/01/22	PAGE 69
#$@KCH	001	000C	0774				
#$@KCN	001	0010	0890				
#$@KCT	001	0009	0742				
#$@KDE	001	0010	0738				
#$@KDI	001	0005	0818				
#$@KDN	001	0010	0726				
#$@KDO	001	000C	0822				
#$@KED	001	000E	0662				
#$@KEN	001	0006	0666				
#$@KEX	001	0003	0686				
#$@KGO	001	0002	0658				
#$@KHE	001	000C	0842				
#$@KKE	001	0006	1070				
#$@KLI	001	0011	0746				
#$@KLL	001	0001	1046				
#$@KLO	001	0008	0750				
#$@KME	001	0003	0730				
#$@KMO	001	0004	0674				
#$@KNA	001	0008	0786				
#$@KOV	001	0009	0706				
#$@KPA	001	0005	0682				
#$@KPO	001	000D	0770				
#$@KPR	001	0009	0794				
#$@KRE	001	0002	0714				
#$@KRL	001	0004	0810				
#$@KRM	001	0003	0678				
#$@KRN	001	0003	0698				
#$@KRO	001	000A	0702				
#$@KRS	001	000A	1026				
#$@KRU	001	0003	0722				
#$@KRV	001	000D	0814				
#$@KSA	001	0011	0758				
#$@KSE	001	0004	0798				
#$@KSO	001	0005	0850				
#$@KSS	001	000B	0782				
#$@KSV	001	0002	0778				
#$@KSY	001	000F	0790				
#$@KWI	001	0002	0718				
#$@KWR	001	0002	0710				
#$@LOA	001	0013	0650				
#$@MIP	001	000D	0846				
#$@SDS	001	0004	0958				
#$@SFF	001	0008	0962				
#$@SFL	001	0005	0954				
#$@SFO	001	0003	0926				
#$@SFS	001	0011	0922				
#$@SPA	001	0004	0762				
#$@SPO	001	0003	0766				
#$@SPS	001	0001	0754				
#$@STR	001	0002	0930				
#$@TDC	001	0003	0734				
#$@TSY	001	0003	0694				
#$@TVK	001	0001	0870				
#$@UAL	001	0011	0886				
#$@UAT	001	000C	0982				
#$@UCD	001	000B	0990				

```
                        CROSS REFERENCE

SYMBOL  LEN VALUE DEFN  REFERENCES                        VER 15, MOD 00  13/01/22  PAGE  70

#$@UCN  001  0009 0974
#$@UCP  001  000F 0978
#$@UDE  001  000E 0994
#$@UDI  001  0008 0998
#$@UEX  001  000E 0882
#$@UIN  001  000F 0986
#$@UPA  001  0004 0966
#$@UPO  001  0005 1034
#$@UPT  001  0012 1030
#$@VCR  001  0008 0826
#$@VLO  001  0002 0862
#$@VOD  001  0016 0866
#$@VVM  001  0030 0874
#$@VXI  001  0002 0854
#$@ZDU  001  0008 1006
#$@ZLB  001  0002 1050  3325
#$@ZLO  001  000C 1010
#$@ZLV  001  0006 1066
#$@ZL1  001  0007 1054
#$@ZL2  001  000D 1058
#$@ZL3  001  000A 1062
#$@ZTR  001  0001 1002
#$@ZUT  001  0014 1014
#$BCOM  001  0080 0644
#$BOLV  001  1780 0916
#$DPRI  001  014C 0652
#$DREA  001  0200 0668
#$DSPL  001  0240 0688
#$ECMA  001  1900 0948
#$EFKE  001  1990 0968
#$ERRP  001  18C0 0940
#$EXMS  001  07D4 0828
#$FILN  001  1724 0908
#$FIST  001  1700 0904
#$FMLN  001  1E00 1036
#$FMST  001  0D00 0876
#$GRAP  001  0690 0800
#$GUFU  001  1880 0936
#$INLN  001  1C84 1016
#$INST  001  0020 0640
#$KALL  001  06A4 0804
#$KCAL  001  1CC4 1020
#$KCHA  001  053C 0772
#$KCND  001  0F80 0888
#$KCTL  001  03BC 0740
#$KDEL  001  035C 0736
#$KDIS  001  0744 0816
#$KDNT  001  0300 0724
#$KDOV  001  0780 0820
#$KEDI  001  0188 0660
#$KENA  001  01C4 0664
#$KEXT  001  0234 0684
#$KGOS  001  0180 0656
#$KHEL  001  0A30 0840
#$KKEY  001  2100 1068
#$KLIS  001  0400 0744
```

```
#$KLLA   001   2004 1044
#$KLOG   001   0444 0748
#$KMER   001   030C 0728
#$KMOU   001   0204 0672
#$KNAM   001   05C0 0784
#$KOVM   001   0290 0704
#$KPAS   001   0220 0680
#$KPOO   001   0508 0768
#$KPRT   001   063C 0792
#$KREA   001   02BC 0712
#$KRLA   001   0700 0808
#$KRMO   001   0214 0676
#$KRNU   001   0280 0696
#$KROV   001   028C 0700
#$KRSU   001   1D24 1024
#$KRUN   001   02CC 0720
#$KRVL   001   0710 0812
#$KSAV   001   0488 0756
#$KSET   001   0680 0796
#$KSOV   001   0AC8 0848
#$KSSP   001   0594 0780
#$KSVL   001   058C 0776
#$KSYM   001   0600 0788
#$KWID   001   02C4 0716
#$KWRI   001   02B4 0708
#$LOAD   001   0100 0648
#$MIPP   001   0A80 0844
#$SDSY   001   192C 0956
#$SFFI   001   193C 0960
#$SFLO   001   1918 0952
#$SFOV   001   1844 0924
#$SFSY   001   1800 0920
#$SPAC   001   04CC 0760
#$SPOV   001   04DC 0764
#$SPSY   001   0484 0752
#$STRO   001   1850 0928
#$TDCK   001   0350 0732
#$TSYK   001   0250 0692
#$TVKB   001   0BAC 0868
#$UALL   001   0F00 0884
#$UATR   001   1A38 0980
#$UCDI   001   1AD8 0988
#$UCNF   001   19B8 0972
#$UCPL   001   19DC 0976
#$UDEL   001   1B24 0992
#$UDIS   001   1B5C 0996
#$UEXL   001   0EA8 0880
#$UINI   001   1A88 0984
#$UPAC   001   1980 0964
#$UPOV   001   1D24 1032
#$UPTF   001   1D5C 1028
#$VCRT   001   07B4 0824
#$VLOA   001   0B80 0860
#$VODK   001   0B88 0864
#$VVMR   001   0C00 0872
#$VXIT   001   0B00 0852
```

```
#$ZDUM  001  1BA4 1004   3430
#$ZLBM  001  2008 1048   3324
#$ZLOA  001  1BC4 1008
#$ZLVR  001  20B0 1064
#$ZL1M  001  2010 1052
#$ZL2M  001  2030 1056
#$ZL3M  001  2088 1060
#$ZTRA  001  1B9C 1000
#$ZUTM  001  1C14 1012   3429
#@#BAD  001  0455 2737
#@#IO1  001  0459 2745
#@#IO2  001  045D 2746
#@#TAT  001  0941 2773
#@#TBA  001  09A1 2777
#@#TFS  001  0941 2771
#@#TSY  001  0941 2775
#@#VFP  001  0700 2763
#@#VLP  001  093D 2766
#@#WDB  001  050C 2758
#@#WFT  001  0500 2756
#@@#BA  001  0001 2738
#@@#IO  001  0001 2750
#@@#SC  001  0002 2747
#@@#TA  001  0010 2774
#@@#TB  001  0010 2778
#@@#TS  001  0005 2776
#@@#TW  001  0020 2772
#@@#VM  001  0100 2767
#@@#WD  001  00BD 2759
#@@#WF  001  0003 2757
#@@#04  001  0004 2749
#@@#08  001  0008 2748
#@@BOV  001  0018 2726
#@@ECM  001  0006 2740
#@@ERR  001  0003 2734
#@@GUF  001  0010 2730
#@@LDS  001  0002 2736
#@@SDS  001  0004 2732
#@@SFF  001  0008 2744
#@@SFL  001  0005 2742
#@@SFO  001  0005 2752
#@@SFS  001  0011 2728
#@@VSF  001  0010 2780
#@@VSL  001  000F 2781
#@@VTR  001  0001 2765
#@BOVL  001  0400 2725
#@ECMA  001  0481 2739
#@ERRP  001  0441 2733
#@GUFU  001  0401 2729
#@LDSV  001  044D 2735
#@SDSY  001  04AD 2731
#@SFFI  001  04BD 2743
#@SFLO  001  0499 2741
#@SFOV  001  04C4 2751
#@SFSY  001  0480 2727
#@VSFI  001  09A1 2779
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES	
#@VTRL	001	0708	2764		
#@WAF1	001	0401	2724		
#@WAR1	001	0400	2723		
#ZUTM	001	0C07	3011		
#ZUTMO	001	0000	0001		
@@E001	001	0000	1619	1621	
@@E003	001	0001	1621	1623	
@@E004	001	0002	1623	1625	
@@E005	001	0003	1625	1627	
@@E006	001	0004	1627	1629	
@@E007	001	0005	1629	1631	
@@E008	001	0006	1631	1633	
@@E009	001	0007	1633	1635	
@@E010	001	0008	1635	1637	
@@E011	001	0009	1637	1639	
@@E012	001	000A	1639	1641	
@@E013	001	000B	1641	1643	
@@E014	001	000C	1643	1645	
@@E015	001	000D	1645	1647	
@@E016	001	000E	1647	1649	
@@E017	001	000F	1649	1651	
@@E018	001	0010	1651	1653	
@@E019	001	0011	1653	1655	
@@E020	001	0012	1655	1657	
@@E021	001	0013	1657	1659	
@@E023	001	0014	1659	1661	
@@E024	001	0015	1661	1663	
@@E025	001	0016	1663	1665	
@@E026	001	0017	1665	1667	
@@E027	001	0018	1667	1669	
@@E028	001	0019	1669	1671	
@@E029	001	001A	1671	1673	
@@E030	001	001B	1673	1675	
@@E031	001	001C	1675	1677	
@@E032	001	001D	1677	1679	
@@E035	001	001E	1679	1681	
@@E036	001	001F	1681	1683	
@@E037	001	0020	1683	1685	
@@E038	001	0021	1685	1687	
@@E039	001	0022	1687	1689	
@@E040	001	0023	1689	1691	
@@E041	001	0024	1691	1693	
@@E042	001	0025	1693	1695	
@@E043	001	0026	1695	1697	
@@E044	001	0027	1697	1699	
@@E045	001	0028	1699	1701	
@@E046	001	0029	1701	1703	
@@E060	001	002A	1703	1705	
@@E080	001	002B	1705		
@@E100	001	0000	1091	1093	
@@E101	001	0001	1093	1095	
@@E102	001	0002	1095	1097	
@@E103	001	0003	1097	1099	
@@E110	001	0004	1099	1101	3713
@@E112	001	0005	1101	1103	
@@E113	001	0006	1103	1105	

```
@@E114   001  0007 1105   1107
@@E115   001  0008 1107   1109
@@E116   001  0009 1109   1111
@@E117   001  000A 1111   1113
@@E120   001  000B 1113   1115
@@E122   001  000C 1115   1117  5279
@@E123   001  000D 1117   1119
@@E124   001  000E 1119   1121
@@E129   001  000F 1121   1123
@@E130   001  0010 1123   1125
@@E131   001  0011 1125   1127
@@E133   001  0012 1127   1129
@@E134   001  0013 1129   1131
@@E135   001  0014 1131   1133
@@E136   001  0015 1133   1135
@@E137   001  0016 1135   1137
@@E138   001  0017 1137   1139
@@E139   001  0018 1139   1141
@@E142   001  0019 1141   1143
@@E143   001  001A 1143   1145
@@E150   001  001B 1145   1147
@@E151   001  001C 1147   1149
@@E160   001  001D 1149   1151
@@E162   001  001E 1151   1153
@@E163   001  001F 1153   1155
@@E164   001  0020 1155   1157
@@E200   001  0021 1157   1159
@@E205   001  0022 1159   1161
@@E210   001  0023 1161   1163
@@E211   001  0024 1163   1165
@@E212   001  0025 1165   1167
@@E213   001  0026 1167   1169
@@E215   001  0027 1169   1171
@@E216   001  0028 1171   1173
@@E217   001  0029 1173   1175
@@E220   001  002A 1175   1177
@@E221   001  002B 1177   1179
@@E222   001  002C 1179   1181
@@E223   001  002D 1181   1183
@@E225   001  002E 1183   1185
@@E226   001  002F 1185   1187
@@E227   001  0030 1187   1189
@@E228   001  0031 1189   1191
@@E229   001  0032 1191   1193
@@E230   001  0033 1193   1195
@@E232   001  0034 1195   1197
@@E234   001  0035 1197   1199
@@E237   001  0036 1199   1201
@@E240   001  0037 1201   1203
@@E241   001  0038 1203   1205
@@E242   001  0039 1205   1207
@@E248   001  003A 1207   1209
@@E249   001  003B 1209   1211
@@E250   001  003C 1211   1213
@@E251   001  003D 1213   1215
@@E252   001  003E 1215   1217
```

```
                               CROSS REFERENCE

SYMBOL   LEN VALUE DEFN   REFERENCES                      VER 15, MOD 00  13/01/22  PAGE  75

@@E253   001  003F 1217   1219
@@E254   001  0040 1219   1221
@@E255   001  0041 1221   1223
@@E256   001  0042 1223   1225
@@E300   001  0043 1225   1227
@@E301   001  0044 1227   1229
@@E302   001  0045 1229   1231
@@E303   001  0046 1231   1233
@@E304   001  0047 1233   1235
@@E305   001  0048 1235   1237
@@E308   001  0049 1237   1239
@@E310   001  004A 1239   1241
@@E315   001  004B 1241   1243
@@E316   001  004C 1243   1245
@@E320   001  004D 1245   1247
@@E325   001  004E 1247   1249
@@E330   001  004F 1249   1251
@@E335   001  0050 1251   1253
@@E338   001  0051 1253   1255
@@E340   001  0052 1255   1257
@@E350   001  0053 1257   1259
@@E351   001  0054 1259   1261
@@E352   001  0055 1261   1263
@@E360   001  0056 1263   1265
@@E361   001  0057 1265   1267
@@E362   001  0058 1267   1269
@@E371   001  0059 1269   1271
@@E380   001  005A 1271   1273
@@E390   001  005B 1273   1275
@@E400   001  005C 1275   1277
@@E410   001  005D 1277   1279
@@E415   001  005E 1279   1281
@@E417   001  005F 1281   1283
@@E420   001  0060 1283   1285
@@E430   001  0061 1285   1287
@@E432   001  0062 1287   1289
@@E433   001  0063 1289   1291
@@E450   001  0064 1291   1293
@@E451   001  0065 1293   1295
@@E460   001  0066 1295   1297
@@E461   001  0067 1297   1299
@@E464   001  0068 1299   1301
@@E465   001  0069 1301   1303
@@E466   001  006A 1303   1305
@@E467   001  006B 1305   1307
@@E469   001  006C 1307   1309
@@E470   001  006D 1309   1311
@@E471   001  006E 1311   1313
@@E473   001  006F 1313   1315
@@E474   001  0070 1315   1317
@@E475   001  0071 1317   1319
@@E476   001  0072 1319   1321
@@E477   001  0073 1321   1323
@@E478   001  0074 1323   1325
@@E479   001  0075 1325   1327
@@E480   001  0076 1327   1329
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES
@@E481	001	0077	1329	1331
@@E482	001	0078	1331	1333
@@E483	001	0079	1333	1335
@@E484	001	007A	1335	1337
@@E485	001	007B	1337	1339
@@E486	001	007C	1339	1341
@@E487	001	007D	1341	1343
@@E488	001	007E	1343	1345
@@E489	001	007F	1345	1347
@@E490	001	0080	1347	1349
@@E491	001	0081	1349	1351
@@E492	001	0082	1351	1353
@@E493	001	0083	1353	1355
@@E494	001	0084	1355	1357
@@E495	001	0085	1357	1359
@@E496	001	0086	1359	1361
@@E497	001	0087	1361	1363
@@E498	001	0088	1363	1365
@@E500	001	0089	1365	1367
@@E501	001	008A	1367	1369
@@E530	001	008B	1369	1371
@@E531	001	008C	1371	1373
@@E535	001	008D	1373	1375
@@E540	001	008E	1375	1377
@@E541	001	008F	1377	1379
@@E542	001	0090	1379	1381
@@E543	001	0091	1381	1383
@@E544	001	0092	1383	1385
@@E545	001	0093	1385	1387
@@E546	001	0094	1387	1389
@@E547	001	0095	1389	1391
@@E548	001	FFFF	1595	
@@E549	001	0096	1391	1393
@@E550	001	0097	1393	1395
@@E551	001	0098	1395	1397
@@E552	001	0099	1397	1399
@@E553	001	009A	1399	1401
@@E554	001	009B	1401	1403
@@E555	001	009C	1403	1405
@@E556	001	009D	1405	1407
@@E558	001	009E	1407	1409
@@E570	001	009F	1409	1411
@@E571	001	00A0	1411	1413
@@E572	001	00A1	1413	1415
@@E573	001	00A2	1415	1417
@@E574	001	00A3	1417	1419
@@E575	001	FFFF	1597	
@@E578	001	00A4	1419	1421
@@E579	001	FFFF	1599	
@@E580	001	FFFF	1601	
@@E585	001	00A5	1421	1423
@@E595	001	FFFF	1603	
@@E597	001	FFFF	1605	
@@E598	001	FFFF	1607	
@@E600	001	00A6	1423	1425
@@E601	001	00A7	1425	1427

SYMBOL	LEN	VALUE	DEFN	REFERENCES
@@E602	001	00A8	1427	1429
@@E603	001	00A9	1429	1431
@@E604	001	00AA	1431	1433
@@E606	001	00AB	1433	1435
@@E607	001	00AC	1435	1437
@@E608	001	00AD	1437	1439
@@E609	001	00AE	1439	1441
@@E610	001	00AF	1441	1443
@@E611	001	00B0	1443	1445
@@E612	001	00B1	1445	1447
@@E613	001	00B2	1447	1449
@@E614	001	00B3	1449	1451
@@E700	001	00B4	1451	1453
@@E701	001	00B5	1453	1455
@@E710	001	00B6	1455	1457
@@E712	001	00B7	1457	1459
@@E713	001	00B8	1459	1461
@@E714	001	00B9	1461	1463
@@E715	001	00BA	1463	1465
@@E716	001	00BB	1465	1467
@@E717	001	00BC	1467	1469
@@E718	001	00BD	1469	1471
@@E720	001	00BE	1471	1473
@@E721	001	00BF	1473	1475
@@E723	001	00C0	1475	1477
@@E724	001	00C1	1477	1479
@@E725	001	00C2	1479	1481
@@E726	001	00C3	1481	1483
@@E727	001	00C4	1483	1485
@@E728	001	00C5	1485	1487
@@E729	001	00C6	1487	1489
@@E730	001	00C7	1489	1491
@@E732	001	00C8	1491	1493
@@E752	001	00C9	1493	1495
@@E753	001	00CA	1495	1497
@@E754	001	00CB	1497	1499
@@E755	001	00CC	1499	1501
@@E756	001	00CD	1501	1503
@@E757	001	00CE	1503	1505
@@E758	001	00CF	1505	1507
@@E759	001	00D0	1507	1509
@@E760	001	00D1	1509	1511
@@E761	001	00D2	1511	1513
@@E762	001	00D3	1513	1515
@@E763	001	00D4	1515	1517
@@E764	001	00D5	1517	1519
@@E765	001	00D6	1519	1521
@@E766	001	00D7	1521	1523
@@E767	001	00D8	1523	1525
@@E768	001	00D9	1525	1527
@@E769	001	00DA	1527	1529
@@E770	001	00DB	1529	1531
@@E771	001	00DC	1531	1533
@@E772	001	00DD	1533	1535
@@E773	001	00DE	1535	1537
@@E774	001	00DF	1537	1539

SYMBOL	LEN	VALUE	DEFN	REFERENCES											
@@E775	001	00E0	1539	1541											
@@E776	001	00E1	1541	1543											
@@E777	001	00E2	1543	1545											
@@E778	001	00E3	1545	1547											
@@E779	001	00E4	1547	1549											
@@E780	001	00E5	1549	1551											
@@E781	001	00E6	1551	1553											
@@E782	001	00E7	1553	1555											
@@E783	001	00E8	1555	1557											
@@E784	001	00E9	1557	1559											
@@E785	001	00EA	1559	1561											
@@E786	001	00EB	1561	1563											
@@E790	001	00EC	1563	1565											
@@E791	001	00ED	1565	1567											
@@E792	001	00EE	1567	1569											
@@E793	001	00EF	1569	1571											
@@E794	001	00F0	1571	1573											
@@E795	001	00F1	1573	1575											
@@E796	001	00F2	1575	1577											
@@E797	001	00F3	1577	1579											
@@E798	001	00F4	1579	1581											
@@E800	001	FFFF	1609												
@@E801	001	FFFF	1611												
@@E802	001	FFFF	1613												
@@E803	001	FFFF	1615												
@@E804	001	FFFF	1617												
@@E900	001	00F5	1581	1583											
@@E901	001	00F6	1583	1585											
@@E902	001	00F7	1585	1587											
@@E903	001	00F8	1587	1589											
@@E905	001	00F9	1589	1591											
@@E906	001	00FA	1591	1593											
@@E910	001	00FB	1593												
@ARR	001	0008	0016	3165	3253	3269	3270*	3271	3539*	3540	3541*	3542	3711	3990	4159
				4486	4515	4560	4562	4563*	4564	4566	4570*	4571	4822	4846	5276
@ASIGN	001	007C	0071												
@ASTER	001	005C	0069												
@BCRDL	001	0050	0088												
@BE	001	0081	0043												
@BF	001	0090	0052												
@BH	001	0084	0041												
@BL	001	0082	0042												
@BLANK	001	0040	0065	3716	3722	5312									
@BM	001	0082	0054												
@BNE	001	0001	0046	3707											
@BNH	001	0004	0044												
@BNL	001	0002	0045												
@BNM	001	0002	0057												
@BNOL	001	0020	0050												
@BNOZ	001	0008	0049												
@BNP	001	0004	0056												
@BNZ	001	0001	0058												
@BOL	001	00A0	0048												
@BOZ	001	0088	0047												
@BP	001	0084	0053												
@BR	001	0001	0013	3164	3179*	3272	3274*	3275	3276	3276*	3278	3279	3280	3281	3282

```
                          3284  3285  3287  3288  3288* 3295* 3527  3536  3538* 3539  3540  3541
                          3542  3544  3545  3545  3546  3547  3547  3549  3549  3550  3551  3551
                          3555  3555  3556  3560  3560  3561  3563  3563  3564  3564  3565  3565
                          3566  3566  3567  3567  3573  3574  3575  3575  3576  3581  3581  3582
                          3582  3584  3584  3590* 3766  3768* 3771  3771  3772  3772  3773  3774
                          3775  3780  3782* 3785  3785  3786  3792  3794* 3797  3797  3798  3798
                          3799  3805  3807* 3810  3810  3811  3811  3812  3813  3818  3820* 3823
                          3823  3824  3824  3825  3827  3832  3834* 3837  3837  3838  3838  3839
                          3852  3858  3859  3864  3875  3877  3881  3882  3887  3986  3988  3989*
                          3990  3993  3994  3995  3996  3996  3997  3998  3998  3999  3999  4000
                          4003  4003  4004  4006  4008  4009  4011  4011  4019  4020  4020  4021
                          4022  4022  4023  4024  4025  4025  4026  4027  4029  4029  4030  4030
                          4031  4031  4032  4032  4034  4034  4035  4037  4037  4046* 4141  4149
                          4151  4155  4163  4164* 4171  4174  4179  4200  4202  4213  4220  4233
                          4268  4295  4298  4299  4299  4301  4302  4304  4305  4312  4312  4314
                          4332  4332  4334  4336  4361* 4373  4374* 4377  4382  4387* 4484  4505*
                          4555  4556  4557* 4558  4559  4560  4561  4562  4563  4564  4565  4566
                          4567  4568  4569  4569  4570  4571  4575  4576  4577  4577  4579  4584
                          4586  4588  4588  4589  4590  4592  4594  4599  4601  4603  4603  4604
                          4605  4605  4606  4606  4607  4608  4612  4613  4614  4614* 4664  4666*
                          4681  4681  4695  4701  4703  4705  4706  4707  4707  4711  4712  4715
                          4717  4718  4720  4721  4726  4727  4728  4729  4730  4731  4732  4732
                          4734  4740  4740  4741  4742  4743  4744  4917  4931* 4979  4985  4986
                          4986  4987  4988  4989  5051  5053* 5055  5056  5072  5074* 5076  5077
                          5078  5105  5105  5107  5108  5110  5111  5115  5116  5122  5122  5126
                          5126  5127  5136  5136  5137  5137  5138  5138  5139  5139  5140  5140
                          5141  5142  5147  5147  5148  5148  5149  5149  5150  5150  5151  5151
                          5152  5152  5271  5273  5274* 5276  5278  5280  5280  5290  5290  5295
                          5295  5296  5296  5297  5297  5298  5298  5299  5299  5303  5304  5304
                          5307  5313  5314  5319  5320  5320  5322*
@BT     001 0010 0051
@BZ     001 0081 0055
@B1     001 0001 0063    4194* 4250* 5306  5311
@CADDR  001 0002 0142    2568  2569  2570  3042  3043  3123  3142  3275  3276  3276  3288  3345
                          3346  3368  3377  3381  3383  3545  3774  3786  3799  3812  3825  3839
                          3995  3996  3998  4003  4011  4020  4022  4029  4030  4032  4158  4162
                          4171  4172  4200  4213  4233  4268  4300  4309  4312  4332  4334  4336
                          4393  4443  4445  4494  4565  4567  4605  4606  4656  4661  4671  4707
                          4886  4889  4890  4908  4979  5136  5137  5139  5140  5147  5148  5150
                          5151
@CARDL  001 0060 0087    1731
@CHARA  001 00C1 0072
@CHARF  001 00C6 0073
@CHARR  001 00D9 0074
@CHARZ  001 00E9 0075
@CLOFF  001 0010 0094
@CLON   001 0011 0093
@COMMA  001 006B 0066    3718
@CPLUS  001 004E 0079
@DADDR  001 0002 0140    3315  3544  3609  4028  4832  4854  4857  4885  4929  4930  4986  5055
                          5056  5077  5078  5088  5105  5107  5110
@DBFR1  001 0004 0129
@DBFR2  001 0005 0130    3995
@DCALK  001 0001 0081
@DCBCY  001 0009 0115    2397
@DCBT1  001 0050 0117    2400
```

```
                          CROSS REFERENCE

SYMBOL   LEN VALUE DEFN   REFERENCES                              VER 15, MOD 00  13/01/22  PAGE  80

@DCNT    001  0003 0128
@DCST1   001  0040 0116   2398
@DCTRL   001  0000 0125
@DCYL    001  0001 0126   3549*
@DD2     001  0003 0030
@DGET    001  0001 0134   3323  3399  3407  4083  5000  5186  5193
@DOLAR   001  005B 0068
@DOP2    001  0004 0028   3540* 3544* 3545* 3607  3608  4312* 4314* 4370* 4384* 4560* 4564* 4566*
                          5127  5139* 5140*
@DPLNG   001  0006 0132   3546  3605
@DPOS    001  0000 0133
@DPUT    001  0002 0135   4875  5200
@DSAD    001  0002 0127   3547* 3551* 3555  3556* 3560* 3563* 3567  3573* 3581* 3584* 3606  4028
                          4088  4832  4832* 5007  5055* 5056* 5077* 5078* 5107* 5108  5110* 5111
                          5136  5137
@DSBCY   001  0004 0106   2335
@DSCS1   001  0000 0107   2336
@DSIVF   001  0003 0138
@DSPIN   001  0002 0131
@DTRSZ   001  0018 0085
@DVBCY   001  0007 0108   2394
@DVRFY   001  0031 0136
@DWAIT   001  00FF 0137   4027
@DWBCY   001  0005 0103   2391
@DWSIZ   001  00C0 0105
@DWTB1   001  0003 0104   2392
@DZERO   001  00F0 0064
@D1      001  0002 0026   4221* 4224* 4225  4295* 4298* 4299  4299* 4302* 4304  4371* 4376* 4380*
                          4383* 5290
@EOF     001  001C 0077
@EOFTC   001  0075 0162
@EOS     001  001E 0076   2407  3724  3853  3876  4692  4726
@FDDBC   001  0000 0195
@FDE1    001  000C 0200
@FDFNA   001  000B 0198
@FDHLN   001  0002 0208
@FDLNC   001  0002 0193
@FDNSC   001  0003 0210
@FDSD    001  0000 0206
@FLACE   001  0009 0197
@FLDBC   001  0001 0196
@FLENT   001  0004 0201
@FLFNA   001  0002 0199
@FLHLN   001  0002 0209
@FLLNC   001  0002 0194
@FLNSC   001  0001 0211
@FLSD    001  0001 0207
@HDRLN   001  0007 0092   1759
@IAR     001  0010 0017   3335*
@INDEX   001  0001 0156   0157  5363
@INST3   001  0003 0032
@INST4   001  0004 0033   5126
@INST5   001  0005 0034
@INST6   001  0006 0035
@I1IAR   001  00C0 0020
@LINSZ   001  00F4 0084   1733
```

```
@MAPEN   001  0005 0089
@MINCR   001  2000 0083
@MINUS   001  0060 0080
@NOP     001  0080 0040   3031  3117  3119  3254  3344  3586  3993  4148  4491  4495  4714  4826
                          4831  5076  5115  5285  5359
@NUMBR   001  007B 0070
@OPD2    001  0004 0029
@OP1     001  0003 0027   3164* 3165* 3253* 3271* 3272* 3273* 3536* 3542* 3711* 3774* 3786* 3799*
                          3812* 3825* 3839* 3988* 3990* 4159* 4160* 4163* 4174* 4200* 4268* 4372*
                          4373* 4484* 4485* 4486* 4515* 4562* 4565* 4567* 4568  4571* 4605* 4606*
                          4656* 4661* 4711* 4712  4717* 4822* 4846* 5139  5273* 5276*
@OP2     001  0005 0031   3269* 3274  4332* 4334* 5122* 5126* 5138  5140
@PCTRL   001  0000 0149
@PDATA   001  0003 0151
@PGCSZ   001  0020 0082   0083
@PPLNG   001  0004 0148
@PRCNT   001  0001 0150
@PRETR   001  00C0 0154   3200  4398  4408  4452  4799  4805  5013  5207  5213
@PRINT   001  0040 0152   0154  3193  3214  3949  3956
@PSR     001  0004 0015   4165
@PWAIT   001  00FF 0158
@P1IAR   001  0020 0018
@P2IAR   001  0040 0019
@Q       001  0001 0024   3344* 3587  3730  3773* 3994* 4148* 4157* 4301  4335* 4377* 4529  4660*
                          4662* 4714* 4715* 4743* 4881  5076* 5116* 5353  5357
@REGL    001  0002 0012
@RETRN   001  0080 0153   0154  3207  4464  4812
@RLDWN   001  004F 0159
@RTRNC   001  0080 0161   3208  4465  4813
@SBLN    001  0005 0170
@SBLNL   001  0002 0184
@SCTSZ   001  0100 0100
@SDFLN   001  0007 0090
@SDF0    001  0000 0166
@SDF1    001  0001 0167
@SDF2    001  0002 0168
@SDF3    001  0003 0169
@SECCY   001  0030 0086
@SIST    001  0001 0181
@SLASH   001  0061 0067
@SLAST   001  0002 0183
@SMIDL   001  0003 0182
@SNULL   001  0080 0173
@SONLY   001  0000 0180
@STEXT   001  0007 0172
@STYPE   001  0006 0171
@TBCNT   001  0000 0160
@TBLEF   001  0010 0155   0157
@TBLIX   001  0011 0157
@UCB     001  0087 0039   3122  3255  3708  3719  3994  4157  4662  4668  4856  5116  5355
@UPARW   001  005A 0078
@VADDR   001  0002 0141   2128  2564  2576  2577  2578  2578  2592  2595  2597  2621  2622  2623
                          2661  2664  2667  2670  2673  2676  2679  2688  2691  2694  2697  2700
@VENTA   001  0056 0113   2395  2650
@VMDDV   001  00FE 0114
@VMFD1   001  0000 0109
```

```
@VMFD2  001  0001 0110
@VMRS3  001  0002 0112
@VMTRL  001  0001 0111
@VOLID  001  0006 0091
@VQ     001  0001 0025
@WSFIT  001  0500 0101
@WSTBL  001  0503 0102
@XR     001  0002 0014   3076   3078   3080   3082   3084   3086   3088   3090   3092   3094   3096   3098
                         3100   3168*  3273   3275*  3278   3279   3289   3289*  3296*  3712   3715   3715*
                         3716   3718   3721   3721*  3722   3724   3726   3853   3859   3864   3876   3882
                         3887   4160   4161*  4162   4165   4170*  4171   4172   4173   4173*  4174   4189*
                         4190   4191   4192   4220*  4222   4267*  4289   4296   4296*  4297   4297   4300
                         4309   4310   4315   4316   4360*  4372   4374   4375*  4378   4386*  4485   4506*
                         4559   4568*  4578   4580   4582   4586   4587   4590*  4591   4593   4595   4597
                         4601   4602   4607*  4612*  4691*  4692   4694   4696   4698   4700   4702   4704
                         4704*  4706*  4710   4710*  4711   4717   4718*  4719   4719*  4720   4721   4722*
                         4734   4824   4825   4840*  4848*  5278   5287   5303   5306   5306*  5311   5311*
                         5312   5319
@ZERO   001  0000 0062   3556   4241   4269
B$ADMK  001  0001 2032
B$ADSW  001  159D 2031
B$ARMK  001  0001 2017
B$ARSW  001  0A45 2016
B$BABF  001  1D00 1822
B$BCKT  001  1590 1944
B$BDPL  001  19E8 1896
B$BDSA  001  19EA 1897
B$BINO  001  1A6A 1960
B$BRLN  001  19F1 1895
B$BROP  001  1AF7 2001
B$BRVA  001  19EF 1894
B$BRVP  001  19EE 1893
B$BTAB  001  1996 1892
B$CADR  001  1AF9 2002
B$CASA  001  0000 1837
B$CASC  001  0671 1841
B$CASM  001  0608 1839
B$CBAS  001  14BB 1967
B$CBFA  001  0CBC 1922
B$CCGT  001  0600 1847
B$CCLS  001  0695 1853
B$CCON  001  001F 1920
B$CDAT  001  0600 1833
B$CDEF  001  0600 1834
B$CDIM  001  0673 1835
B$CDUM  001  0000 1871
B$CEND  001  0600 1869   1870
B$CEOF  001  0600 1870
B$CFOR  001  0600 1842
B$CGET  001  06A3 1850
B$CGSB  001  0690 1848
B$CGTO  001  06B3 1846
B$CIFA  001  0600 1844
B$CIFC  001  0600 1845
B$CIMG  001  0600 1859
B$CINP  001  0600 1854
```

```
SYMBOL  LEN VALUE DEFN   REFERENCES                        VER 15, MOD 00  13/01/22  PAGE  83

B$CLTA  001  0000 1836
B$CLTC  001  0669 1840
B$CLTM  001  0600 1838
B$CMAT  001  0600 1860
B$CMGT  001  0665 1861
B$CMIN  001  06D3 1862
B$CMPR  001  069B 1865
B$CMPT  001  069B 1864
B$CMPU  001  0600 1866
B$CMRD  001  06D0 1863
B$CNXT  001  0600 1843
B$CPCT  001  0CA8 1925
B$CPRT  001  0600 1857
B$CPRU  001  0600 1858
B$CPSE  001  06E7 1867
B$CPUT  001  0600 1851
B$CPWA  001  0CA6 1996
B$CRAD  001  150D 1966
B$CRBS  001  1509 1968
B$CREA  001  06CF 1855
B$CREM  001  0000 1832
B$CRMK  001  0001 2044
B$CRSR  001  06E3 1856
B$CRST  001  06A6 1852
B$CRSW  001  0E42 2043
B$CRTN  001  06CF 1849
B$CSBF  001  0600 1819   1833  1834  1835  1838  1839  1840  1841  1842  1843  1844  1845  1846
                        1847  1848  1849  1850  1851  1852  1853  1854  1855  1856  1857  1858
                        1859  1860  1861  1862  1863  1864  1865  1866  1867  1868  1869  1872
                        1873  1874  1875  1876
B$CSCN  001  14B0 1941
B$CSMK  001  0007 2047
B$CSSW  001  14BC 2046
B$CSTP  001  06D6 1868
B$CSTR  001  14CC 1965
B$CSXA  001  2000 1825
B$CTYP  001  0A5F 1919
B$CVPD  001  0C5D 1924
B$CVPG  001  0CA5 1923
B$CWRK  001  F500 1993
B$DIST  001  0700 1885
B$DLNK  001  1B37 1991
B$DL4T  001  1A6B 1962
B$DPWA  001  0E46 1997
B$DST2  001  073A 1886
B$ERMK  001  0007 2020
B$ERSW  001  0993 2019
B$FACA  001  0E53 1928
B$FAIS  001  15AC 1945
B$FAIW  001  15A0 1946
B$FCON  001  0A46 1918
B$FORT  001  1B0E 1987
B$FPWA  001  15AC 1998
B$FRMK  001  0007 2038
B$FRSW  001  16CC 2037
B$FSC1  001  0E4C 1929
```

```
B$FSC2   001  0E4D 1930
B$FSMK   001  0007 2029
B$FSSW   001  0E5C 2028
B$FSVA   001  0E4F 1931
B$FTND   001  1B0B 1989
B$FTPT   001  1B0D 1988
B$FVME   001  15A2 1950
B$FVMP   001  15A4 1951
B$FVMS   001  15A6 1952
B$FVPE   001  15A8 1947
B$FVPP   001  15AA 1948
B$FVPS   001  15AC 1949
B$GBSW   001  08AF 2022
B$GBWK   001  0001 2023
B$GETC   001  0867 1899
B$GPTR   001  0878 1901
B$GTBF   001  1E00 1823
B$IFMK   001  0007 2041
B$IFSW   001  16E5 2040
B$INVT   001  1B38 1981
B$KWMK   001  0001 2035
B$KWSW   001  159E 2034
B$LBAS   001  185E 1972
B$LBSV   001  18E7 1970
B$LDRP   001  1A00 1820
B$LINE   001  07D0 1887
B$LIST   001  1853 1954
B$LRTN   001  18EB 1971
B$LSTR   001  1862 1969
B$LTYP   001  18F2 1955
B$MATR   001  18F3 1957
B$MBMK   001  0007 2056
B$MBSW   001  1903 2055
B$MFBK   001  1B8F 1983
B$MGMK   001  0007 2053
B$MGSW   001  18FF 2052
B$MPMK   001  0007 2059
B$MPSW   001  1981 2058
B$MRMK   001  0007 2050
B$MRSW   001  0DDE 2049
B$NUMC   001  0873 1900
B$NXMK   001  0007 2026
B$NXSW   001  071D 2025
B$PARP   001  0A41 1908
B$PBNL   001  0A01 1914
B$PCAD   001  0A40 1909
B$PCDL   001  09D3 1913
B$PCPG   001  0A35 1912
B$PECT   001  0A44 1916
B$PERC   001  0A39 1915
B$PFAE   001  0033 1906
B$PFCL   001  009D 1907
B$PFNC   001  094E 1904
B$PFWP   001  0015 1905
B$PNBY   001  0A41 1910
B$PPWA   001  0A35 1995
```

```
B$PRM1  001  1AF3 1999
B$PTBF  001  1F00 1824
B$PUTC  001  093A 1903
B$PVAD  001  0A43 1911
B$RMRK  001  1AE6 1964
B$RTRN  001  1AF5 2000
B$SABF  001  1C00 1821
B$SCAN  001  1514 1943
B$SCAT  001  13C8 1938
B$SCON  001  001B 1921
B$SCVT  001  12E0 1936
B$SDPL  001  07DA 1889
B$SFAB  001  0E48 1933
B$SFNT  001  143C 1939
B$SLDT  001  109C 1935
B$SLVT  001  1062 1934
B$SNAT  001  131A 1937
B$SPAT  001  07E0 1890
B$SSTA  001  1BAC 1985
B$STAS  001  061B 1874
B$STIF  001  0606 1876
B$STMA  001  061B 1875
B$STML  001  0600 1873
B$STRL  001  0600 1872
B$SVRB  001  0E46 1932
B$SYMB  001  0DBC 1927
B$TCD2  001  0001 2005
B$TLTH  001  0002 2006   2007
B$TOD1  001  0000 2004
B$TOTB  001  1AF8 2007
B$TTAB  001  1AFA 2003   2007
B$TYPE  001  0739 1888
B$WORK  001  15A0 1992
B$ZDBN  001  19F2 1959
B@ABAS  001  0007 2592
B@ACD1  001  0001 2589   2590
B@ACD2  001  0003 2590   2591
B@AFLG  001  0000 2584
B@ALLA  001  005C 2409
B@AMAX  001  0005 2591   2592
B@BLNK  001  0040 2418
B@BLSZ  001  0100 2543   2682  2685  2688  2703  2706
B@BREQ  001  0084 2198
B@BRHI  001  0088 2199
B@BRLO  001  0082 2197
B@BRNE  001  0094 2201
B@BRNH  001  0098 2202
B@BRNL  001  0092 2200
B@CADD  001  0006 2067
B@CADF  001  0058 2108
B@CBAS  001  0003 2595
B@CBNX  001  004A 2101
B@CBRA  001  0046 2099
B@CBRC  001  0044 2098
B@CBRD  001  0048 2100
B@CBRS  001  004C 2102
```

```
SYMBOL   LEN VALUE DEFN   REFERENCES                          VER 15, MOD 00  13/01/22  PAGE  86

B@CCLS   001  005E 2111
B@CCMC   001  0042 2097
B@CCMF   001  0040 2096
B@CCNT   001  001F 2521
B@CCSA   001  003E 2095
B@CDCA   001  006A 2117
B@CDDL   001  006C 2118
B@CDIV   001  000C 2070
B@CDMN   001  0001 2594   2595
B@CDWA   001  006E 2119
B@CEOF   001  0070 2120
B@CEOP   001  0068 2116
B@CFCI   001  0016 2075
B@CFN0   001  0012 2073
B@CFN1   001  0014 2074
B@CFOR   001  004E 2103
B@CGET   001  0052 2105
B@CHAR   001  0000 2534
B@CHLT   001  0004 2066
B@CIEX   001  00C5 2494
B@CIMH   001  0066 2115
B@CINI   001  0056 2107
B@CIPI   001  00D7 2497
B@CIS2   001  00E2 2500
B@CMF1   001  0018 2076
B@CMF2   001  001A 2077
B@CMF3   001  001C 2078
B@CMMA   001  006B 2429
B@CMPY   001  000A 2069
B@CMSM   001  001E 2079
B@CNEG   001  0010 2072
B@CNXT   001  0050 2104
B@COLN   001  007A 2431
B@CPMK   001  00FF 2339   2343  2347  2348  2382
B@CPRS   001  0060 2112
B@CPRU   001  0062 2113
B@CPUT   001  0054 2106
B@CPWR   001  000E 2071
B@CRSR   001  005A 2109
B@CRST   001  005C 2110
B@CSA1   001  0036 2091
B@CSA2   001  0038 2092
B@CSB1   001  003A 2093
B@CSC1   001  002A 2085
B@CSD0   001  002E 2087
B@CSD1   001  0030 2088
B@CSD2   001  0032 2089
B@CSF1   001  0022 2081
B@CSF2   001  0024 2082
B@CSTA   001  0034 2090
B@CSTC   001  0028 2084
B@CSTF   001  0020 2080
B@CSTH   001  0064 2114
B@CSTX   001  003C 2094
B@CSUB   001  0008 2068
B@CSVC   001  0002 2065
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES
B@CTYP	001	0020	2519	
B@CUSC	001	002C	2086	
B@CUSF	001	0026	2083	
B@CVAR	001	005B	2408	
B@DAMK	001	0080	2587	
B@DASA	001	00FF	2348	
B@DASC	001	0040	2352	
B@DASM	001	0038	2350	
B@DCGT	001	0050	2358	
B@DCLS	001	0054	2364	
B@DDAT	001	0024	2344	
B@DDEF	001	0034	2345	
B@DDIM	001	0004	2346	
B@DDUM	001	00FF	2382	
B@DEC0	001	00F0	2477	
B@DEC1	001	00F1	2478	
B@DEC2	001	00F2	2479	
B@DEC3	001	00F3	2480	
B@DEC4	001	00F4	2481	
B@DEC5	001	00F5	2482	
B@DEC6	001	00F6	2483	
B@DEC7	001	00F7	2484	
B@DEC8	001	00F8	2485	
B@DEC9	001	00F9	2486	
B@DEND	001	0058	2380	2381
B@DEOF	001	0058	2381	
B@DFOR	001	0028	2353	
B@DGET	001	0040	2361	
B@DGSB	001	0020	2359	
B@DGTO	001	0044	2357	
B@DIFA	001	0048	2355	
B@DIFC	001	004C	2356	
B@DIGS	001	007B	2411	
B@DIMG	001	003C	2370	
B@DINP	001	0000	2365	
B@DIVD	001	0061	2428	
B@DLTA	001	00FF	2347	
B@DLTC	001	0040	2351	
B@DLTM	001	0038	2349	
B@DL01	001	0001	2662	2665
B@DL02	001	0003	2665	2668
B@DL03	001	0005	2668	2671
B@DL04	001	0007	2671	2674
B@DL05	001	0009	2674	2677
B@DL06	001	000B	2677	2680
B@DL07	001	0045	2680	2683
B@DL08	001	0145	2683	2686
B@DL09	001	0245	2686	2689
B@DL10	001	0289	2689	2692
B@DL11	001	02C3	2692	2695
B@DL12	001	02FD	2695	2698
B@DL13	001	0337	2698	2701
B@DL14	001	0371	2701	2704
B@DL15	001	0471	2704	2707
B@DL16	001	0507	2707	
B@DMAT	001	0008	2371	

```
B@DMGT   001  0044 2372
B@DMIN   001  0038 2373
B@DMPR   001  0048 2376
B@DMPT   001  004C 2375
B@DMPU   001  0054 2377
B@DMRD   001  003C 2374
B@DNXT   001  0044 2354
B@DPNT   001  004B 2419
B@DPRT   001  002C 2368
B@DPRU   001  0030 2369
B@DPSE   001  0050 2378
B@DPUT   001  0040 2362
B@DREA   001  000C 2366
B@DREM   001  00FF 2343
B@DRSR   001  005C 2367
B@DRST   001  0050 2363
B@DRTN   001  005C 2360
B@DSCY   001  0004 2335
B@DSIF   001  001C 2384
B@DSLT   001  0010 2383
B@DSML   001  0010 2385
B@DSNS   001  0018 2337
B@DSS1   001  0000 2336
B@DSTP   001  0054 2379
B@DTBN   001  0010 2401
B@DTB1   001  0050 2400
B@DTCY   001  0009 2397
B@DTSN   001  0010 2399
B@DTS1   001  0040 2398
B@DTYP   001  0040 2513
B@DVCY   001  0007 2394
B@DVC1   001  0056 2395
B@DWCY   001  0005 2391
B@DWT1   001  0003 2392
B@D1MK   001  0080 2585
B@D2MK   001  00C0 2586
B@EOST   001  001E 2407
B@EQUL   001  007E 2433
B@EXPC   001  00C5 2410
B@FOFL   001  005C 2412
B@FVAD   001  0001 2597
B@GETC   001  0001 2536
B@GETE   001  00FF 2537
B@GETS   001  0000 2535
B@GRTR   001  006E 2430
B@ICON   001  0050 2492
B@LADD   001  0001 2136
B@LADF   001  0002 2177
B@LADV   001  0008 2621   2642
B@LBIN   001  0002 2546   2547  2553
B@LBNX   001  0003 2170
B@LBRA   001  0003 2168
B@LBRC   001  0004 2167
B@LBRD   001  0003 2169
B@LBRS   001  0001 2171
B@LCCA   001  0004 2577
```

```
B@LCCC  001  0001 2129   2167
B@LCDV  001  0004 2622   2643
B@LCER  001  0001 2127   2191
B@LCFN  001  0004 2578
B@LCLN  001  0002 2132   2183  2184  2191
B@LCLS  001  0001 2180
B@LCMC  001  0001 2166
B@LCMF  001  0001 2165
B@LCNA  001  0006 2576
B@LCNN  001  0001 2130   2155  2164  2176  2188
B@LCOP  001  0001 2126   2134  2135  2136  2137  2138  2139  2140  2141  2142  2143  2144  2145
                         2146  2147  2148  2149  2150  2151  2152  2153  2154  2155  2156  2157
                         2158  2159  2160  2161  2162  2163  2164  2165  2166  2167  2168  2169
                         2170  2171  2172  2173  2174  2175  2176  2177  2178  2179  2180  2181
                         2182  2183  2184  2185  2186  2187  2188  2189
B@LCRV  001  0013 2620   2640
B@LCSA  001  0002 2164
B@LCVA  001  0002 2128   2142  2143  2144  2145  2146  2147  2148  2149  2150  2151  2153  2154
                         2156  2157  2158  2159  2160  2161  2162  2167  2168  2169  2170  2172
                         2173  2174  2186  2187
B@LCXX  001  0001 2131   2163  2175  2177  2181  2182
B@LDAT  001  0004 2290
B@LDCA  001  0003 2186
B@LDDL  001  0003 2187
B@LDDM  001  0004 2550
B@LDEF  001  0003 2291
B@LDIM  001  0003 2292
B@LDIN  001  0004 2549   2550  2551
B@LDIV  001  0001 2139
B@LDMN  001  0002 2547   2576  2577  2589  2590  2591  2594  2621  2622
B@LDSN  001  0004 2551
B@LDWA  001  0002 2188
B@LELP  001  0010 2619
B@LEND  001  0003 2319
B@LEOF  001  0001 2189
B@LEOP  001  0001 2185
B@LERC  001  0003 2191
B@LESP  001  0008 2618
B@LESS  001  004C 2420
B@LET$  001  005B 2440
B@LET#  001  007B 2441
B@LET@  001  007C 2442
B@LETA  001  00C1 2444
B@LETB  001  00C2 2446
B@LETC  001  00C3 2447
B@LETD  001  00C4 2448
B@LETE  001  00C5 2449
B@LETF  001  00C6 2450
B@LETG  001  00C7 2451
B@LETH  001  00C8 2452
B@LETI  001  00C9 2453
B@LETJ  001  00D1 2454
B@LETK  001  00D2 2455
B@LETL  001  00D3 2456
B@LETM  001  00D4 2457
B@LETN  001  00D5 2458
```

```
B@LETO   001  00D6 2459
B@LETP   001  00D7 2460
B@LETQ   001  00D8 2461
B@LETR   001  00D9 2462
B@LETS   001  00E2 2463
B@LETT   001  00E3 2464
B@LETU   001  00E4 2465
B@LETV   001  00E5 2466
B@LETW   001  00E6 2467
B@LETX   001  00E7 2468
B@LETY   001  00E8 2469
B@LETZ   001  00E9 2470
B@LEXP   001  0008 2509
B@LFCI   001  0003 2144
B@LFNA   001  0002 2623   2644
B@LFN0   001  0003 2142
B@LFN1   001  0003 2143
B@LFOR   001  0003 2172
B@LFRT   001  0004 2564   2565
B@LGET   001  0003 2174
B@LGSB   001  0005 2298
B@LGTO   001  0004 2297
B@LHLT   001  0001 2135
B@LIEX   001  0002 2495
B@LIFN   001  0003 2558
B@LILP   001  0009 2617   2635   2636   2637
B@LIMG   001  0001 2309
B@LIMH   001  0003 2184
B@LINI   001  0002 2176
B@LINP   001  0005 2304
B@LIPI   001  0003 2498
B@LISP   001  0005 2616   2624   2630   2631   2632
B@LIS2   001  0005 2501
B@LIVT   001  0001 2574
B@LKCL   001  0005 2303
B@LKFR   001  0003 2294
B@LKGT   001  0003 2300
B@LKIF   001  0002 2296
B@LKON   001  0002 2329
B@LKPT   001  0003 2301
B@LKPU   001  000A 2308
B@LKRR   001  0007 2306
B@LKRT   001  0005 2302
B@LKTO   001  0002 2323
B@LLET   001  0003 2293
B@LL01   001  0002 2661   2662
B@LL02   001  0002 2664   2665
B@LL03   001  0002 2667   2668
B@LL04   001  0002 2670   2671
B@LL05   001  0002 2673   2674
B@LL06   001  0002 2676   2677
B@LL07   001  003A 2679   2680
B@LL08   001  0100 2682   2683
B@LL09   001  0100 2685   2686
B@LL10   001  0044 2688   2689
B@LL11   001  003A 2691   2692
```

```
B@LL12   001  003A 2694    2695
B@LL13   001  003A 2697    2698
B@LL14   001  003A 2700    2701
B@LL15   001  0100 2703    2704
B@LL16   001  0096 2706    2707
B@LMAT   001  0003 2310
B@LMF1   001  0003 2145
B@LMF2   001  0003 2146
B@LMF3   001  0003 2147
B@LMGT   001  0006 2311
B@LMIN   001  0008 2312
B@LMPR   001  0008 2315
B@LMPT   001  0006 2314
B@LMPU   001  000D 2316
B@LMPY   001  0001 2138
B@LMRD   001  0007 2313
B@LMSM   001  0003 2148
B@LNEG   001  0001 2141
B@LNEX   001  0004 2295
B@LNXT   001  0003 2173
B@LPAR   001  004D 2421
B@LPRS   001  0002 2181
B@LPRT   001  0005 2307
B@LPRU   001  0002 2182
B@LPSE   001  0005 2317
B@LPUT   001  0002 2175
B@LPWR   001  0001 2140
B@LREA   001  0004 2305
B@LREM   001  0003 2289
B@LRSR   001  0001 2178
B@LRST   001  0001 2179
B@LRTN   001  0006 2299
B@LSA1   001  0003 2160
B@LSA2   001  0003 2161
B@LSB1   001  0003 2162
B@LSC1   001  0003 2154
B@LSDF   001  0004 2544
B@LSD0   001  0003 2156
B@LSD1   001  0003 2157
B@LSD2   001  0003 2158
B@LSF1   001  0003 2150
B@LSF2   001  0003 2151
B@LSKW   001  0002 2560
B@LSNO   001  0002 2553
B@LSPT   001  0003 2568    2571
B@LSTA   001  0003 2159
B@LSTC   001  0003 2153
B@LSTE   001  0004 2324
B@LSTF   001  0003 2149
B@LSTH   001  0003 2183
B@LSTP   001  0004 2318
B@LSTX   001  0002 2163
B@LSUB   001  0001 2137
B@LSVC   001  0001 2134
B@LTHN   001  0004 2325
B@LTYP   001  0001 2554
```

```
B@LUFN   001  0002 2561
B@LUSC   001  0002 2155
B@LUSF   001  0001 2152
B@LVPG   001  0100 2648    2651
B@MINS   001  0060 2427
B@MULT   001  005C 2424
B@NAAR   001  001D 2612    2642  2694
B@NCAR   001  001D 2613    2643  2697
B@NCRV   001  001D 2611    2640  2691
B@NDGT   001  000A 2604    2610
B@NEQL   001  007F 2434
B@NFRT   001  000A 2563    2565
B@NICN   001  0006 2606    2608
B@NIEL   001  0007 2608    2624  2630  2635
B@NIFN   001  0018 2557
B@NIVR   001  0001 2607    2608
B@NIVT   001  0057 2573
B@NLDV   001  0122 2610    2632  2637  2688
B@NLRV   001  001D 2609    2631  2636  2679
B@NLTR   001  001D 2603    2609  2610  2611  2612  2613  2614
B@NSKW   001  0004 2559
B@NSPT   001  0028 2567
B@NUFN   001  001D 2614    2644  2700
B@NVPG   001  0100 2647    2651
B@NXHI   001  00E3 2528
B@NXLO   001  001E 2527
B@NXZR   001  0080 2526    2527  2528
B@PLUS   001  004E 2422
B@POWR   001  005A 2423
B@PREC   001  0020 2515
B@PROD   001  0023 2624
B@PRPL   001  0002 2211
B@PRPN   001  0001 2210
B@PRPR   001  0004 2213
B@PRPS   001  0003 2212
B@PRRC   001  0007 2216
B@PRRL   001  0008 2217
B@PRSL   001  0005 2214
B@PRSS   001  0006 2215
B@PTAB   001  0000 2569
B@PTAD   001  0001 2570
B@PTSA   001  0002 2571
B@PUD1   001  0006 2227
B@PUD2   001  0007 2228
B@PUI0   001  0001 2221
B@PUI1   001  0004 2222
B@PUI2   001  0005 2223
B@PUNL   001  0002 2225
B@PUNS   001  0003 2226
B@PURE   001  0020 2231
B@PUTM   001  0010 2230
B@RPAR   001  005D 2425
B@SADV   001  00E8 2642    2645
B@SAVL   001  0B76 2638    2655
B@SAVS   001  065E 2633    2654
B@SCDV   001  0074 2643    2645
```

```
                              CROSS REFERENCE

SYMBOL  LEN VALUE DEFN   REFERENCES                        VER 15, MOD 00  13/01/22  PAGE  93

B@SCLN  001  005E 2426
B@SCRV  001  0227 2640   2654   2655
B@SDMK  001  0080 2555
B@SEXP  001  0004 2508
B@SFAT  001  0196 2645   2654   2655   2706
B@SFNA  001  003A 2644   2645
B@SFRT  001  0028 2565
B@SIEL  001  003F 2635   2638
B@SIES  001  0023 2630   2633
B@SIGN  001  0010 2517
B@SLDL  001  0A32 2637   2638
B@SLDS  001  05AA 2632   2633
B@SLVL  001  0105 2636   2638
B@SLVS  001  0091 2631   2633
B@SQUO  001  007D 2432
B@STAT  001  0000 2507
B@TASA  001  0012 2242
B@TASC  001  001E 2248
B@TASM  001  0018 2244
B@TASS  001  007B 2249
B@TCGT  001  0030 2257
B@TCLS  001  0042 2263
B@TDAT  001  0006 2238
B@TDEF  001  0009 2239
B@TDIM  001  000C 2240
B@TDUM  001  0078 2281
B@TEND  001  0072 2279
B@TEOF  001  0075 2280
B@TFOR  001  0021 2251
B@TGET  001  0039 2260
B@TGSB  001  0033 2258
B@TGTO  001  002D 2256
B@TIFA  001  0027 2253
B@TIFC  001  002A 2254
B@TIFS  001  007D 2255
B@TIMG  001  0054 2269
B@TINP  001  0045 2264
B@TLTA  001  000F 2241
B@TLTC  001  001B 2245
B@TLTM  001  0015 2243
B@TLTS  001  0079 2246
B@TMAS  001  007C 2250
B@TMAT  001  0057 2270
B@TMGT  001  005A 2271
B@TMIN  001  005D 2272
B@TMLS  001  007A 2247
B@TMPR  001  0066 2275
B@TMPT  001  0063 2274
B@TMPU  001  0069 2276
B@TMRD  001  0060 2273
B@TNXT  001  0024 2252
B@TPRT  001  004E 2267
B@TPRU  001  0051 2268
B@TPSE  001  006C 2277
B@TPUT  001  003C 2261
B@TRAC  001  0080 2511
```

```
B@TREA   001  0048 2265
B@TREM   001  0003 2237
B@TRSR   001  004B 2266
B@TRST   001  003F 2262
B@TRTN   001  0036 2259
B@TSTP   001  006F 2278
B@VMC1   001  0056 2650
B@VMLB   001  F0CD 2655
B@VMSB   001  F5E5 2654
B@VMSZ   001  0000 2651    2653   2654   2655
B@VMTB   001  0000 2653
B@ZNEG   001  00D0 2524
B@ZPOS   001  00F0 2523
CPATCH   001  1842 4655    3942
CVBHCT   001  0E1D 3301    3277*  3290*
CVBHEX   001  0DB9 3268    3435   4185   4273   4284   4937   5143
CVBH20   001  00F0 3265    3280   3281
CVBH25   001  00FA 3266    3282   3285
CVBH35   001  0001 3267    3289
CVBH50   006  0DD7 3277    3269*  3274
CVBH52   004  0DDD 3278    3291
CVBH55   003  0DF6 3285    3283
CVBH60   003  0E01 3288    3286
CVBH76   004  0E11 3295    3272*
CVBH78   004  0E15 3296    3273*
CVBH80   004  0E19 3297    3271*
CVBH90   002  0E1F 3305    3270
CVBH92   001  0E20 3306    3284   3287
CVBH94   001  0E21 3307    3132   3290
C4BCHC   001  0004 5347
C4BCHR   001  1D4E 5335    5303*  5304
C4BINI   001  1D4D 5333    5280
C4BIN2   001  1CE2 5270    3366   3375   4927   5086   5271   5274
C4BLEN   002  1D4A 5345    5319*  5320*
C4BLNK   003  1CFD 5353
C4BLOW   001  00F0 5349    5287
C4BLVL   002  0002 5351    5280   5295   5296   5297   5298   5299   5304
C4BNMC   004  1CF9 5357
C4BNOP   001  0080 5359
C4BSAV   002  1D50 5339    5278*  5320
C4BSPC   001  0087 5355
C4BVAL   002  1D4C 5331    3368   3377   4929   5088   5280*  5295   5295*  5296   5297   5297*  5298   5298*
                          5299*  5304*  5351
C4BWRK   002  1D4A 5328    5296*  5299   5345   5351
C4BYT1   001  1D4B 5330
C4B100   004  1CF8 5281    5357
C4B200   003  1CFC 5285    5307   5353
C4B300   003  1CFF 5287    5313
C4B590   003  1D2E 5311    5290   5314
C4B600   003  1D31 5312    5285
C4B700   003  1D3A 5319    5288
C4B800   004  1D41 5322    5273*  5291
C4B850   004  1D45 5324    5276*
C4B900   001  1D51 5341    5281*  5290*
C4END    001  1D52 5360
DL2C01   002  103A 3599    3539   3541   3549
```

```
DL2C05  002  103C 3600    3545
DL2C48  001  1036 3597    3547  3551
DL2DPL  006  1042 3605    3546*
DL2END  001  1045 3610
DL2E01  001  0001 3529    3547  3549  3551  3555  3567  3575
DL2E02  001  0002 3530    3560  3563  3581
DL2E18  001  0018 3531    3561
DL2E60  001  0060 3532    3576
DL2E7C  001  007C 3534    3573
DL2ICS  001  0FAC 3535    3348  3385  4039  4755  4859  4867
DL2K18  002  1038 3598    3564
DL2K60  002  1033 3595    3582
DL2K80  002  1035 3596    3563  3581
DL2LST  001  103D 3604    3547* 3549* 3551* 3555  3556* 3560* 3563* 3567  3573* 3581* 3584* 3589
                         3606
DL2PHY  001  103F 3606
DL2RAD  002  1044 3609    3345* 3346* 3381* 3383* 3560  4028* 4854*
DL2SAD  005  0FC4 3607    3567* 3574* 3575* 3576  3582* 3584
DL2SEC  005  0FCD 3608    3555* 3561  3564* 3565  3565* 3566  3566* 3575
DL2SWH  003  1022 3587
DL2TSD  001  0083 3533    3574
DL2000  001  0FB0 3537    3527  3538
DL2001  005  0FC0 3544    3540* 3607
DL2002  005  0FC9 3546    3544* 3545* 3608
DL2005  004  0FCE 3547    3550
DL2006  004  0FDC 3551    3548
DL2008  004  0FF9 3565    3562
DL2010  003  100F 3576
DL2100  004  101D 3584    3577
DL2110  003  1021 3586    3587
DL2900  004  102A 3590    3536* 3586
DL2910  004  102E 3591    3542*
DPATCH  001  184C 4658    3943
SCACNT  002  1091 3736    3726* 3727*
SCACOF  001  0087 3708
SCACOM  001  0001 3707
SCAINC  001  0001 3706    3715  3721
SCAMMA  003  106E 3730
SCANIT  001  1051 3710    3173
SCASVE  002  108F 3735    3712* 3727
SCASV1  001  108E 3734
SCA100  003  1060 3715    3717
SCA200  003  1063 3716    3714
SCA250  003  106D 3719    3730
SCA300  003  1070 3721    3723
SCA400  004  1080 3726    3719
SCA500  004  108A 3729    3711* 3725
UCLCHG  001  14A9 4369    4317
UCL900  004  177D 4556    3860  3883  4555  4557
UCL905  005  178E 4561    4560*
UCL907  005  179C 4565    4564*
UCL908  005  17A4 4567    4566*
UCL910  003  17B6 4575    4608
UCL915  003  17D2 4584    4581
UCL920  001  17D5 4585    4583
UCL930  005  17D9 4587    4565* 4568  4579  4605*
```

```
SYMBOL   LEN VALUE DEFN    REFERENCES                      VER 15, MOD 00  13/01/22  PAGE  96

UCL932   003 1800 4599    4596
UCL933   001 1803 4600    4598
UCL935   005 1807 4602    4567* 4594   4606*
UCL945   003 1821 4612    4576  4589   4592   4604
UCL949   005 1824 4613    4562*
UCL950   004 1830 4616    4571*
UCL951   001 0000 4620    4575  4578   4580   4582   4586* 4587   4591   4593   4595   4597   4601* 4602
                         4615
UCL952   001 0040 4621    4575  4591
UCL953   001 00F0 4622    4578  4593
UCL954   001 00FF 4623    4558  4584   4599
UCL961   002 1835 4624    4570
UCL962   002 1837 4625    4563
UCL963   001 1838 4626    4586  4601
UCL964   002 183A 4627    4577  4588   4590   4603   4605   4606   4607
UCL970   002 183C 4628    4556* 4614
UCL971   002 183E 4629    4559* 4612
UCL972   001 183F 4630    4561* 4588*  4603*
UCL973   001 1840 4631    4569  4569*  4577*  4613
UCL974   001 1841 4632    4558* 4584*  4599*  4615
UDBASE   004 1AD4 5009    4917  4931
UDCBN2   001 0002 4423    4298
UDCB20   001 0014 4430
UDCB21   001 0015 4424    4309*
UDCB22   001 0016 4425    4310*
UDCB24   001 0018 4435    4461
UDCB28   001 001C 4434    4442
UDCB32   001 0020 4427    4315  4458   4459   4460   4472   4473
UDCB50   001 0032 4429
UDCB54   001 0036 4426    4315*
UDCB55   001 0037 4428    4316*
UDCDP3   001 0003 4122    4172
UDCIML   001 0033 4121    4251  4439   4440
UDCPLL   001 0078 4392    4405  4413   4414   4415   4453
UDC050   001 0020 4111    4225
UDC060   001 005C 4112    4310  4316
UDC089   001 00FF 4113    4199  4261
UDC090   001 001F 4114    4179  4370   4371
UDC091   001 0000 4115
UDC093   001 0007 4116    4295
UDC094   001 0037 4117    4297
UDC095   001 0040 4118    4249
UDC101   001 0006 4119    4191  4192   4446
UDC103   001 0008 4120    4188  4190   4296   4297
UDC106   001 0046 4123    4162*
UDC107   001 004B 4124    4158*
UDC108   001 0050 4125    4165*
UDC111   001 0045 4126    4187  4251*
UDC112   001 0004 4127    4173  4289
UDC120   001 0007 4128    4289*
UDC127   001 0077 4129    4250
UDC130   001 0007 4130    4190*
UDC131   001 0011 4131    4191*
UDC132   001 001B 4132    4192*
UDC137   001 0038 4133    4287
UDC143   001 0057 4134    4194
```

```
                                      CROSS REFERENCE

SYMBOL   LEN VALUE DEFN    REFERENCES                      VER 15, MOD 00  13/01/22  PAGE  97

UDC144   001  04F2 4135    4158
UDC145   001  04FA 4136    4162
UDC200   004  12D5 4148    4201
UDC210   005  12DC 4151    4214   4502
UDC215   004  1316 4170    4159*
UDC230   004  1330 4185
UDC300   004  135F 4201    4148*  4157*
UDC400   005  1366 4213    4234   4337   4470
UDC402   003  137A 4222    4221*  4224*  4225   4226   4335*
UDC405   006  1393 4232
UDC410   004  13A2 4240    4154   4223
UDC451   004  13D2 4267    4153   4242   4469
UDC452   004  13DF 4273
UDC453   002  13F2 4286    4141   4151*  4164   4171*  4179*  4213   4220   4233*  4275   4312   4332   4336*
                          4492   4494   4499   4503*  4519*  4524*
UDC454   003  1401 4296    4305
UDC455   004  1404 4297    4295*  4299   4302*  4304
UDC456   005  140F 4300    4298*  4299*  4301
UDC458   005  143F 4315    4312*  4314*
UDC459   001  144B 4318    4388
UDC460   006  146C 4335    4332*  4334*
UDC461   004  147B 4339    4202
UDC470   004  1499 4356    4149   4200*  4268*
UDC497   004  149D 4360    4155   4160*
UDC498   004  14A1 4361    4163*
UDC499   004  14A5 4362    4174*
UDC540   004  14C0 4376    4385
UDC550   003  14C9 4378    4376*  4377*  4380*  4381
UDC551   005  14C4 4377    4370*  4384*
UDC553   003  14D9 4382    4371*  4383*
UDC555   006  14DC 4383    4379
UDC560   004  14EC 4386    4372*
UDC565   004  14F0 4387    4373*
UDC570   001  0017 4432    4374   4441
UDC571   001  003F 4433    4376
UDC720   002  14F9 4393    4172*  4213
UDC740   001  14FA 4394    4152   4199*  4232*  4241   4261*  4269*
UDC741   001  14FB 4395    4301*  4302   4303*
UDC746   001  14FC 4397    4254   4324
UDC747   001  0078 4403    4399
UDC749   001  1500 4404    4158*  4161   4187   4251*  4267   4276   4287   4400
UDC750   120  1577 4405    4249*  4250   4250*
UDC751   001  157C 4412    4188   4189   4410
UDC752   120  15F3 4413    4193*  4194   4194*
UDC753   001  0059 4431    4409
UDC754   001  1578 4407    4340
UDC810   051  1626 4439    4251
UDC814   002  1628 4443    4334
UDC817   001  1629 4444    4224   4232   4380   4383   4384
UDC818   002  162B 4445    4151   4233   4336
UDC820   006  1631 4446    4289   4300   4309
UDC830   008  1639 4447    4190
UDC840   006  163F 4448    4191
UDC850   006  1645 4449    4192
UDC880   001  1646 4451    4346   4960
UDC884   001  164A 4457    4454
```

```
UDC885   024  16C1 4461
UDC886   001  16C2 4463   4352   4949   4974
UDC900   002  16C7 4469   4200
UDC901   002  16C9 4470   4268
UDC902   001  16CA 4471   4375
UDDB04   001  0004 4907   4932   4932   4985
UDDB17   001  0011 4904   4913
UDDB27   001  001B 4906   4915
UDDB76   001  004C 4905   4914
UDDDPL   001  1B3E 4999   4953   5007
UDDL01   001  0078 5018   5014
UDDM01   001  1B46 5012   4944
UDD100   004  1AD4 4931   5009
UDD110   001  1ADE 4936   4989
UDD400   005  1B29 4985   4980
UDD410   004  1B2E 4986   4988
UDD700   001  1B3B 4993   4985
UDD710   002  1B3D 4994   4986
UDD800   001  1B40 5007   4930*  4939   4986*  4987
UDD801   001  0001 5006   5002
UDD820   002  1B45 5010   4929*  4979*
UDD901   002  1A3D 4908   4979
UDD910   017  1A4E 4913   4940
UDD919   001  1A3E 4912   5015
UDD920   027  1AB5 4915   4932   4932*  4985*
UDUMPC   006  12F4 4158   3938
UDUMPD   004  1AB6 4922   3939   4928
UDUMP1   004  12F0 4157   4967
UPADDR   002  1948 4768   4671*  4706
UPADRR   002  1963 4790   4707
UPATCH   001  1860 4665   4657   4788
UPBADC   004  1956 4785   4695   4701   4703
UPBLNK   001  0040 4650   4728
UPBUF    002  1961 4789   4707*  4718   4721*
UPCHCK   004  187C 4683   4684
UPDATA   001  1949 4769   4720*  4726   4728   4730   4732*  4734
UPEVEN   003  18C3 4710   4742
UPGDTA   004  1874 4681   4664   4666
UPGOTC   001  0009 4643
UPKEY    001  0007 4649
UPLOOP   001  18D6 4716   4744
UPQNN    001  0003 4647   4743
UPQNZ    001  0001 4645   4715
UPQZN    001  0002 4646
UPREAD   001  0011 4648
UPRETN   004  1928 4749   4727
UPRTNC   001  004E 4651
UPSNSQ   001  0011 4653
UPSTS    001  194A 4770
UPTCNT   002  194C 4771   4681   4681*  4740*  4741
UPTCON   003  18B9 4706   4693
UPTDSK   001  1A37 4887
UPTD1    002  1950 4777   4740   4857
UPTD2    002  1952 4778   4855
UPTD4    002  1954 4779
UPTD9    001  1955 4780   4732
```

```
SYMBOL   LEN VALUE DEFN   REFERENCES                         VER 15, MOD 00  13/01/22  PAGE  99

UPTHC0   001  194E 4776
UPTH39   001  194D 4775
UPTLP0   003  188C 4692   4705
UPTLP1   003  18B3 4704   4697   4699
UPTL02   001  0029 4762   4796   4800
UPTL03   001  0001 4763   4806
UPTM02   001  198D 4798   4675
UPTM03   001  1991 4804   4786
UPTM04   001  1996 4811   4750
UPT001   001  0001 4644   4704   4710   4719
UPT200   003  18D3 4715   4708   4713
UPT210   004  18E6 4722   4711*  4712   4717*
UPT215   004  1900 4733   4731
UPT220   004  1904 4734   4715*  4743*
UPT225   004  1910 4737   4729
UPT230   004  1918 4740   4736
UPT250   004  1941 4760   4753
UPT810   002  1946 4767
UPT920   001  1964 4794
UPT930   001  1964 4795   4801
UPT940   001  1995 4809   4807
UVMADD   001  1100 3425   3402   3410
UVMCVB   002  0FAB 3435   3335
UVMDP1   001  0F80 3398   3386
UVMDP2   001  0F86 3406   3349
UVMDR1   001  0000 3421   3400
UVMDR2   001  0005 3422   3408
UVMEND   001  0F08 3342   5106
UVMG00   001  1107 3419   3391
UVMLNL   001  000D 3420   3358   3359   3431   3432
UVMLN1   013  0F9C 3431   3358
UVMLN2   013  0FA9 3432   3359
UVML01   001  000E 3423   3401   3409
UVML02   001  000D 3424
UVMZDU   002  0F8F 3430   3381
UVMZUD   002  0F8D 3429   3345
UVM100   002  0F05 3337   3368*
UVM101   002  0F07 3338   3377*
ZAPP06   001  1AE7 4941
ZAPP07   001  1AED 4946
ZAPP08   001  1AFF 4957
ZAPP09   001  1B17 4970
ZAPP10   001  1B17 4971
ZCDBN6   001  0006 3912   3918   3923
ZCDB10   001  000A 3913   3920   3925
ZCDCHK   013  120D 3933   3811
ZCDCMP   002  1214 3940   3812
ZCDCM3   002  1218 3942   3786
ZCDCM4   002  121A 3943   3774
ZCDCOP   002  1216 3941   3799
ZCDCRD   002  1210 3938   3825
ZCDDMP   002  1212 3939   3839
ZCDDPL   013  11E6 3930   3772
ZCDDSK   001  120E 3934   3826*  3840*  4489   4517
ZCDEND   013  11CC 3928   3824
ZCDFFF   001  00FF 3909   3826   3840   4489
```

```
                                CROSS REFERENCE

SYMBOL  LEN VALUE DEFN   REFERENCES                    VER 15, MOD 00  13/01/22  PAGE 100

ZCDFOR  001 0004 3907   3858  3881  4660
ZCDLL2  001 0002 3908   3773
ZCDL01  001 001D 3905   3950
ZCDL02  001 001D 3906   3957
ZCDMM1  001 1179 3917   3951
ZCDMM2  001 1196 3922   3958
ZCDMSG  001 000D 3910   3771  3772  3785  3797  3798  3810  3811  3823  3824  3837  3838  3919
                       3924  3927  3928  3929  3930  3931  3932  3933
ZCDMV1  013 118B 3919   3358* 3771* 3797* 3810* 3823* 3837* 5079*
ZCDMV2  013 11A8 3924   3359* 3772* 3785* 3798* 3811* 3824* 3838*
ZCDM01  001 121B 3948   3362  3847  4923  5082
ZCDM02  001 121F 3955   3371  3870
ZCDRDA  013 11F3 3931   3797  3810  3838
ZCDSEC  013 11D9 3929   3837  5079
ZCDSTA  013 11BF 3927   3785  3823
ZCDWRT  013 1200 3932   3771  3798
ZCD005  001 1092 3767   3087
ZCD006  001 10A9 3781   3081
ZCD020  004 1114 3846   3766  3768  3775  3780  3782  3792  3794  3800  3805  3807  3813  3818
                       3820  3827  3832  3834  3852  3854  3863
ZCD030  003 112C 3858
ZCD040  001 1136 3861   3858*
ZCD050  002 1138 3862   3859*
ZCD060  004 1141 3869   3787  3841  3875  3877  3886  4838
ZCD062  003 1158 3881   3773* 4660*
ZCD070  001 1162 3884   3881*
ZCD080  002 1164 3885   3882*
ZCD087  004 116D 3891   3774* 3786* 3799* 3812* 3825* 3839*
ZCD090  002 1172 3892   3864* 4854  5055  5077
ZCD100  002 1174 3893   3887* 4659  4659* 4671  4930  5056  5078
ZCORED  001 10E3 3819   3079
ZCPADD  002 1A36 4886   4839
ZCPCNT  001 0001 4882   4878
ZCPDPL  001 1A2D 4874   4756  4832* 4868
ZCPDSK  001 19F3 4852   4890
ZCPDS1  004 1A0D 4859   4856
ZCPEST  001 199A 4821   4733  4737
ZCPFFF  001 00FF 4883   4712  4752  4827  4833  4853
ZCPFTS  004 19A7 4881   4668* 4831*
ZCPF00  001 00F0 4884   4835
ZCPRST  001 19E7 4845   4735  4738
ZCPSVA  002 1A34 4885   4825* 4848
ZCPTST  004 1A03 4856   4662* 4714*
ZCPUCS  002 1A3B 4890
ZCPWRT  004 1A1D 4867   4829
ZCPX12  004 19E3 4842   4822* 4834
ZCPX22  004 19EF 4850   4846*
ZCPZCS  002 1A39 4889   4656  4661
ZCP010  004 19A6 4826   4881
ZCP020  004 19B5 4830   4656* 4661* 4826  4872
ZCP025  004 19B9 4831   4864
ZCP030  004 19C3 4833   4828
ZCP040  006 19D9 4839   4836
ZCSADD  002 12C3 4057   4020  4022  4025  4029  4494* 4824* 4827  4839* 4840
ZCSAVE  001 1223 3984   4496  4889
ZCSCTR  002 12C6 4060   4029* 4030* 4032*
```

```
                                 CROSS REFERENCE

SYMBOL  LEN VALUE DEFN    REFERENCES                         VER 15, MOD 00  13/01/22  PAGE 101

ZCSDAT  001 1F00 4073    4086   4879   5003
ZCSDPD  001 12D1 4088    4037*  4855*  4857*
ZCSDPL  001 12CF 4082    4040   4088   4832   4860
ZCSEND  002 12C8 4061    4003*  4008*  4009*  4011*  4022
ZCSFFF  001 00FF 4074    4004   4008   4019   4837
ZCSHGH  002 12C0 4055    3996*  3998*  4003   4004   4006   4025
ZCSHUN  002 12CA 4078    3998   4032
ZCSLOW  002 12BE 4054    3995*  3996   4020   4030
ZCSOFF  001 000F 4065    4006   4009   4024
ZCSONE  002 12CE 4080    3999   4034
ZCSOUT  001 12B5 4044    4021   4026
ZCSOU0  004 12B5 4046    3988*
ZCSOU2  004 12B9 4047    3990*
ZCSREL  001 12C4 4059    4019*  4024*  4027*  4498   4501   4752   4833   4835   4837*  4853*
ZCSSCT  001 12C1 4056    3997*  3999*  4031   4031*  4034*  4037
ZCSSEC  001 0001 4072    4085
ZCSTHO  002 12CC 4079    4011
ZCSTWO  001 1267 4012    3993   4007
ZCS010  001 1223 3987    3986   3989
ZCS020  003 122E 3993    3344*  3994*
ZCS030  004 1242 3998    4000
ZCS040  003 125D 4008    4005
ZCS050  004 127B 4025    4023
ZCS060  004 1297 4032    4035
ZCS070  004 12A5 4037    4033
ZCTADD  002 177C 4531    4503
ZCTDSK  006 174B 4503    4490
ZCTEST  001 170A 4483    4219   4283   4311   4331
ZCTFFF  001 00FF 4527    4180   4492   4498   4517
ZCTFTS  004 171E 4529    4180*  4495*
ZCTF00  001 00F0 4528    4501
ZCTRDK  004 1772 4524    4518
ZCTRST  001 175D 4514    4227   4240   4288   4313   4333
ZCTSVA  001 177A 4530    4499*  4519
ZCTX10  004 1751 4505    4484*  4500
ZCTX12  004 1759 4507    4486*
ZCTX22  004 176E 4521    4515*  4525
ZCT010  004 171D 4491    4529
ZCT020  006 1728 4494    4491
ZCT030  004 1736 4498    4493
ZDCADV  001 1BCD 5104    5128   5161
ZDCBLA  004 1C78 5169    5147   5148   5149   5150   5151   5152
ZDCBN4  001 0004 5178    5122   5149
ZDCBUM  004 1C80 5171    5126
ZDCCMP  006 1C01 5123    5122*  5126*  5127   5129   5138   5139   5140
ZDCCNT  002 1CAC 5177    5088*  5105*
ZDCCOP  001 1B4A 5052    3941
ZDCCPL  001 0100 5237    5101
ZDCC01  001 1E00 5233    5189
ZDCC02  001 1F00 5234    5196   5203
ZDCDA1  001 0000 5230    5187
ZDCDA2  001 0000 5231    5194   5201
ZDCDL1  001 0001 5232    5123   5188   5195   5202
ZDCDPL  001 1CAD 5185    5077*  5091   5107*  5108   5136
ZDCDP2  001 1CB3 5192    5055*  5059   5078*  5097   5110*  5111   5137
ZDCDP3  001 1CB9 5199    5056*  5065
```

```
                              CROSS REFERENCE

SYMBOL   LEN VALUE DEFN   REFERENCES                          VER 15, MOD 00  13/01/22  PAGE 102

ZDCDT1   001  1EFF 5235   5101
ZDCDT2   001  1FFF 5236   5101
ZDCENT   001  1B74 5073   3940
ZDCEN2   004  1B8B 5081   5087
ZDCERR   001  1BF1 5114   5051  5053  5072  5074  5102
ZDCFOR   002  1C82 5172   5107  5110
ZDCL01   001  001B 5222   5208  5214  5224
ZDCM01   001  1CC7 5223   5209
ZDCOMP   001  10CF 3806   3089
ZDCONE   002  1C74 5168   5105
ZDCPL1   001  1CBF 5206   5119
ZDCPP2   001  1CC3 5212   5155
ZDCPRA   001  1C83 5173   5146  5215  5240  5241  5242  5243  5244  5245
ZDCPRT   001  1C1E 5130   5124
ZDCPR1   001  1C84 5240   5147*
ZDCPR2   001  1C8A 5241   5148*
ZDCPR3   001  1C92 5242   5149*
ZDCPR4   001  1C96 5243   5150*
ZDCPR5   001  1C9A 5244   5151*
ZDCPR6   001  1C9D 5245   5152*
ZDCPTR   004  1C7C 5170   5122
ZDCSIX   001  0060 5238   5108  5111
ZDCTWO   001  0020 5239   5127
ZDCWKA   001  1C9E 5175   5145  5246  5247  5248  5249  5250
ZDCWK1   001  1CA0 5246   5136*
ZDCWK2   001  1CA3 5247   5137*
ZDCWK3   001  1CA6 5248   5138*
ZDCWK4   001  1CA8 5249   5141*
ZDCWK5   001  1CAA 5250   5142*
ZDC010   004  1BA7 5090   5113
ZDC020   005  1BD5 5107   5109
ZDC030   005  1BE1 5110   5112
ZDC040   003  1BF1 5115   5076* 5116*
ZDC050   001  1BFD 5121   5115
ZDC055   004  1C0F 5126
ZDC060   005  1C38 5141   5139*
ZDC070   005  1C3D 5142   5140*
ZDMVM    001  0F28 3357   3097
ZDUMDK   001  10FB 3833   3085
ZDV001   004  0F34 3361   3367
ZDV002   004  0F4C 3370   3376
ZOTX11   004  1755 4506   4485*
ZUTBLK   001  0DA2 3238   3216
ZUTCOP   001  10BA 3793   3091
ZUTIRI   001  0DA5 3252   3063  3114  3172  4262  4330  4343  4349  4355  4685  4966  5084  5103
                         5125
ZUTKER   001  0D99 3232   3070  3174* 3176* 3851  3874
ZUTKEY   001  0D31 3161   3069  3365  3374  3850  3873  4926  5085
ZUTKR0   004  0D5D 3179   3164*
ZUTKR2   004  0D61 3180   3165*
ZUTLBC   001  1100 3319
ZUTLBD   001  2008 3317
ZUTLBL   001  0002 3318
ZUTLBM   001  0E28 3322   3315
ZUTLGO   001  0F6A 3382
ZUTLIB   004  0E22 3314   3101
```

```
                                   CROSS REFERENCE

SYMBOL   LEN VALUE DEFN    REFERENCES                          VER 15, MOD 00  13/01/22  PAGE 103


ZUTLMA   001  0DA1 3237    3032* 3131   3140
ZUTMAR   001  0DA0 3236    3040* 3141
ZUTMRR   001  0082 3246    3041
ZUTPRR   002  0D9D 3234    3042* 3142
ZUTPRT   002  0D9B 3233    3043
ZUTRET   004  0DB5 3257    3253*
ZUTTAB   032  1709 4473
ZUTTFL   001  0CC8 3116    3099
ZUTTF1   004  0CD7 3122    3118
ZUTTIP   002  0D9F 3235    3131* 3132*
ZUT010   001  0C07 3028
ZUT012   004  0C15 3035    5368
ZUT015   004  0C1B 3039
ZUT020   004  0C39 3049    3115   3256   3353   3897   4760   4981   5069
ZUT021   003  0C6F 3082    3077
ZUT022   003  0C91 3092    3083
ZUT030   004  0CB4 3106    3071
ZUT031   006  0CDB 3123    3095   3120
ZUT033   004  0CE1 3125    3093
ZUT034   006  0CED 3132    3139
ZUT035   006  0D00 3140    3133
ZUT036   004  0D16 3147
ZUT200   001  0D31 3163
ZUT210   004  0D41 3170    3171
ZUT220   001  0D5D 3177    3175
ZUT510   001  0D65 3192    3055
ZUT530   001  0D69 3199    3107
ZUT540   001  0D6D 3206    3036   3050   3126
ZUT541   001  0D71 3213    3136
ZUT700   001  0D75 3223    3195
ZUT710   001  0D98 3225    3202
ZUT750   002  0DA4 3239    3123
ZUT800   001  0023 3222    3194   3224
ZUT810   001  0001 3243    3201   3215
ZUT820   001  00FF 3244    3070   3174   3176   3851   3874
ZUT830   001  0000 3245
ZUT900   004  1D52 5363    3033
ZUT950   004  1D60 5366    5364


TOTAL STATEMENTS IN ERROR IN THIS ASSEMBLY =      0

OL105 I   THE CODE LENGTH OF #ZUTMO IS  7534 DECIMAL.
OL103 I   TOTAL NUMBER OF LIBRARY SECTORS REQUIRED IS    23
          NAME-#ZUTMO,PACK-R1R1R1,UNIT-R1,RETAIN-P,LIBRARY-R,CATEGORY-000
```

```
         OVERLAY LINKAGE EDITOR CORE USAGE MAP     13/01/22


   START      CATEGORY   NAME AND       CODE LENGTH
  ADDRESS                  ENTRY      HEXADECIMAL DECIMAL

  0C00          0      #ZUTMO           1D6E     7534


OL100 I   THE TOTAL CORE USED BY #ZUTMO IS  7534 DECIMAL.
OL101 I   THE START CONTROL ADDRESS OF THIS MODULE IS 0C00.
OL104 I   TOTAL NUMBER OF LIBRARY SECTORS REQUIRED IS   30
          NAME-#ZUTMO,PACK-R1R1R1,UNIT-R1,RETAIN-P,LIBRARY-O
```