

```

1 * GENERAL AUTOMATION, INC. ALL RIGHTS RESERVED
2 *****
3 *
4 * PROGRAM NAME FPH-27
5 *
6 * MODEL NUMBER 8F027
7 *
8 * PURPOSE FORTRAN PHASE-27
9 *
10 * PROGRAMMER DICK WALLMANN
11 *
12 ***** REVISION LIST *****
13 *
14 * RV DATE SCO BY REASON FOR CHANGE
15 * -- -----
16 *
17 * 01 11/16/70 NONE RPH INITIAL RELEASE
18 *
19 *****
20 *****
21 HDNG MPX FORTRAN ** OUTPUT II
22 *****
23 *STATUS-VERSION 1, MODIFICATION 0
24 *
25 *FUNCTION/OPERATION-
26 * * TESTS FOR EXTENDED PRECISION AND CHANGES
27 * THE NAME CODES FOR THE EFFECTED SUBROUTINES
28 * TO REFLECT THIS CONDITION
29 * * CONVERTS THE COMPILED STATEMENT STRING TO
30 * OUTPUT CODE AND STORES IT IN THE BUFFER AREA
31 *
32 *INPUT-
33 * * THE STATEMENT STRING
34 * * THE SYMBOL TABLE
35 * * THE FORTRAN COMMUNICATIONS AREA
36 *
37 *OUTPUT-
38 * * THE COMPILED PROGRAM IN DISK SYSTEM FORMAT
39 * INCLUDING REAL AND INTEGER CONSTANTS, DATA
40 * STATEMENTS AND DEFINE FILES.
41 *
42 *EXTERNAL REFERENCES-
43 * * SUBROUTINES-
44 * ROLRX-INTERPHASE ROLLING ROUTINE
45 *
46 *EXITS-
47 * * NORMAL-
48 * PHASE 28 IS CALLED VIA THE ROLRX
49 * ROUTINE AND CONTROL IS PASSED TO IT
50 * * ERRORS-
51 * OVERLAP-
52 * IF A PREVIOUS PHASE HAS DETECTED AN
53 * OVERLAP ERROR THEN AN IMMEDIATE EXIT
54 * IS PERFORMED. OTHERWISE, NO OVERLAP
55 * ERROR IS DETECTED.
56 * SYNTAX-
57 * NO SYNTAX ERRORS ARE DETECTED
58 *
59 *TABLES/WORK AREAS

```

```

60 * * THE STATEMENT STRING
61 * * THE SYMBOL TABLE
62 * * THE FORTRAN COMMUNICATIONS AREA
63 *
64 64 ABS REF CORE
65 *
66 * SYSTEM AND FORTRAN EQUATES
67 *
68 MEMRY EQU 17FFF CORE MAXIMUM CORE SIZE
69 PHSIZ EQU 4*320 MAXIMUM PHASE SIZE
70 OVERL EQU MEMRY-PHSIZ PHASES 2-29 START
71 FCOM EQU OVERL-22 FORTRAN COMM. TABLE
72 PHNTB EQU FCOM-56 PHASE TABLE
73 RCLRX EQU PHNTB-50 INTERPHASE CALL
74 LIO EQU 185 IOCS ENTRY
75 *
76 * FORTRAN COMMUNICATIONS AREA
77 *
78 ORG FCOM
79 SOFS BSS 1 START OF STRING
80 EOFS BSS 1 END OF STRING
81 SCFST BSS 1 START OF SYMBOL TABLE
82 SOFNS BSS 1 PROG LENGTH AT OBJ TIME
83 SOFXT BSS 1 SIZE OF WORK AREA
84 SOFGT BSS 1 SIZE OF CONSTANT AREA
85 EOFST BSS 1 END OF SYMBOL TABLE
86 COMON BSS 1 ENTRY POINT
87 CSIZE BSS 1 SIZE OF COMMON
88 ERROR BSS 1 OVERLAP ERROR
89 FNAME BSS 1 PROGRAM NAME
90 BSS 1 2ND WORD PROG NAME
91 SORF BSS 1 SUBR (-) OR FUNC (+)
92 CCWD BSS 1 CONTROL CARD WORD
93 * BIT 15 TRANSFER TRACE
94 * BIT 14 ARITHMETIC TRACE
95 * BIT 13 EXTENDED PRECISION
96 * BIT 12 LIST SYMBOL TABLE
97 * BIT 11 LIST SUBPROGRAM NAMES
98 * BIT 10 LIST SOURCE PROGRAM
99 * BIT 9 ONE WORD INTEGERS
100 IOCS BSS 1 IOCS CONTROL CARD WORD
101 *
102 * SEE PHASE ONE FOR BIT PATTERNS
103 *
104 DFCNT BSS 1 DEFINE FILE COUNT
105 *
106 LCOMN BSS 2 INSKEL COMMON
107 *
108 ICCER BSS 2 IOCS CONTROL CARD ERROR
109 *
110 BSS 2 SYSTEM LOADER USE
111 *
112 * END OF FORTRAN COMMUNICATION
113 * AREA
114 *****
115 *
116 * BUFFER COMMUNICATIONS AREA
117 * AND BUFFER
118 *
119 ORG OVERL+3*320

```

```

120          BSS      E
121  BUFPT EQU      *          BUFFER POINTER
122  INPT  EQU      BUFPT+1    INDICATOR POINTER
123  INCT  EQU      INPT+1     INDICATOR COUNTER
124  LOCTR EQU      INCT+1     LOCATION COUNTER
125  RECCT EQU      LOCTR+1    RECORD COUNT
126  BUFF  EQU      RECCT+1    OUTPUT BUFFER
127  WD1   EQU      BUFF+1
128  WD2   EQU      BUFF+2
129  WD3   EQU      BUFF+3
130  WD4   EQU      BUFF+4
131  WD5   EQU      BUFF+5
132  WD6   EQU      BUFF+6
133  WD7   EQU      BUFF+7
134  WD8   EQU      BUFF+8
135  WD9   EQU      BUFF+9
136  WD10  EQU      BUFF+10
137  *
138  *
139          ORG      OVERL
140  NEQ    BSC      L  ENT      ENTRANCE
141  *
142  *          INITIALIZTRANSFER VECTOR POINTER
143  START LDX      L3 ZERO    SET POINTER TO TABLE
144  *
145  *****
146  *
147  *  EXAMINE CONT RECORDS FOR UDISK AND EXT PREC
148  *
149  *****
150  *
151          LD      L  IOCS      GET IOCS CONTROL RECORD
152          SRA      14          CHECK FOR UDISK
153          BSI      L  UTOY,E    BR IF ODD (BIT 15 ON=UDISK)
154          LD      L  CCWD      LOAD CONTROL CARD WORD
155          SLA      13          SHIFT EXTENDED PREC INDR
156          BSC      L  Q1011,-  BR IF NOT EXTENDED PREC
157  *
158  *          CHANGE SYSTEM SUBROUTINE NAMES
159  *          SO THAT APPLICABLE NAMES BEGIN
160  *          WITH E RATHER THAN WITH F
161  *
162  *          CHANGE SUBROUTINE NAMES
163  *          FAXIX THROUGH FADD
164  *
165          LDX      1 FAXIX-TABSS 2
166  Q1005 BSI      FTOE      CHANGE NEXT NAME
167          MDX      1 -2      DECR COUNT ON NAMES
168          MDX      Q1005    BR IF INDEX NOT ZERO
169  *
170          LD      3 H0004-Z    DIFF IN SECOND CHAR
171          STO      DIFF      BETWEEN E AND F TO CHG NAME
172  *
173  *          CHANGE NAMES OF TRACE-ROUTINES
174  *          TO HAVE SECOND CHARACTER  E
175  *          INSTEAD OF  F
176  *
177          LDX      1 FARI-TABSS 2
178          BSI      FTOE      BR TO CHANGE MFAR TO MEAR
179          LDX      1 FARIX-TABSS 2

```

```

180      BSI      FTOE      BR TO CHANGE MFARX TO MEARX
181      LDX      1 FIF-TABSS 2
182      BSI      FTOE      BR TO CHANGE SFIX TO SEIX
183      MDX      Q1011     BR PAST SUBROUTINE
184      *
185      *
186      *          SUBROUTINE
187      *          CHANGE .F. IN 1ST OR 2ND CHARACTER 0
188      *          NAME INTO .E.
189      *          OR E INTO V ETC
190      *          DISPLACEMENT IN NAME LIST   X1
191      *
192      FTOE DC      0          LINK ENTRY POINT
193      LD      L1 TABSS-2     LOAD ENTRY FR TBL PLTV XR1
194      S        DIFF         SUBTRACT CHARACTER DIFF
195      STO     L1 TABSS-2     SAVE NEW NAME
196      BSC     I  FTOE         RETURN
197      *****
198      *
199      *          SUBROUTINE TO CHANGE U TO Y IN FIO CALLS
200      *
201      *****
202      *
203      UTOY DC      *-*          LINK ADDR WORD
204      STX     1 XR1SV         SAVE XR1
205      LDX     L1 UCOMP-URED+2 GET A TABLE LENGTH
206      UTOYL LD     L1 URED-2    GET A TABLE ENTRY
207      A        H0400         CHANGE THE U TO A Y
208      STO     L1 URED-2     PUT IT BACK IN THE TABLE
209      MDX     1 -2          MOVE THE POINTER
210      MDX     UTOYL         LOOP BACK TO DO IT AGAIN
211      *
212      LDX     L1 UIOI-UFIO+2 GET ANOTHER TBL LENGTH
213      UTOYM LD     L1 UFIO-2   LOAD AN ENTRY
214      A        H0400         CHANGE THE U TO A Y
215      STO     L1 UFIO-2     PUT IT BACK IN THE TABLE
216      MDX     1 -2          MOVE THE POINTER
217      MDX     UTOYM         LOOP BACK
218      LDX     L1 *-*          RESTORE XR1 UPON FALL THRU
219      XR1SV EQU   *-1        XR1 SAVED HERE FROM ABOVE
220      BSC     I  UTOY         EXIT THE UTOY ROUTINE
221      *
222      *
223      * * * * *
224      *          CONSTANTS
225      DIFF DC      /0100     FOR CHANGING F TO E
226      H0400 DC    /0400     FOR CHANGING U TO Y
227      *
228      *****
229      *
230      Q201A LDX   I1 SOFS     INITIALIZE INPUT POINTER
231      Q1011 EQU   Q201A
232      *
233      LDX     L3 ZERO        RESTORE CONSTANT POINTER
234      *
235      MDX     1 -1
236      Q2012 BSI   3 INIT-Z    INITIALIZE NEXT STATEMENT
237      LD      3 STOIC-Z     LOAD STORED ID TYPE
238      *
239      *          SPOT FOR TEST STOP

```

```

240          SLA      0          NOP
241 *
242          S        3 IDFOR-Z  TEST FOR FORMAT STATEMENT
243          BSC     L  Q2013, - BR IF FORMAT
244 *
245          S        3 IDDEF-Z  TEST FOR DEFINE FILE STMNT
246          BSC     L  Q2101, - BR IF DEFINE FILE
247          S        3 IDDAT-Z  TEST FOR DATA STATEMENT
248          BSC     L  Q2101, - BR IF DATA STATEMENT
249          S        3 IDCAL-Z  TEST FOR CALL LINK/EXIT
250          BSC     L  Q2201, - BR IF CALL LINK/EXIT
251          MDX      Q3011      BR IF NOT ABOVE STMNT TYPES
252 *
253 *          OUTPUT FORMAT STMT
254 Q2013 BSI      3 INCPT-Z  INCR INPUT POINTER
255 Q2014 LD       1 0          LOAD STRING WORD
256          BSI      3 OUABS-Z  SUBR OUTPUT, ENTRY  ABS
257          LD       3 WCNT-Z  TEST WORD COUNT
258          BSC     L  Q2013,Z- BR IF WORD COUNT IS POSITIV
259          MDX      Q2012      BR IF COUNTER IS ZERO
260 *
261 *          MOVE POINTER WITHOUT OUTPUT
262 *          DATA, DEFINE FILE HAVE ALREADY
263 *          BEEN OUPUT
264 *
265 Q2101 SLA      0          NOP
266          BSI      3 INCPT-Z  BR TO INCR I/P POINTER
267          LD       3 WCNT-Z  LOAD WORD COUNT
268          BSC     L  Q2101,Z- BR IF WORD COUNT IS POSITIV
269          MDX      Q2012      BR IF ZERO  END OF STMNT
270 *
271 *          SPECIAL OUTPUT OF
272 *          CALL LINK OR CALL EXIT
273 *
274 *
275 *          TEST IF CALLLINK OR CALLEXIT
276 *
277 Q2201 LD       1 1          LOAD DATA WORD
278          S        3 H5B00-Z  TEST CALL LINK FLAG
279          BSC     L  Q2213,Z  BR IF NOT CALL LINK
280 *
281 *          OUTPUT CALL LINK
282 *
283 Q2211 LDD      LINKN      GET TRUNCATED EBC NAME
284          BSI      3 OULCL-Z  GO OUTPUT IT
285 *
286 *          OUTPUT .CALL XXXXX.
287 *
288          LD       1 4          GET 2ND WD OF CALL
289          RTE      15         PUT BITS 1 THRU 15 IN Q
290          LD       1 2          GET 1ST WD OF CALL
291          SLT      1          EXPUNGE BIT 0
292          SRT      2          RIGHT JUSTIFY A AND Q
293 Q2212 BSI      3 OULCL-Z  GO OUTPUT CALL
294          MDX      Q2101      BR BACK, CONTINUE OUTPUT
295 *
296 *          OUTPUT CALL EXIT
297 *
298 Q2213 LDD      EXITN      CALL .EXIT. NAME
299          MDX      Q2212      BR TO A BR

```

```

300          BSS E
301 *
302 *      EXIT AND LINK IN TRUNCATED EBCDIC
303 *
304 LINKN DC      /1325      CONSTANT
305          DC      /5480      CONSTANT
306 EXITN DC      /059C      CONSTANT
307          DC      /98C0      CONSTANT
308 *
309 *
310 *
311 Q3011 LD      3 WCNT-Z      LOAD WORD COUNTER
312          BSC      Z-        SKIP IF WORDCOUNTER NOT POS
313          MDX      Q3013      BR IF POSITIVE
314          LD      3 STOIC-Z   LOAD STORED ID WORD TYPE
315          S      3 IDEND-Z   TEST IF END STATEMENT
316          BSC      Z        SKIP IF STORED ID .END.
317          MDX      Q2012      BR IF NOT
318 *
319          BSC L OUTER      TERMINATE OUTPUT
320 *
321 *
322 Q3013 BSI      3 INCPT-Z     INCR INPUT POINTER
323 Q3014 LD      1 0          LOAD WORD FROM I/P STRING
324          BSC      -        SKIP IF .PACKED. WORD
325          MDX      Q4011      BR IF NOT
326 Q3021 LD      1 0          LOAD WORD FROM I/P STRING
327          AND      3 H7800-Z  GET INDEX TO INSTRUCTION TB
328          BSC L Q3023, -    BR IF NO INSTR INCLUDED
329 *                          WORD CONSISTS OF ADDR ONLY
330          SRA      11        RIGHT JUSTIFY INDEX
331          A      3 CON4B-Z    4-BIT-CONV-TABLE ORIGIN
332          STO      Q3022 1    COMPUTE ADDR OF INST IN TBL
333 Q3022 LD      L *-*        LOAD INST FROM TBL TAB4
334          BSI      3 OUABS-Z  SUBR OUTPUT, ENTRY ABS
335 Q3023 LD      1 0          LOAD WORD FROM I/P STRING
336          BSI      3 GETST-Z  GET OBJ TIME ADDR FR SYM TB
337          BSI      3 OUREL-Z  SUBR OUTPUT, ENTRY RLTV
338          MDX      Q3011      CONTINUE PROCESSING
339 *
340 *      TEST IF SPECIAL BSC ONE WHICH IS
341 *      FOLLOWED BY A RELATIVE ADDRESS .
342 *
343 Q4011 S      3 H5000-Z      .SPEC BSC L.
344          BSC      -        SKIP IF NOT .SPEC BSC L.
345          MDX      Q4021      BR IF SPEC BSC L
346 *
347          S      3 HFF00-Z    .BSC I1. - .SPEC BSC L.
348          BSC      Z        SKIP IF .BSC I1.
349          MDX      Q4031      BR IF NOT
350          LD      3 H4D80-Z    OBJECT TIME .BSC I1. TO ACC
351          BSI      3 OUABS-Z  SUBR OUTPUT, ENTRY ABS
352          BSI      3 INCPT-Z  INCREMENT INPUT POINTER
353          LD      L LOGTR     LOCATION COUNTER
354 *
355 Q4012 BSI      3 OURNC-Z     SUBR OUTPUT, ENTRY RLTV
356          MDX      Q3011      BR TO CONT OR TERMINATE
357 *
358 *      SPECIAL BSC L ENCOUNTERED
359 *      IS FOLLOWED BY A RELATIVE ADDRESS

```

```

360 *           RATHER THAN BY A SYMBOL TABLE NAME
361 *
362 Q4021 LD     3 H4C08-Z  OBJ TIME .BSC L. TO ACC
363         BSI     3 OUABS-Z  SUBR OUTPUT, ENTRY  ABS
364         BSI     3 INCPT-Z  INCR INPUT POINTER
365         LD      1 0        OUTPUT RLTV ADDR FR STRING
366         MDX     Q4012     BR TO OUTPUT ROUTINE
367 *
368 *           TEST IF .LDX L1.
369 *           WHICH IS FOLLOWED BY AN ABSOLUTE
370 *           VALUE AND OUTPUT .LDX L1. FOLLOWED
371 *           BY THE ABSOLUTE VALUE
372 *
373 Q4031 LD     1 0        LOAD WORD FROM I/P STRING
374         S       3 H5B00-Z  SUBTRACT CODE FOR .LDX L1.
375         BSC     Z        SKIP IF .LDX L1.
376         MDX     Q4041     BR IF NOT
377         LD     L LDXL1   OUTPUT .LDX L1.
378         BSI     3 OUABS-Z  OUTPUT IN ABS MODE
379         SLA     16       CLEAR ACC
380         S       1 1      CHANGE SIGN OF STRING WD
381         BSI     3 OUABS-Z  OUTPUT FOLLOWING WD IN ABS
382         BSI     3 INCPT-Z  INCR INPUT POINTER
383         MDX     Q3011     CONTINUE THROUGH STRING
384 *
385 *           TEST IF .BSC I.
386 *           WHICH REQUIRES OUTPUT OF ENTRY ADDR
387 *
388 Q4041 LD     1 0        LOAD WORD FROM I/P STRING
389         S       3 H5080-Z  SUBTRACT .BSC I. CONSTANT
390         BSC     L Q5011,Z  BR IF NOT BSC I
391         LD     3 H4C80-Z  OUTPUT .BSC I.
392         BSI     3 OUABS-Z  IN ABSOLUTE MODE
393         LD     L COMON    OUTPUT ENTRY ADDR
394 Q4042 BSI     3 OURNC-Z   ADDR OF OBJECT TIME LINK W
395         BSI     3 INCPT-Z  INCREMENT INPUT POINTER
396         MDX     Q3011     CONTINUE
397 *
398 *           TEST IF TWO-WORD CALL OPERATOR
399 Q5011 LD     1 0        LOAD WORD FROM INPUT STRING
400         AND     3 HFF80-Z  TEST BITS 0-8
401         BSC     Z        SKIP IF .CALL SUBPROGR.
402         MDX     Q7011     BR IF NOT
403 *
404 *           TWO-WORD CALL
405         LD     1 1      LOAD SECOND WORD OF CALL
406         BSI     3 GETST-Z  GET OBJ TIME ADDR OR NAME
407         BSI     3 OULCL-Z  SUBR O/P, ENTRY  2-WD CALL
408         BSI     3 INCPT-Z  INCREMENT INPUT COUNTER
409         MDX     Q3011     CONTINUE THROUGH STRING
410 *
411 *           TEST IF SYST SUBROUTINE CALL
412 Q7011 LD     1 0        LOAD STRING WORD
413         SLA     1        TEST BIT 1 FOR CALL
414         BSC     Z        SKIP IF SYST.CALL
415         MDX     Q7021     BR IF NOT
416 *
417 *           OUTPUT SYSTEM SUBROUTINE CALL
418 *           ONE-WORD CALL
419 *

```

```

420      SRA      7      SHIFT POINTER SYST SUBR TBL
421      A        3      SSTOR-Z COMPUTE ADDRESS IN SYSTEM
422      STO      Q7012 1  SUBROUTINE TABLE AND INSERT
423      *
424      Q7012 LD   L    *-*      LOAD WD2 FROM SYST SUBR TBL
425      RTE      16      PUT IN EXTENSION
426      MDX      L    Q7012 1,-1 DECR ADDR SYSTEM SUBR TBL
427      SLA      0      NOP
428      LD       I    Q7012 1  LOAD WD1 FROM SYST SUBR TBL
429      BSI      3      OUCAL-Z  SUBR.OUTPUT,ENTRY  CALL
430      *
431      *
432      *      TEST WHETHER OUTPUT SUBR CALL WAS
433      *      .SUBSC. OR .SUBIN.
434      *      .FIOAF. OR .FIOAI.
435      *      ARGUMENTS OF THESE CALLS
436      *      REQUIRE SPECIAL TREATMENT
437      *      SUBROUTINES HAVE BOTH ABSOLUTE
438      *      AND RELATIVE ARGUMENTS
439      LD       1 0      LOAD WORD FROM STRING
440      S        3 H1D00-Z TEST FOR .CALL SUBIN.
441      BSC      L    Q7051, - BR IF .CALL SUBIN.
442      LD       1 0      LOAD WORD FROM STRING
443      S        3 CSUBS-Z TEST FOR .CALL SUBSC.
444      BSC      L    Q8011, - BR IF .CALL SUBSC.
445      LD       1 0      LOAD WORD FROM STRING
446      S        3 M1600-Z TEST FOR .CALL SIOAI.
447      BSC      Z      SKIP IF .CALL SIOAI.
448      S        3 M1680-Z TEST FOR .CALL SIOAF.
449      BSC      Z      SKIP IF EITHER SPECIAL CALL
450      S        3 M2400-Z TEST FOR .CALL SDAF.
451      BSC      Z      SKIP IF EITHER SPECIAL CALL
452      S        3 M2380-Z TEST FOR .CALL SDAI.
453      BSC      Z      SKIP IF EITHER SPECIAL CALL
454      S        3 M1E00-Z TEST FOR .CALL UIOAI.
455      BSC      Z      SKIP IF EITHER SPECIAL CALL
456      S        3 M1E80-Z TEST FOR .CALL UIOAF.
457      BSC      L    Q8051, - BR IF ANY OF ABOVE FOUND
458      *
459      *      NO SPECIAL TREATMENT REQ.
460      *      BSC      L    Q3011 BR TO NEXT ITEM ON STRING
461      *
462      *      NO SYSTEM SUBROUTINE CALLS FOUND
463      *      TEST NEXT FOR .STX L1. CODE
464      Q7021 LD   1 0      LOAD WORD FROM STRING
465      S        3 H6200-Z TEST FOR .STX L1.
466      BSC      L    Q7031, - BR IF STX L1
467      *
468      *      NO SPECIAL TREATMENT REQUIRED
469      *      CONVERT ONE WORD OF PRE-OUTPUT CODE
470      *      INTO OBJECT TIME-INSTRUCTION,
471      *      OUTPUT .ABSOLUTE.
472      *
473      *      COMPUTE 6-BIT CONVERSION
474      LD       1 0      LOAD WORD FROM STRING
475      AND      3 H3F00-Z GET BITS 2-7 PLTV TBL ADDR
476      SRA      8      RIGHT JUSTIFY
477      A        3 CON48-Z ADD TO ADDR CONVERSION TBL
478      STO      Q7022 1  TO COMPUTE ADDR TO LOAD
479      LD       1 0      LOAD STRING WORD

```



```

480          AND    3 H00FF-Z  GET OUTPUT STORAGE ADDR
481 Q7022 A      L  *-*      TABLE VALUE
482          BSI    3 0UABS-Z  SUBR OUTPUT, ENTRY  ABS
483          BSC    L  Q3011    BR TO NEXT ITEM ON STRING
484 *
485 *
486 Q7031 LD      L  STXL1     OUTPUT .STX L1.
487          BSI    3 0UABS-Z  IN ABSOLUTE MODE
488          BSI    3 INCPT-Z  INCR INPUT COUNTER
489          LD      1 0        LOAD STRING WORD
490          AND    3 H7FFF-Z  OUTPUT *  DISPLACEMENT
491          A      L  LOCTR    INCR BY LOC COUNTER
492          STO    1 0        STORE FOR DUMP  F TEST
493          BSI    3 0URN-C  PUT OUT RELATIVE OUTPUT
494          BSC    L  Q3011    CONTINUE THROUGH STRING
495 *
496 *
497 Q7051 BSI    3 INCPT-Z  INCR INPUT POINTER
498          LD      1 0        LOAD STRING WORD
499          BSC    L  Q7055, Z BR IF  TAGGED  ADDRESS
500 *
501 *
502          BSI    3 0UABS-Z  IN ABSOLUTE MODE
503          LD      1 0        LOAD STRING WORD
504          BSC    L  Q7051, - BR IF  DELIMETER  ZERO
505          SRA    1          SHIFT OFF LOW ORDER BIT
506          BSC    L  Q3011, - BR IF  ONE
507 *
508          WAIT   3          PAUSE, PROGRAM
509          MDX    *-2        ERROR IF NEITHER
510 *
511 Q7055 AND    3 H7FFF-Z  REMOVE .TAG.
512          BSI    3 0URN-C  OUTPUT RELATIVE
513          *
514          MDX    Q7051     BR TO OUTPUT ARGUMENTS
515 *
516 *
517          *
518 Q8011 LD      1 1        LOAD STRING WORD  1
519          BSI    3 GETST-Z  GET OBJ TIME ADDR OF SGT
520          BSI    3 0UREL-Z  SUBR OUTPUT, ENTRY  RLTV
521          LD      1 2        ACC STRING WORD  2  04
522          BSI    3 0UABS-Z  SUBR OUTPUT, ENTRY  ABS
523          BSI    3 INCPT-Z  INCR I/P POINTER  1 TIME
524          BSI    3 INCPT-Z  INCR I/P POINTER  2 TIMES
525          BSI    3 INCPT-Z  INCR I/P POINTER  3 TIMES
526 Q8021 LD      1 0        LOAD STRING WORD
527          BSI    3 GETST-Z  GET OBJ TIME ADDR
528          BSI    3 0UREL-Z  SUBR OUTPUT, ENTRY  RLTV
529          LD      1 1        D1 OR D2 OR D3  TC ACC
530          BSI    3 0UABS-Z  SUBR OUTPUT, ENTRY  ABS
531          BSI    3 INCPT-Z  INCR I/P POINTER  1 TIME
532          BSI    3 INCPT-Z  INCR I/P POINTER  2 TIMES
533          LD      1 -1       LOAD PREVIOUS WORD
534          BSC    L  Q3014, Z BR IF  PREVIOUS O/P TAGGED
535          MDX    Q8021     BR IF NOT
536 *
537 *
538          *
539          *
          OUTPUT ARGUMENTS OF
          .CALL FIOAX. OR .CALL FIOIX.

```

```

540 Q8051 LD      1 1      LOAD 2ND STRING WORD
541      BSI     3 GETST-Z  GET OBJECT TIME ADDR
542      BSI     3 OUREL-Z  SUBR OUTPUT, ENTRY  RLTV
543      LD      1 2      ARRAY SIZE
544      AND     3 H7FFF-Z  REMOVE SIGN
545      BSI     3 OUABS-Z  SUBR OUTPUT, ENTRY  ABS
546      BSI     3 INCPT-Z  INCR INPUT POINTER
547      BSI     3 INCPT-Z  INCR INPUT POINTER AGAIN
548      BSC     L Q3011    BR TO PROCESS NEXT WORD
549 *
550 *
551 * WRITE BINARY DATA
552 *
553 WRITE DC      *-*      ENTRY POINT
554      LDX     3 54
555      SLA     17        CLEAR CARRY
556      LD      L RECCT   RECORD COUNT
557 WR01  A      L3 BUFF   FORM CHECKSUM
558      BSC     C
559      A       ONE
560      MDX     3 -1
561      MDX     WR01      LOOP
562      RCP     4,4
563      RIC     4,4      MAKE 2*S COMPLIMENT
564      STO     L WD2     CHECKSUM
565      BSI     L LIO     WRITE BINARY RECORD
566      DC      /2104
567      DC      BUFF
568      DC      0
569      BSI     L LIO     WAIT FOR TRANSFER
570      DC      /F004
571      LDX     3 53
572      SLA     17
573 WR02  STO     L3 WD1   CLEAR BUFFER
574      MDX     3 -1
575      MDX     WR02      LOOP
576      LDX     L3 Z      RESTORE Z POINTER
577      LD      3 CD16-Z  -16
578      STO     L INCT    CNT.-BIT COUNT = 06
579      LDD     WR03
580      STO     L BUFPT   BUFPT=WD10, INPT=WD4
581      LD      L LOCTR
582      STO     L WD1     NEW LOCATION COUNTER
583      LD      WR03+2    /A00
584      STO     L WD3     DATA RECORD TYPE
585      MDX     L RECCT,1 INCREMENT RECORD COUNT
586      BSC     I WRITE   EXIT
587      BSS     E 0
588 WR03  DC      WD10    DATA START
589      DC      WD4      CONTROL START
590      DC      /A00
591 *
592 *
593 *
594 DC20  DC      20      CONSTANT
595 DC36  DC      36      CONSTANT
596 TEN   DC      10      CONSTANT
597 *
598 M1600 DC      /1600    SIOAI
599 M1680 DC      /1680-1600 SIOAF-SIOAI

```

600	H1D00	DC	/1D00	CONSTANT MASK	
601	H7FFF	DC	/7FFF	CONSTANT MASK	
602	H5B00	DC	/5B00	.LDX L1.	
603	*				
604	CD16	DC	/0010	.ONE. IN WORDCOUNT, -12	
605	*			* ALSO INITIAL VALUE	
606	*				
607	TEMP1	DC	0	TEMPORARY STORAGE	
608	TEMP2	DC	0	TEMPORARY STORAGE	
609	H2000	DC	/2000	MASK FOR COMMON	
610	H4C80	DC	/4C80	CONSTANT MASK	
611	H5080	DC	/5080	CONSTANT MASK	
612	H6200	DC	/6200	CONSTANT MASK	
613	*				
614	ZERO	DC	/0000	CONST ZERO	0
615	Z	EQU	ZERO		
616	*				
617	*		ZERO	ORIGIN OF TRANSVECTOR	
618	ONE	DC	/0001	CONST ONE	
619	TWO	DC	/0002	CONST TWO	
620	THREE	DC	/0003	CONST THREE	
621	FOUR	DC	/0004	CONST FOUR	
622	H0004	EQU	FOUR		
623	IDFOR	DC	/6000	ID .FORMAT.	7
624	IDEND	DC	/1000	ID .END.	9
625	H7800	DC	/7800	CONSTANT MASK	
626	CON48	DC	TAB4-1	CONV-TABL-ORIGIN	
627	H5000	DC	/5000	.SPEC BSC L.	12
628	H4D80	DC	/4D80	OBJ.TIME BSC L1	
629	HFF00	DC	/FF00	.BSC I1. - .SPECBSC.	14
630	BFEND	DC	BUFF+55	BUFFER END	
631	H4C08	DC	/4C08	OBJ.TIME .BSC L1.	17
632	CSUBS	DC	/1900	CALL SUBSCRIPT	
633	SSTOR	DC	TABSS-1	SYSTEM SUBR TABLE ORIGIN	
634	H3F00	DC	/3F00	MASK F CONVERSION	
635	H00FF	DC	/00FF	CONSTANT MASK	
636	H07FF	DC	/07FF	CONSTANT MASK	
637	HF800	DC	/F800	CONSTANT MASK	
638	C320	DC	320	CONSTANT	
639	HFF80	DC	/FF80	CONSTANT MASK	
640	H0005	DC	/0005	CONSTANT MASK	
641	H07FC	DC	/07FC	CONSTANT MASK	
642	M2400	DC	/2400-/1680	SDAF-SIOAF CONSTANT	
643	M2380	DC	/2380-/2400	SDAI-SDAF CONSTANT	
644	M1E00	DC	/1E00-/2380	UIOAI-SDAI CONSTANT	
645	M1E80	DC	/1E80-/1E00	UIOAF-UIOAI CONSTANT	
646	C16	DC	16	CONSTANT	
647	SWISH	DC	0	ANOTHER CALL INDICATOR	
648	*				
649		BSS	E 0		
650	STONA	DC	0	STORED NAME	
651		DC	0		
652	WCNT	DC	0	WORD COUNTER F. STMT	8
653	STOID	DC	0	STORED ID-TYPE	6
654	TWOWC	DC	0	TWO WORD CALL SWITCH	
655	*				
656	IDDEF	DC	/F000-/6000	DEFINE FILE CONSTANT	
657	IDDAT	DC	/F800-/F000	DATA CONSTANT ID	
658	IDCAL	DC	/E000-/F800	CALL CONSTANT ID	
659	H4000	DC	/4000	CONSTANT MASK	

```

660 *
661 *          SUBROUTINE
662 *          INCREMENT INPUT POINTER
663 *
664 INCPT DC      0          LINK
665          MDX    1 1      INCR INPUT POINTER
666          MDX    L WCNT,-4  DECR WD COUNTER
667          SLA    0          GUARDS AGAINST SKIP
668          BSC    I INCPT    RETURN
669 *
670 *          SUBROUTINE
671 *          GET OBJ.TIME ADDR OR ALPHABETIC
672 *          NAME FROM SYMBOL TABLE
673 *          ON ENTRY, THE ACCUMULATOR CONTAINS
674 *          THE SYMBOL TABLE REFERENCE WORD
675 *
676 GETST DC      0          LINK ENTRY POINT
677          AND    3 H07FF-Z  GET RLTV ENTRY PT IN SYM TB
678          STO    GET1 1    SET UP AS ADDR
679          LD     L SOFST    LOAD START OF SYMBOL TABLE
680          S      GET1 1    SUBTRACT POINTER ONE TIME
681          S      GET1 1    SUBTRACT POINTER SECOND TIM
682          S      GET1 1    SUBTRACT POINTER THIRD TIME
683          A      3 H0005-Z  ADD 5 TO GIVE LOC OF 3RD WD
684          STO    GET1 1    SAVE SYMBOL TABLE ADDR WD 3
685 GET1 LD     L *-*        LOAD 3RD WORD SYMBOL TABLE
686          RTE    16        PUT IN EXTENSION
687          MDX    L GET1 1,-1  DECR SYMBOL TABLE POINTER
688          SLA    0          GUARDS AGAINST SKIP
689          LD     I GET1 1    LOAD 2ND SYM TBL WD IN ACC
690          BSC    I GETST    RETURN
691 *
692 *          SUBROUTINE
693 *          INITIALIZE NEXT STATEMENT
694 *
695 INIT  DC      0          LINK ENTRY POINT
696          BSI    3 INCPT-Z  INCR INPUT COUNTER
697          LD     1 0        LOAD STRING WORD
698          AND    3 H07FC-Z  EXTRACT AND STORE
699          S      3 FOUR-Z   NORM - 1 WORD
700          STO    3 WCNT-Z   WDCOUNTER
701          LD     1 0        LOAD STRING WORD
702          AND    3 HF800-Z  MASK ALL BUT TYPE CODE ID W
703          STO    3 STOIO-Z  SAVE TYPE CODE
704          BSC    I INIT     RETURN
705 *
706 *
707 *          OUTPUT ADDRESS OR CONSTANT IN
708 *          ABSOLUTE MODE
709 *          ENTRY  ABSOLUTE OUTPUT
710 *
711 OUABS DC      0          LINK ENTRY POINT
712          MDX    T8011     BR TO OUTPUT IN ABSOLUTE
713 *
714 *          OUTPUT ADDRESS OR CONSTANT IN
715 *          RELOCATABLE MODE EXCEPT FOR VARIABLE
716 *          IN COMMON WHICH WILL BE ABSOLUTE MOD
717 *          ENTRY  RELATIVE OUTPUT
718 *          WITH .COMMON-CHECK.
719 *

```

720	OUREL DC	0	LINK ENTRY POINT	
721	MDX	T8013	BR TO OUTPUT IN RLTV MODE	
722	*	ENTRY	REALTIVE OUTPUT	
723	*	NO	.COMMON-CHECK.	
724	*			
725	OURNC DC	0	LINK ENTRY POINT	
726	MDX	T7051	BR TO OUTPUT IN RLTV MODE	
727	*			
728	*	ENTRY	CALL-OUTPUT	
729	*	ONE WORD	CALL	
730	*			
731	OUCAL DC	0	LINK ENTRY POINT	
732	MDX	T7011	BR TO OUTPUT ONE WD CALLS	
733	*			
734	*	ENTRY FOR	TWO-WORD-CALL-OUTPUT	
735	OULCL DC	0	LINK ENTRY POINT	
736	T7005 STD	3	STONA-Z	STORE ACC AND EXTENSION
737	LD	3	ZERO-Z	SECOND HALF CALL BITS
738	STO	3	TEMP2-Z	SAVE IN TEMP STORAGE
739	LD	3	ZERO-Z	LOAD ZERO IND 2WD SW
740	STO	3	TWOWC-Z	SET 2 WD CALL SWITCH
741	LD		OULCL	MOVE LINKWORD
742	T7006 STD		OUCAL	SAVE IN RETURN WD AREA
743	LD	3	THREE-Z	LOAD INDICATOR BITS
744	STO	3	TEMP1-Z	SAVE IN TEMPORARY STORAGE
745	MDX		T7013	BR TO MOVE BITS
746	T7011 STD	3	STONA-Z	STORE REGISTERS
747	LD	3	ZERO-Z	LOAD 2ND HALF CALL BITS 0
748	STO	3	TEMP2-Z	SAVE IN TEMP STO
749	LD	3	TWO-Z	LOAD 1ST HALF CALL BITS 2
750	STO	3	TEMP1-Z	SAVE IN TEMP STO
751	STO	3	TWOWC-Z	TWO WD CALL SW SET TO NO
752	T7013 STD	3	SWISH-Z	SET ANOTHER CALL INDICATOR
753	*	TEST	IF ONE SPACE LEFT IN BUFFER	
754	LD	L	BUFPT	
755	S	3	BFEND-Z	
756	RIC		4,4	
757	BSC		-	
758	BSI	3	WRITE-Z	WRITE IF ONE WORD LEFT
759	LD	3	TEMP1-Z	CALL BITS 10 OR 11
760	BSI		IBITS	TO INDICATOR AREA
761	LD	3	STONA-Z	FIRST HALF OF NAME
762	BSI		TOBUF	MOVE TO BUFFER
763	LD	3	TEMP2-Z	SEC HALF OF CALL BITS
764	*		00 OR 11	
765	BSI		IBITS	TO INDICATOR AREA
766	*		CHECK TWO WORD CALL SWITCH	
767	LD	3	TWOWC-Z	LOAD 2 WD CALL SWITCH
768	BSC	L	T7014, -	BRANCH IF 2 WORD CALL
769	*		COUNT ONE LOCATION LESS IF	
770	*		IT IS A ONE-WORD CALL	
771	MDX	L	LOCTR,-1	DECR LOCATION COUNTER
772	NOP		POSSIBLE SIGN CHANGE	
773	T7014 LD	3	STONA 1-Z	SECOND HALF OF NAME
774	BSI		TOBUF	MOVE TO BUFFER
775	BSC	I	OUCAL	RETURN
776	*			
777	*		RELATIVE OUTPUT, NO COMMON CHECK	
778	*			
779	T7051 STD	3	STONA-Z	STORE REGISTERS

780	LD	OURNC	LOAD LINK WORD PREV ENTRY P
781	STO	OUREL	MOVE TO EXIT POINT
782	MDX	T8015	BR TO PROCESS RELATIVE O/P
783	*		
784	*		ABSOLUTE OUTPUT
785	*		
786	T8011	STD 3	STONA-Z STORE REGISTERS
787	LD	3	ZERO-Z ZERO-ZERO
788	BSI		IBITS TO INDICATOR AREA
789	*		
790	*		MOVE STORED ACCUMULATOR TO BUFFER
791	*		
792	T8012	LD 3	STONA-Z LOAD OUTPUT WORD
793	BSI		TOBUF MOVE TO BUFFER
794	*		
795	BSC	I	OUABS RETURN
796	*		
797	*		RELATIVE OUTPUT INCLUDES TEST
798	*		WHETHER VARIABLE IS IN COMMON, IN
799	*		WHICH CASE OUTPUT IS ABSOLUTE
800	*		
801	T8013	STD 3	STONA-Z STORE REGISTERS
802	LD	L	GET1 1 GET SYMBOL TABLE ADDRESS
803	S	3	ONE-Z MODIFY TO GET SYM TBL ID WD
804	STO		T8014 1 SAVE ADDR OF SYM TBL ID WD
805	T8014	LD L	*-* LOAD SYM TBL ID WORD
806	AND	3	H2000-Z MASK TO GET COMMON BIT ONLY
807	BSC	L	T8021,Z BR IF IN COMMON
808	*		
809	T8015	LD 3	ONE-Z ZERO-ONE FOR INDICATOR BITS
810	T8016	BSI	IBITS MOVE TO INDICATOR AREA
811	LD		OUREL MOVE LINK TO RETURN EXIT
812	STO		OUABS SAVE LINK WD
813	MDX		T8012 BR TO O/P WD IN RLTV MODE
814	T8021	LD 3	STONA-Z GET NAME
815	STO	3	STONA+1-Z MOVE NAME ONE WORD
816	LD	3	STONA-Z GET NAME
817	BSC	L	T8022,- GO TO IF VARIABLE
818	*		
819	*		PREPARE FOR BLANK COMMON
820	*		
821	LD	3	THREE-Z ELSE GET A 3 IN ACC
822	MDX		T8023 BR TO CONTINUE
823	*		
824	*		PREPARE FOR LABELED COMMON
825	*		
826	T8022	LD 3	STONA-Z ADJ HIGH LIMIT OF LAB COM
827	S	3	H4000-Z MODIFY IT
828	STO	3	STONA-Z STORE IT IN TWO WORDS
829	STO	3	STONA+1-Z WORD TWO
830	LD	3	TWO-Z GET A TWO IN ACC
831	T8023	STO 3	TEMP2-Z PUT IT IN TEMP2
832	STX	L0	TWOWC SET SWITCH ON (TWO WD COM)
833	LD		OUREL TO MOVE LINK
834	MDX		T7006 CONTINUE
835	*		
836	*		INDICATOR BITS
837	*		ACCUMULATOR CONTAINS BIT PATTERNS
838	*		TO MASK INTO THE INDICATOR POINTER
839	*		BASED ON SHIFTS SPECIFIED IN THE

```

840 *          INDICATOR COUNTER
841 *
842 IBITS DC      0          LINK ENTRY POINT
843         STO    IBIT5     SAVE INDICATOR BITS
844         LD     L INCT     PRODUCE SLA AND SRA
845         A      IBIT6     INSTRUCTIONS AND INSERT
846         STO    IBIT3     LENGTH OF SHIFT DEPENDS
847         A      IBIT7     ON INDICATOR COUNTER
848         STO    IBIT2     SET UP SRA INSTRUCTION
849         LD     I INPT     LOAD WORD FROM BUFFER
850 IBIT2 SRA     *-*       SRA INCT - 2
851         A      IBIT5     VALUE FROM ACC ON ENTRY
852 IBIT3 SLA     *-*       SLA INCT - 2
853         STO    I INPT     SAVE WORD IN BUFFER
854         MDX   L INCT,-2   DECR INDICATOR COUNTER
855         MDX   IBIT4     BR IF NOT ZERO NO SKIP
856         MDX   L INPT,1   INCR. INDICATOR PNTR.
857         LD     3 CD16-Z   REINITIALIZE INDR COUNTER
858         STO    L INCT
859 IBIT4 BSC     I IBITS    RETURN
860 IBIT5 DC      0          TEMPORARY STORAGE
861 IBIT6 DC      /0FFE     .SLA 0. - 0002
862 IBIT7 DC      /0800     .SRA 0. - .SLA 0.
863 *
864 *          SUBROUTINE
865 *          MOVE TO BUFFER
866 *
867 TOBUF DC      0          LINK ENTRY POINT
868         STO    I BUFPT    SAVE I/P WD IN BUFR POINTER
869         MDX   L LOCTR,1   INC. LOCATION COUNTER
870         MDX   L BUFPT,1   INCR. BUFFER PNTR.
871         MDX   L WD3,1     INCREMENT DATA COUNT
872 *          TEST IF END OF BUFFER
873         LD     L BUFPT
874         S      3 BFEND-Z   COMPARE TO LAST WD OF BUFFE
875         BSC    -          BUFFER NOT FULL
876         BSI   3 WRITE-Z   WRITE RECORD
877         BSC    I TOBUF    RETURN
878 *
879 *          TERMINATE OUTPUT
880 OUTER SLA     0          NOP
881 *
882 *          MAKE PROGRLENGTH EVEN NUMBER
883         LD     L LOCTR    LOAD LOCATION COUNTER
884         BSC    E          SKIP IF EVEN
885         MDX   *+1        ODD VALUE
886         MDX   *+2        EVEN LAST VALUE
887         BSI   L QUABS    DUMMY CELL
888         LD     L WD3
889         AND   3 H00FF-Z   GET DATA WORD COUNT
890         BSC    Z          NO OUTPUT IF EMPTY
891         BSI   3 WRITE-Z   WRITE LAST DATA
892         LD     OUTC1     /0F00
893         STO    L WD3     SET EOP FLAG
894         LD     L SORF
895         BSC    L OUTX1,Z  SUBROUTIN/FUNCTION
896         LD     L COMON
897         STO    L WD4     MAIN PROGRAM ENTRY
898 OUTX1 BSI     3 WRITE-Z   OUTPUT EOP RECORD
899         BSC    L EXIT    TO RECOVERY PHASE

```

900	*				
901	OUTC1	DC	/0F00		
902	*				
903	*			.4-BIT.-OBJ. TIME INSTRUCTION TABLE	
904	*				
905	TAB4	DC	/C400	LD L	CODE 1
906		DC	/D400	STO L	2
907		DC	/8400	A L	3
908		DC	/9400	S L	4
909		DC	/AC00	D L	5
910		DC	/A400	M L	6
911	BSIL	DC	/4400	BSI L	7
912		DC	/6580	LDX I1	8
913		DC	/C500	LD L1	9
914		DC	/D500	STO L1	10
915		DC	/8500	A L1	11
916		DC	/9500	S L1	12
917		DC	/AD00	D L1	13
918		DC	/A500	M L1	14
919	*				
920	BSCL	DC	/4C00	BSC L	15
921	*			USED FOR UNCOND BRANCH ONLY	
922	*				
923	*			EXTENSION	
924	*			.6-BIT. OBJ. TIME INSTRUCTION TABLE	
925	*				
926	TAB6	DC	/4C00	BSC L	CODE 16 5000
927		DC	/7400	MDX L	17 5100
928		LD	3 0	LD FAC 1	18 527E
929		STO	3 0	STO FAC 1	19 537E
930		S	3 0	S FAC 1	20 547E
931		D	3 0	D FAC 1	21 557E
932		STD	3 0	STD FAC 1	22 567E
933		SD	3 0	SD FAC 1	23 577E
934		SRT	16	SRT 16	24 5800
935		SLA	16	SLA 16	25 5900
936		DC	0	RES FOR *-1	26 5A00
937	LDXL1	DC	/6500	LDX L1	27 5B00
938		DC	/4D80	BSC I1	28 5C00
939	SBSCL	DC	/4C08	BSC L SPEC	29 5D00
940		DC	0	RES F GENLB	30 5E00
941		DC	0	CONST ZERO	31 5F00
942		MDX	* 0		32 6000
943		DC	/7500	MDX L1	33 6100
944	STXL1	DC	/6D00	STX L1	34 6200
945		LDX	0	LDX	35 6300
946		DC	0	WORK WORD	
947		DC	0	WORK WORD	
948		DC	0	WORK WORD	
949	*				
950	*				
951	*			SYSTEM SUBR. TABLE	
952	*				
953	TABSS	DC	/0604	FADD	CODE 008
954		DC	/4100		
955		DC	/0604	FADDX	010
956		DC	/4127		
957		DC	/068A	FSUB	018
958		DC	/4080		
959		DC	/068A	FSUBX	020

960	DC	/40A7		
961	DC	/0651	FMPY	028
962	DC	/7A00		
963	DC	/0651	FMPYX	030
964	DC	/7A27		
965	DC	/0610	FDIV	038
966	DC	/9940		
967	DC	/0610	FDIVX	040
968	DC	/9967		
969	DC	/064C	FLD	048
970	DC	/4000		
971	DC	/064C	FLDX	050
972	DC	/49C0		
973	DC	/068A	FSTO	058
974	DC	/3580		
975	DC	/068A	FSTOX	060
976	DC	/35A7		
977	DC	/0688	FSBR	068
978	DC	/2640		
979	DC	/0688	FSBRX	070
980	DC	/2667		
981	DC	/0612	FDVR	078
982	DC	/5640		
983	DC	/0612	FDVRX	080
984	DC	/5667		
985	DC	/0606	FAXI	088
986	DC	/7240		
987	FAXIX DC	/0606	FAXIX	090
988	DC	/7267		
989	DC	/0626	FIXI	098
990	DC	/7240		
991	DC	/0626	FIXIX	0A0
992	DC	/7267		
993	FARI DC	/1418	MFAR	0A8
994	DC	/1640		
995	FARIX DC	/1418	MFARX	0B0
996	DC	/1667		
997	FIAR DC	/1424	MIAR	0B8
998	DC	/1640		
999	FIARX DC	/1424	MIARX	0C0
1000	DC	/1667		
1001	AIFIX DC	/0918	IFIX	0C8
1002	DC	/99C0		
1003	AFLT DC	/064C	FLOAT	0D0
1004	DC	/6063		
1005	DC	/0359	COMGO	0D8
1006	DC	/41D6		
1007	DC	/098A	ISTOX	0E0
1008	DC	/35A7		
1009	FIIF DC	/1424	MIIF	0E8
1010	DC	/9180		
1011	FIF DC	/1418	MFIF	0F0
1012	DC	/9180		
1013	FGOTO DC	/141D	MGOTO	0F8
1014	DC	/68D6		
1015	DC	/1310	LDFAC	100
1016	DC	/6043		
1017	DC	/228C	STFAC	108
1018	DC	/6043		
1019	DC	/2208	SBFAC	110

1020		DC	/6043		
1021		DC	/0494	DVFAC	118
1022		DC	/6043		
1023	*				
1024	*				
1025	*				
1026	*				
1027	*				
1028	*				
1029	*				
1030	*				
1031	URED	DC	/2464	URED	120
1032		DC	/5100		
1033		DC	/2499	UWRT	128
1034		DC	/9800		
1035	UCOMP	DC	/2400	UCOMP	130
1036		DC	/6517		
1037		DC	/1914	READZ	138
1038		DC	/1129		
1039	AERD	DC	/1464	MRED	140
1040		DC	/5100		
1041		DC	/1499	MWRT	148
1042		DC	/9800		
1043		DC	/1400	MCOMP	150
1044		DC	/6517		
1045	AFIO	DC	/1418	MFIO	158
1046		DC	/9580		
1047		DC	/1425	MIOAI	160
1048		DC	/6049		
1049	AIOAF	DC	/1425	MIOAF	168
1050		DC	/6046		
1051		DC	/1425	MIOFX	170
1052		DC	/61A7		
1053		DC	/1425	MIOIX	178
1054		DC	/6267		
1055		DC	/1425	MIOF	180
1056		DC	/6180		
1057	IOI	DC	/1425	MIOI	188
1058		DC	/6240		
1059	SUBSC	DC	/2290	SUBSC	190
1060		DC	/2883		
1061		DC	/0000		198
1062		DC	/0000		
1063	BCKSP	DC	/0200	BCKSP	1A0
1064		DC	/2897		
1065	EOF	DC	/0558	EOF	1A8
1066		DC	/6000		
1067	REWND	DC	/1916	REWND	1B0
1068		DC	/6544		
1069		DC	/2280	STOP	1B8
1070		DC	/6500		
1071		DC	/1706	PAUSE	1C0
1072		DC	/4885		
1073	RSIGN	DC	/2255	SNR	1C8
1074		DC	/9000		
1075		DC	/2290	SUBIN	1D0
1076		DC	/2255		
1077	UFIO	DC	/2418	UFIO	1D8
1078		DC	/9580		
1079		DC	/2425	UIOAI	1E0

THE SUBR ENTRY NAMES FOR UNFORMATTED
I/O ARE LOCATED IN TWO GROUPS, THUS
ANY ONE HAS ONE OF TWO POSSIBLE
DISPLACEMENTS, WHICH ARE
/1200-/1400 OR
/1080-/1400

```

1080      DC      /6049
1081      DC      /2425      UIOAF      1E8
1082      DC      /6046
1083      DC      /2425      UIOFX      1F0
1084      DC      /61A7
1085      DC      /2425      UIOIX      1F8
1086      DC      /6267
1087      DC      /2425      UIOF      200
1088      DC      /6180
1089      UIOI   DC      /2425      UIOI      208
1090      DC      /6240
1091      MDFIO  DC      /1410      MDFIO      210
1092      DC      /6256
1093      MDRED  DC      /1411      MDRED      218
1094      DC      /9144
1095      DC      /1412      MDWRT      220
1096      DC      /6663
1097      DC      /1410      MDCOM      228
1098      DC      /3594
1099      DC      0          RESERVED TO KEEP CODE
1100      DC      0          DIFF CONSTANT
1101      *          BETWEEN DISK I/O
1102      *          ROUTINES AND OTHERS
1103      DC      /1410      MDAI      238
1104      DC      /1240
1105      DC      /1410      MDAF      240
1106      DC      /1180
1107      DC      /1410      MDFX      248
1108      DC      /69C0
1109      DC      /1410      MDIX      250
1110      DC      /99C0
1111      DC      /1410      MDF      258
1112      DC      /6000
1113      DC      /1410      MDI      260
1114      DC      /9000
1115      DC      /1410      MDFND     268
1116      DC      /6544
1117      DC      /0000      270
1118      DC      /0000
1119      ECTAB  DC      0          NOT USED
1120      DC      0
1121      *
1122      *
1123      *****
1124      *
1125      *          PROGRAM ENTRY
1126      *          TEST IF OUTPUT IS TO BE DEFERRED
1127      ENT    LD    L      ERROR
1128      BSC    L      START, - BR TO START IF NO ERROR
1129      *
1130      *          LOAD OVERLAY-RESTORE
1131      *
1132      EXIT   LDX    1 0
1133      STX    L1 SOFS      ZERO THE START OF STRING A
1134      BSI    L      ROLRX   CALL DOWN RECOVERY PHASE
1135      DC      28          NEXT PHASE NUMBER
1136      BSS    OVERL-++320*3  PHASE-27 PATCH AREA
1137      END    NEG

```