

```

1  *   GENERAL AUTOMATION, INC.  ALL RIGHTS RESERVED
2  *****
3  *
4  *   PROGRAM NAME   FPH10
5  *
6  *   MODEL NUMBER   8F010
7  *
8  *   PURPOSE        FORTRAN PHASE-10
9  *
10 *   PROGRAMMER     DICK WALLMANN, MODS-MARK ELFIELD
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 ** FORMAT
22 *****
23 *STATUS-VERSION 1, MODIFICATION 0
24 *
25 *FUNCTION/OPERATION-
26 *   * CONVERTS FORMAT STATEMENTS INTO A CHAIN OF
*****
*****   ACHTUNG DOPPEL FORMATKARTEN *****
29 *   FORMAT SPECIFICATIONS FOR INTERPERTATION BY
*   FORMAT SPECIFICATIONS FOR INTERPERTATION BY
32 *   THE FORTRAN I/O SUBROUTINE,
33 *   * CHECKS THAT ALL FORMAT STATEMENTS HAVE A
34 *   STATEMENT NUMBER,
35 *   * CHECKS FORMAT STATEMENTS FOR SYNTAX AND
36 *   VALID STATEMENT TYPES,
37 *   * CONVERTS QUOTE TYPE FORMAT TO H TYPE,
38 *
39 *ENTRY POINTS
40 *   * START-PHASE 10 IS READ INTO CORE BY PHASE 9
41 *   VIA A CALL TO THE ROLRX ROUTINE,
42 *   EXECUTION IS BEGUN AT LOCATION START
43 *
44 *INPUT-
45 *   * STRING AREA
46 *   * SYMBOL TABLE AREA
47 *   * FCOM
48 *
49 *OUTPUT-
50 *   * STRING AREA
51 *   * SYMBOL TABLE AREA
52 *   * FCOM
53 *
54 *EXTERNAL REFERENCES
55 *   SUBROUTINES-
56 *   * ROLRX
57 *   OTHER FORTRAN PHASES
58 *   * NONE
59 *

```

```

60 *EXITS-
61 *   NORMAL-
62 *     * A CALL TO ROLRX IS ISSUED TO LOAD THE
63 *       FOLLOWING PHASE,
64 *
65 *   ERROR-
66 *     * OVERLAP ERROR CAUSES PHASE TO STOP
67 *       PROCESSING AND AN EXIT TO THE NEXT PHASE
68 *     * COMPILATION ERRORS DETECTED BY THIS PHASE
69 *       ARE 27, 28, 29, 30
70 *
71 *TABLES/WORK AREAS-
72 *   * FCOM
73 *   * SYMBOL TABLE
74 *   * STRING AREA
75 *
76 *ATTRIBUTES-N/A
77 *
78 *NOTES-
79 *   THE SWITCHES USED IN THIS PHASE FOLLOW
80 *   IF NON-ZERO, THE SWITCH IS TRANSFER T,
81 *   IF ZERO, THE SWITCH IS NORMAL N,
82 *   SW1-BEGINNING OF SPECIFICATION
83 *     N BEGINNING OF SPECIFICATION
84 *   SW2-I TYPE SPECIFICATION
85 *     T I TYPE SPECIFICATION
86 *   SW3-SECOND NUMBER OF SPECIFICATION
87 *     T SECOND NUMBER OF SPECIFICATION
88 *   SW4-LEFT/RIGHT SW FOR INPUT DATA GET SU3R
89 *     N LEFT
90 *   SW5-END OF STMT SW GET SUBR
91 *     T END OF STATEMENT
92 *   SW6-COMMA ALLOWABLE
93 *     N COMMA ALLOWABLE
94 *   SW7-LEFT/RIGHT SW FOR HOLL O/P SPEC
95 *     N LEFT
96 *   SW8-COMMA MANDATORY
97 *     T COMMA NOT MANDATORY
98 *   SW9-REDO OUTPUT
99 *     T OK TO PUT OUT REDO COUNT
100 *   SW11-H OR QUOTE TYPE SPECIFICATION
101 *     T H OR QUOTE TYPE SPECIFICATION
102 *   SW12-QUOTE TYPE SPECIFICATION
103 *     T QUOTE TYPE SPECIFICATION
104 *   SW15-DIGIT ENCOUNTERED
105 *     T DIGIT ENCOUNTERED
106 *   SW16-INCOMPLETE SPECIFICATION
107 *     T INCOMPLETE SPECIFICATION
108 *
109 *****
110 HDNG MPX FORTRAN ** FORMAT
111 ABS REF CORE
112 *
113 *   SYSTEM AND FORTRAN EQUATES
114 *
115 MEMRY EQU 17777 CORE MAXIMUM CORE SIZE
116 PHSIZ EQU 4*320 MAXIMUM PHASE SIZE
117 OVERL EQU MEMRY-PHSIZ PHASES 2-29 START
118 FCOM EQU OVERL-22 FORTRAN COMM. TABLE
119 PHNTB EQU FCOM-56 PHASE TABLE

```

```

120 ROLRX EQU      PHNTB-50      INTERPHASE CALL
121 *
122 *      FORTRAN COMMUNICATION AREA
123 *
124 *
125 *      FORTRAN COMMUNICATION AREA
126          ORG      FCOM      FORTRAN COMM AREA
127 SOFS  BSS      1      START OF STRING
128 EOFS  BSS      1      END OF STRING
129 SOFST BSS      1      START OF SYMBOL TABLE
130 SOFNS BSS      1      START OF NON-STMNT NOS,
131 SOFXT BSS      1      START OF SUBSC TEMPS
132 SOFGT BSS      1      START OF GENERATED TEMPS
133 EOFST BSS      1      END OF SYMBOL TABLE
134 COMON BSS      1      NEXT AVAILABLE COMMON
135 CSIZE BSS      1      SIZE OF COMMON
136 ERROR BSS      1      OVERLAP ERROR
137 FNAME BSS      1      PROGRAM NAME WD 1
138          BSS      1      PROGRAM NAME WD 2
139 SORF  BSS      1      SUBR (-) OR FUNC (+)
140 CCWD  BSS      1      CONTROL CARD WORD
141 *
142 *      BIT 15 TRANSFER TRACE
143 *      BIT 14 ARITHMETIC TRACE
144 *      BIT 13 EXTENDED PRECISION
145 *      BIT 12 LIST SYMBOL TABLE
146 *      BIT 11 LIST SUBPROGRAM NAMES
147 *      BIT 10 LIST SOURCE PROGRAM
148 *      BIT 9  ONE WORD INTEGERS
148 IOCS  BSS      1      IOCS CONTROL CARD WORD
149 *
150 *      SEE PHASE ONE FOR BIT PATTERNS
151 *
152 DFCNT BSS      1      DEFINE FILE COUNT
153 *
154 LCOMN BSS      2      SIZE OF INSKEL COMMON
155 *
156 ICCER BSS      2      IOCS CONTROL CARD ERROR
157 *
158          BSS      2      SYSTEM LOADER USE
159 *
160 *      END OF FORTRAN COMMUNICATION
161 *      AREA
162 *****
163          HDNG      MPX FORTRAN ** FORMAT
164          ORG      OVERL      PHASE ENTRY
165 START LD      ERROR      LD OVERLAP ERROR FLAG
166          BSC  L  LXQ, -    BR IF NO ERROR
167 *
168 *
169 OUT  BSI  L  ROLRX      CALL DOWN PHASE 11
170          DC      11      NEXT PHASE NUMBER
171 *
172 LXQ  LDX  I2 EOFST      MOVE STRING TO SYMBOL TABLE
173          LDX  I1 EOFS      GET POINTER TO O/P STRING
174          LD   EOFS      GET POINTER TO I/P STRING
175          S    SOFS      COMPUTE NUMBER OF WORDS IN
176          STO  XX  1      *THE STRING-1
177          XX  LDX  L3 0      SAVE STRING LENGTH-1
178          *      PUT STRING LENGTH-1 IN XR3
179          MDX  3 1      *MODIFIABLE
179          *      INCR STRING LENGTH-1 BY 1

```

```

180 LOOPP LD      1 0      LD WD FROM I/P STRING
181      STO     2 -1     SAVE IN O/P STRING
182      MDX     1 -1     DECR I/P POINTER
183      MDX     2 -1     DECR O/P POINTER
184      MDX     3 -1     DECR STRING LENGTH COUNT
185      MDX     LOOPP    LOOP UNLESS FINISHED
186      STX     2 ORIG 1  SAVE START LOC O/P STRING
187      LDX     12 SOFS   GET START LOC I/P STRING
188      STX     2 NEOFS   SAVE TEMPORARILY
189 *
190 *           INITIALIZE PHASE
191 ORIG  LDX     L1 0     PREV O/P STRING PT NOW I/P
192 *           *MODIFIABLE
193 *           END STATEMENT
194 ABEL  LD      1 0      GET ID WD FR I/P STRING
195      SRA     1        SHIFT OUT LOW ORDER BIT
196      AND     IDTPE    MASK IN ID TYPE BITS 1-5
197      S       ENDC     TEST FOR END STMT
198      BSC     L CBEL,Z  BR IF NOT END
199      LD      1 0      MOVE END STMT FR I/P
200      STO     2 0      *STRING TO O/P STRING
201      STX     2 EOFS   SAVE END OF O/P STRING PT
202      MDX     OUT      BR TO EXIT FR THIS PHASE
203 *
204 *           FORMAT STATEMENT
205 CBEL  S       FORMC   TEST FOR FORMAT STMT
206      BSC     L DECEM, - BRANCH IF FOUND
207 *
208 *           MOVE NON FORMAT STATEMENT
209 NOVEM STX     1 VENT   SAVE I/P STRING POINTER
210      LD      1 0      LD ID WD FR STRING
211      SRA     2        RIGHT JUSTIFY STMT NORM
212      AND     IDNRM    MASK STMT NORM
213      STO     VENT1 1   SAVE AS WORD COUNT
214 VENT1 LDX     L3 0     GET WD CNT THIS STMT
215 *           *MODIFIABLE
216 LUP   LD      1 0      LD WD I/P STRING
217      STO     2 0      SAVE IN O/P STRING
218      MDX     1 1      INCR, I/P STRING PT
219      MDX     2 1      INCR O/P STRING PT
220      MDX     3 -1     DECR WD COUNT
221      MDX     LUP      LOOP UNLESS FINISHED
222      STX     2 NEOFS   SAVE END OF O/P STRING
223      MDX     ABEL     BR TO PROCESS NXT STMT
224 *
225 *           STATEMENT NUMBER
226 DECEM STX     1 IDSAV  SAVE LOC OF I/P ID WD
227      LD      1 0      GET ID WORD
228      STO     IDSTO    SAVE IN TEMP STORAGE
229      BSC     L INDIC,E BR IF STMT NUMBERED
230 *
231 *           SET UP ERROR NO 28
232      LD      ERRA    LOAD ERROR NO, CON 28
233      STO     ERRNO   SAVE FOR OUTPUT PURPOSE
234 *
235 *           REPLACE STATEMENT WITH EPROR
236 BAKER LD      IDSTO   LD ID WORD
237      SRA     2        RIGHT JUSTIFY NORM
238      AND     IDNRM    MASK NORM
239      STO     NRMSV   SAVE NORM

```

240	A		IDSAV	INCR BY LOC OF ID
241	STO		GET1	SAVE AS START OF NXT STMT
242	LDX	I2	NEOFS	PUT END O/P STRING PT XR2
243	LD		IDSTO	LD ID WD
244	BSC	L	JAM,E	BR IF STMT NUMBERED
245	MDX		JAM 4	BR TO CONTINUE
246	JAM	LD	ERRID	LD ERR ID WD STMT NO,FLA
247	STO	2	0	SAVE IN O/P STRING
248	MDX	2	1	INCR O/P STRING PT
249	MDX		JAMB	SKIP NEXT 2 INSTRUCTIONS
250	LD		ERRID	LD ERR ID WD W/O STMT NO,
251	STO	2	0	SAVE IN O/P STRING
252	JAMB	LD	ERRNO	GET ERROR NUMBER
253	STO	2	1	SAVE IN O/P STRING
254	MDX	2	2	INCR O/P STRING PT
255	STX	2	NEOFS	SAVE POINTER
256	LDX	I1	GET1	GET LOC OF NXT STMT
257	MDX		ABEL	BR TO PROCESS NXT STMT
258	VENT	DC	0	LOC I/P STRING ID WD
259	ERRID	DC	/A00D	ERR ID W/ STMT NO, FLAG
260	BSS	E	0	
261	SUM	DC	0	SUM TABLE
262	SW1	DC	0	BEGINNING OF SPEC SW
263	SW2	DC	0	I TYPE SPEC SW
264	SW3	DC	0	2ND NO, OF SPEC SW
265	SW4	DC	0	LEFT/RIGHT CHAR SW FOR GET
266	SW5	DC	0	END OF STMT SW FOR GET
267	SW6	DC	0	COMMA ALLOWABLE SW
268	SW7	DC	0	LEFT/RIGHT SW FOR H SPEC
269	SW8	DC	0	COMMA MANDATORY SW
270	SW9	DC	0	REDO SW
271	IDTPE	DC	/7C00	MASK FOR STRING ID WORD
272	ENDC	DC	/0800	END STMT CONSTANT
273	FORMC	DC	/3000-/0800	FORMAT TYPE CONSTANT
274	ERRA	DC	28	CONSTANT ERR 28
275	IDSTO	DC	0	ID WD STORAGE
276	NEOFS	DC	0	END OF OUTPUT STRING
277	ERRNO	DC	0	ERROR NO, FOR O/P
278	IDSAV	DC	0	LOC OF ID WORD
279	IDNRM	DC	/01FF	MASK TO GET NORM
280	NRMSV	DC	0	NORM CURRENT STMT
281	ERRID	DC	/A008	ERROR ID NO STMT NO,
282	GET1	DC	0	LOC, OF NXT STMT IN STRIN
283	MASK	DC	/07FF	STMT NO, MASK
284	MASK1	DC	/0040	FORMAT FLAG FOR SYM TBL
285	MASKK	DC	/00FF	RIGHT CHAR MASK,
286	SAVE	DC	0	SYM TBL POINTER
287	THREE	DC	/FFFD	MINUS THREE CONSTANT
288	MASK2	DC	/00F0	NUMERIC EBC CONSTANT MASK
289	TEN	DC	10	CONSTANT TEN
290	N156	DC	156	MAX RECORD LENGTH *V1M3
291	ERRB	DC	27	ERR 27 CONSTANT
292	MONE	DC	/FFFF	MINUS ONE CONSTANT
293	REDO	DC	0	REDO INDICATOR
294	NORM	DC	0	NORM O/P STMT=2
295	NCNT	DC	0	NORM O/P STMT=2
296	TWO	DC	2	CONSTANT
297	NINE	DC	9	CONSTANT
298	SW15	DC	0	DIGIT ENCOUNTERED SW
299	*			

```

300 *          INDICATE FORMAT STATEMENT
301 *          IN SYMBOL TABLE ID WORD
302 *
303 INDIC LD   1 1      MOVE SYM TBL POINTER FR I/
304      STO  2 1      *STRING TO O/P STRING
305      AND      MASK  MASK OUT HIGH ORDER BITS
306      M       THREE  SYM TBL PT TO STMT NO,
307      SLT     16     *BY *3
308      S       THREE  SUBTRACT *3
309      A       L SOFST ADD TO START OF SYM TBL
310      STO     SAVE   SAVE SYM TBL ADDR STMT NO
311      LDX    I3 SAVE  PUT SYM TBL ADDR XR3
312      LD     3 0     LD SYM TBL ID WD
313      OR     MASK1   FLAG BIT 9 OF SYM TBL ID W
314      STO    3 0     *TO INDICATE FMT STMT
315 *
316 *          INITIALIZE TO SCAN BODY OF
317 *          STATEMENT
318      LD     1 0     MOVE STRING ID WD FR I/P
319      STO    2 0     *TO O/P STRING
320      STX   L2 VIGG  SAVE LOC O/P STRING ID WD
321      SRA     2     RIGHT JUSTIFY NORM
322      AND    IDNRM  MASK TO GET NORM BITS ONLY
323      STO    NRMSV  SAVE NORM
324      S      TWO   DECR BY 2 WDS ID STMT NO
325      STO    NCNT  SAVE WD CNT
326      STO    NORM  SAVE WD CNT
327      LD     NONE  LD *1
328      STO    L LOOP  INITIALIZE NO, REPEATS,
329      MDX    1 2     MOVE PTR PAST ID STMT N
330      LDX    3 10   SET UP TO CLEAR SUM AND SW
331      SLA    16    CLEAR ACC
332      STO    L TABC  CLEAR WD CNT O/P STRING
333      MDX    2     INCR O/P STRING PT
334      STO    L3 SUM+1  ZERO SUM SWITCH AREA,
335      MDX    3 -1   DECR TABLE COUNT
336      MDX    *-4   LOOP IF NOT DONE
337      STO    L SW11  ZERO W OR QUOTE SW
338      STO    SW12  ZERO QUOTE TYPE SPEC SW
339      STO    REDO  ZERO REDO SW
340      STO    L REP  ZERO REPEAT SW
341      STO    L GRP  ZERO GROUP REPEAT SW
342      STO    SW15  ZERO DIGIT ENCOUNTERED SW
343 *
344 *          GET CHAR FROM FORMAT STRIN
345      BSI   L GET    BR 10 SUBR TO GET CHAR
346 *
347 *          X LEFT PARENTHESIS
348      S     PARNL  TEST FOR LEFT PARENTHESIS,
349      BSC   L J,Z  BR IF NOT FOUND ERR 27
350 *
351 *          GET X
352      DOG   BSI   L GET  GET NXT CHAR FR FMT STMT
353      STO   X     SAVE CHAR
354 *
355 *          IS X NUMERIC
356      S     MASK2  SUBTRACT NUMERIC EBC CON F
357      BSC   L TST, Z  BR IF NOT NUMERIC
358      S     NINE   TEST IF NO, GT 9
359      BSC   L TST,Z-  BR IF YES

```

```

360 *
361 *          SUM SUM+10 X
362          LD      SJM      GET PREV SUM
363          M      TEN      MPY BY 10
364          SLT    16      SHIF ANS TO ACC
365          S      MASK2    REMOVE BITS 0-3 FR PRODUCT
366          A      X        ADD CHAR
367          STO    SUM      SAVE
368 *
369 *          TAG SW15      DIGIT ENCOUNTERED SW
370 *          SET SW NON ZERO
371          STX    0 SW15
372 *          SUM GREATER THAN 156          *V1M3
373          S      N156      SUBTRACT CON 156          *V1M3
374          BSC   L  ERR29,-Z BR IF SUM GT 156      ERR 29
375 *
376 *          NORMALIZE SW16
377          SLA    16      CLEAR INCOMPLETE SPEC
378          STO    SW16    *SWITCH
379          MDX    DOG     BR TO GET ANOTHER CHAR
380 *
381 *          SET UP ERROR 27
382          J      LD      ERRB      LD ERR CON 27
383          STO    ERRNO      SAVE IN ERR NO, BFR
384          BSC   L  BAKER      BR TO ERROR Q/P SUBR
385 *
386 *          SET UP ERROR 30
387          ERR30 LD      ER30      LD ERR CON 30
388          STO    ERRNO      SAVE IN ERR BFR
389          BSC   L  BAKER      BR TO ERROR Q/P SUBR
390 *
391 *          CONSTANTS
392          PARNL DC      /004D
393          X      DC      0          CHAR FR FMT STRING
394          FEF    DC      /00C6      F TEST CONSTANT
395          EE     DC      /00C5-/00C6 E TEST CONSTANT
396          EYE   DC      /00C9-/00C5 E TEST CONSTANT
397          ACH   DC      /00C8-/00C9 H TEST CONSTANT
398          XEX   DC      /00E7-/00C8 X TEST CONSTANT
399          AA    DC      /00C1-/00E7 A TEST CONSTANT
400          LP    DC      /004D          TEST CONSTANT
401          DIV   DC      /0061      / TEST CONSTANT
402          RPET DC      /005D-/0061 TEST CONSTANT
403          GREP DC      0          GROUP REPEAT SW
404          TENC DC      /B000      REDU COUNT CON
405          ER30 DC      30        ERROR 30 CONSTANT
406          SW12 DC      0          QUOTE TYPE SPEC SW
407          SW16 DC      0          INCOMPLETE SPEC SW
408 *
409 *          TEST SW1
410          TST   LD      L  SW1      TEST FOR BEGINNING OF SPEC
411          BSC   L  TST1,Z BR IF NOT BEGINNING
412 *
413 *          X F
414          LD    X      LD STRING CHAR
415          S     FEF     TEST FOR F
416 *
417 *          CONSTANTS
418          BSC   L  B, - BR IF CHAR F
419 *

```

```

420 *           X E
421 S           EE      TEST FOR E
422 BSC L C, -      BR IF CHAR E
423 *
424 *           X I
425 S           EYE     TEST FOR I
426 BSC L D, -      BR IF CHAR I
427 *
428 *           X H
429 S           ACH     TEST FOR H
430 BSC L E, -      BR IF CHAR H
431 *
432 *           X X
433 S           XEX     TEST FOR X
434 BSC L ZX, -     BR IF CHAR X
435 *
436 *           X A
437 S           AA      TEST FOR A
438 BSC L I, -      BR IF CHAR A
439 *
440 *           X LEFT PARENTHESIS
441 LD X          LOAD STRING CHAR
442 S LP         TEST FOR
443 BSC L GRP, -   BR IF CHAR
444 *
445 *           X DIVIDE
446 ECHO LD X     LOAD STRING CHAR
447 S DIV       TEST FOR /
448 BSC L RP, -  BR IF CHAR /
449 *
450 *           X RIGHT PARENTHESIS
451 S RPET      TEST FOR
452 BSC L NAN, - BR IF CHAR
453 *
454 *           X T
455 S TCON1     TEST FOR T
456 BSC L INK,Z BR IF CHAR NOT T
457 *
458 *           INDICATE T TYPE
459 LD L TTYPE  LD T INDICATOR CON
460 STO L COMP  SAVE IN O/P INDR AREA
461 BSI L SJBR  BR TO TEST SW15 AND SUM
462 BSC L POP   BR TO PROCESS T FORMAT
463 *
464 *           QUOTES
465 INK LD X     TEST CHAR FOR QUOTE
466 S L QJOTC   *MARK
467 BSC L J,Z   BR TO O/P ERR 27 IF NOT
468 *
469 *           TAG SW12
470 MDX L SW12,1 TAG QUOTE TYPE SPEC SW
471 *
472 *           WW 0
473 SLA 16      CLEAR ACC
474 STO L WW    CLEAR TOTAL FIELD WIDTH
475 BSC L 0     BR TO PROCESS QUOTE
476 *
477 *           RIGHT PARENTHESSES FOUND
478 *           SW15 AND SW16 NORMAL
479 MAN LD L SW15 TEST IF DIGIT ENCOUNTERED

```



```

480          OR          SW16          *AND INCOMPLETE SPEC SW OF
481          BSC L      J,Z          BR IF EITHER SW ON ERR 27
482 *
483 *
484          LD          GREP 0
485          BSC L      FOX,Z          TEST GROUP REPEAT SW
486 *
487 *
488          LD L      REDO          PUT REDO
489          OR          TENC          GET REDO INDICATOR
490          BSI L      PUT          ADD COUNT MASK
491 *
492 *
493          MDX L      SW9,1        PUT REDO IN O/P STRING
494 *
495 *
496          BSI L      GET          SET SW9 TO TRANSFER
497          MDX          J          INDICATES OK TO PUT REDO
498 *
499 *
500 PERD DC          /004B          GET X
501 EITC DC          /8000          BR TO GET NXT CHAR
502 ONE DC          1          BR IF RETURN FR GET=ERR 27
503 *
504 *
505 *
506 *
507 *
508 FOX LD          GREP 1
509 S          ONE          LD GROUP REPEAT SW
510 BSC L      XENO, -          TEST FOR GROUP REPEAT 1
511 *
512 *
513          LD          EITC          BR IF
514          OR          GREP          PUT GROUP REPEAT
515          BSI L      PUT          LD GROUP REPEAT CON
516          LD          LOOP          ADD TO GROUP REPEAT COUNT
517          A          ONE          PUT ON O/P STRING
518          BSI L      PUT          GET -NO, WDS TO REPEAT
519 *
520 *
521 XENO SLA          16          ADD 1
522 STO          GREP          PUT ON O/P STRING
523 *
524 *
525          BSI L      GET          GREP 0
526          STO L      X          RESET GROUP REPEAT CON
527 *
528 *
529          MDX L      SW8,1        *TO 0
530          BSC L      QJAD          GET X
531 *
532 *
533 TST1 LD L      SW3          TAG SW8
534          BSC L      M, -          SET COMMA MANDATORY SW
535 *
536 *
537          LD L      SUM 0          BR TO PROCESS NEW SPEC TYP
538          BSC L      J, -          TEST SW3
539 *

```

```

540 *                WW SUM
541 *                STO   WW          SAVE TOTAL FIELD WIDTH
542 *
543 *                SUM 0
544 *                BSI   L  NLIZE    BR 10 CLEAR SUM AND SW15
545 *
546 *                TEST SW2
547 *                LD    L  SW2      TEST FOR I TYPE SPEC
548 *                BSC   L  K, -    BR IF NOT I TYPE SPEC
549 *
550 *                WW LESS THAN 127
551 *                LD    WW          TEST TOTAL FIELD WIDTH
552 *                S      N127      *VS 127
553 *                BSC   L  ERR30,-Z BR IF GT 127
554 *
555 *                X PERIOD
556 *                LD    L  WALT     GET RIGHT CHAR FR TEMP STO
557 *                S      PERD      TEST FOR PERIOD
558 *                BSC   L  J,Z     BR IF NOT PERIOD  EPR 27
559 *
560 *                TAG SW16
561 *                MDX   L  SW16,1   SET INCOMPLETE SPEC SW
562 *
563 *                NORMALIZE SW2
564 *                SLA   L  16       CLEAR ACC
565 *                STO   L  SW2     SET SW 10 IND FOR I TYPE
566 *
567 *                NORMALIZE SW3
568 *                STO   L  SW3     SET SW TO IND NO 2ND NO,
569 *                BSC   L  DOG     BR 10 PROCESS NXT CHAR
570 *
571 *                CONSTANTS
572 *                LOOP  DC          0      =NO, WDS BACK TO GROUP REP
573 *                WW    DC          0      TOTAL FIELD WIDTH
574 *                DD    DC          0      DECIMAL WIDTH
575 *                N127 DC          127     MAX TOTAL FIELD WIDTH
576 *                N7    DC          /7000  SLASH FLAG
577 *                N31   DC          31     MAX DECML WIDTH
578 *                REP   DC          0      REPEAT COUNT
579 *                BOX   DC          0      SPECIAL TYPE CODE STORAGE
580 *                OPA   DC          /A035  SPECIAL TYPE CODE CHAR  A
581 *                OPB   DC          /A014  SPECIAL TYPE CODE CHAR  B
582 *                OPS   DC          /A040  SPECIAL TYPE CODE CHAR  S
583 *                OPT   DC          /A005  SPECIAL TYPE CODE CHAR  T
584 *                OPD   DC          /A016  SPECIAL TYPE CODE CHAR  D
585 *                OPL   DC          /A025  SPECIAL TYPE CODE CHAR  L
586 *                OPR   DC          /A015  SPECIAL TYPE CODE CHAR  R
587 *                VIS1  DC          0      SIMNT NORM-2
588 *                ACON  DC          /00C1  A TEST CONSTANT
589 *                BCON  DC          /00C2-/00C1 B TEST CONSTANT
590 *                SCON  DC          /00E2-/00C2 S TEST CONSTANT
591 *                TCON  DC          /00E3-/00E2 T TEST CONSTANT
592 *                DCON  DC          /00C4-/00E3 D TEST CONSTANT
593 *                LCON  DC          /00D3-/00C4 L TEST CONSTANT
594 *                RCON  DC          /00D9-/00D3 R TEST CONSTANT
595 *                RPARN DC          /005D   TEST CONSTANT
596 *                TCON1 DC          /00E3-/005D T TEST CONSTANT
597 *                TTYPE DC          /6000   T INDICATOR CONSTANT
598 *
599 *                SAVE POINTER AND SAVE SW4

```

```

600 * X LEFT PARENTHESIS
601 GRP STX 1 BK 1 SAVE I/P STRING PT
602 LDD L SW4 GET CONTENTS SW4 AND SW5
603 STD L SW45H SAVE IN TEMPORARY STORAGE
604 LD L NCNT GET STMNT NORM=2
605 STO V1S1 SAVE
606 *
607 * GET X
608 BSI L GET GET STRING CHAR
609 MDX BK BR IO PROCESS
610 *
611 * X A
612 S ACON TEST CHAR FOR A
613 BSC L BOY, - BR IF A
614 *
615 * X B
616 S BCON TEST CHAR FOR B
617 BSC L BOY1, - BR IF B
618 *
619 * X S
620 S SCON TEST CHAR FOR S
621 BSC L BOY2, - BR IF S
622 *
623 * X T
624 S TCON TEST CHAR FOR T
625 BSC L BOY3, - BR IF T
626 *
627 * X D
628 S DCON TEST CHAR FOR D
629 BSC L BOY4, - BR IF D
630 *
631 * X L
632 S LCON TEST CHAR FOR L
633 BSC L BOY5, - BR IF L
634 *
635 * X R
636 S RCON TEST CHAR FOR R
637 BSC L BOY6, - BR IF R
638 *
639 * BACK UP
640 BK LDX L1 *** RESET I/P STRING POINTER
641 * *MODIFIABLE
642 LDD L SW45H RELOAD SW4 AND SW5
643 STD L SW4 RESTORE TO PREV STATE
644 LD V1S1 RESET STMNT NORM=2
645 STO L NCNT SAVE
646 MDX GRP1 BR IO PROCESS PREV CHAR
647 *
648 * INDICATE A SPECIAL TYPE
649 BOY LD OPA CHAR A, LD SPECIAL TYPE CD
650 MDX BOY3 1 BR IO STORE
651 *
652 * INDICATE B SPECIAL TYPE
653 BOY1 LD OPB CHAR B, LD SPECIAL TYPE CD
654 MDX BOY6 1 BR IO STORE
655 *
656 * INDICATE S SPECIAL TYPE
657 BOY2 LD OPS CHAR S, LD SPECIAL TYPE CD
658 MDX BOY6 1 BR IO STORE
659 *

```

```

660 *          INDICATE T SPECIAL TYPE
661 BOY3 LD     OPT   CHAR T, LD SPECIAL TYPE CD
662 MDX     ROY6 1   BR IO STORE
663 *
664 *          INDICATED D SPECIAL TYPE
665 BOY4 LD     OPD   CHAR D, LD SPECIAL TYPE CD
666 MDX     BOY6 1   BR IO STORE
667 *
668 *          INDICATE L SPECIAL TYPE
669 BOY5 LD     OPL   CHAR L, LD SPECIAL TYPE CD
670 MDX     BOY6 1   BR IO STORE
671 *
672 *          INDICATE R SPECIAL TYPE
673 BOY6 LD     OPR   CHAR R, LD SPECIAL TYPE CD
674 STO     BOX     SAVE SPECIAL TYPE CODE
675 *
676 *          GET X
677 BTEST BSI L   GET   BR IO GET ANOTHER CHAR
678 *
679 *          RIGHT PARENTHESIS
680 S       RPARN  TEST CHAR FOR
681 BSC L   BK,Z   BR IF NOT
682 *
683 *          PUT SPECIFICATION
684 LD     BOX     GET SPECIAL TYPE CODE
685 BSI L   PUT    PUT ON O/P STRING
686 *
687 *          SUM 0
688 BSI L   SUBR   BR IO SEE IF DIGIT FOUND
689 LD L   SUM    GET FIELD WIDTH
690 BSC L   BT11A, - BR IF FIELD WIDTH 0
691 *
692 *          PUT FIELD REPEAT
693 LD     N9     GET FIELD REPEAT FLAG
694 OR L   SUM    ADD FIELD WIDTH
695 BSI L   PUT    PUT ON O/P STRING
696 *
697 *          CALL NLIZE
698 BT11A BSI L   NLIZE  CLEAR SW15 AND SUM
699 *          GET X
700 BT11 BSC L   LIMA   BR IO PROCESS NXT CHAR
701 *
702 *          GREP 0
703 GRP1 LD L   GREP   LD GROUP REPEAT CNT
704 BSC L   J,Z     BR IO ERR IF 0, ERR 27
705 *
706 *          INITIALIZE RE DO
707 STO L   REDO    SAVE IN REDO COUNT
708 *
709 *          SUM 0
710 BSI L   SUBR   BR IO TEST SW15 AND SUM
711 *
712 *          GREP SUM OR 1 IF SUM 0
713 LD L   SUM    LD SUM
714 BSC L   * 2,Z  BR IF NOT 0
715 LD L   ONE    SET GROUP REPEAT CONSTANT
716 STO L   GREP  * IO 1
717 *
718 *          CALL NLIZE
719 BSI L   NLIZE  CLEAR SW15 AND SUM

```

```

720 *
721 *          LOOP *1
722          LD   L   NONE   GET *1 CONSTANT
723          STO  L   LOOP   SAVE
724          BSC  L   DOG    BR 10 PROCESS NXT CHAR
725 *
726 *          REP 0
727 RP        LD   L   REP   TEST REPEAT FLAG
728          BSC  L   J,Z    BR 10 ERR IF NE 0  ERR 27
729 *
730 *          PUT SLASH
731          LD   L   N7     GET SLASH INDICATOR
732          BSI  L   PUT    PUT ON O/P STRING
733          BSC  L   LIMA   BR 10 GET ANOTHER CHAR
734 *
735 *          TEST INDICATORS AND GET NEXT
736 *          CHAR AND FIELD WIDTH
737 *
738 *          REP 0
739 F        LD   L   REP   GET REPEAT FLAG
740          BSC  L   J,Z    BR 10 ERR IF NE 0  ERR 27
741 *
742 *          SUM 0
743          LD   L   SUM   TEST SUM FIELD WIDTH
744          BSC  L   J,-   BR 10 ERR IF 0  ERR 27
745 *
746 *          WW SUM
747          LD   L   SUM
748          STO  L   WW    SAVE SUM IN TOT FLD WIDTH
749 *
750 *          CALL NLIZE
751          BSI  L   NLIZE  CLEAR SUM AND SW15
752 *
753 *          TEST SW6
754          LD   L   SW6   TEST FOR COMMA ALLOWABLE
755          BSC  L   0,Z   BR IF NOT ALLOWABLE
756 *
757 *          TAG SW8
758          MDX  L   SW8,1 SET COMMA NOT MANDATORY SW
759 *
760 *          GET X
761          BSI  L   GET   GET NEXT CHAR
762          STO  L   X    SAVE
763          BSC  L   K    BR 10 BUILD SPECIFICATIONS
764 *
765 *          TEST SW15
766 M        LD   L   SW15  LD DIGIT ENCOUNTERED SW
767          BSC  L   J,-   BR IF NO DIGIT FOUND
768 *
769 *          DD LESS THAN 32
770          LD   L   SUM   GET FIELD WIDTH
771          S    L   N31   SUBTRACT CONSTANT 31
772          BSC  L   ERR30,Z- BR IF WIDTH GT 31
773 *
774 *          DD SUM
775          LD   L   SUM   GET FIELD WITH
776          STO  L   DD    SAVE AS DECML WIDTH
777 *
778 *          WW G.T. DD
779          S    L   WW    COMPARE WITH TOT FLD WIDTH

```

```

780          BSC  L  ERR30,-Z  BR ERR IF TOT LT DECML
781  *
782  *          CALL NLIZE
783          BSI  L  NLIZE      BR IO CLEAR SUM AND SW15
784          BSC  L  K          BR IO BUILD SPECIFICATIONS
785  *
786  *          TAG SW6
787  E      MDX  L  SW6,1      SET COMMA NOT ALLOWABLE SW
788          MDX  F          BR IO TEST IF GET NXT CHAR
789  *
790  *          CONSTANTS
791  BLANK DC      /0040      EBC BLANK
792  Y      DC      0          LEFT CHAR TO PACK O/P AREA
793  COMMA DC      /006B      EBC COMMA
794  N5     DC      /5000      HOLLERITH MASK
795  N9     DC      /9000      FIELD REPEAT FLAG
796  QUOTC DC      /007D      QUOTE MASK
797  CT     DC      0          NO, CHARS HOLL FMT
798  JAMEY DC      0          O/P STRING WD CNT
799  ER29  DC      29         ERROR 29 CONSTANT
800  C146  DC      146        MAX HOLL CHAR CNT
801  SW11  DC      0          H OR QUOTE TYPE SPEC SW
802          BSS  E  0
803  SW45H DC      ***        TEMP STU SW4
804          DC      ***        TEMP STU SW5
805  NCNTH DC      ***        TEMP STU NORM COUNT
806  *
807  *          NORMALIZE SW6  COMMA ALLOWABLE
808  0      SLA  16          CLEAR ACC
809          STO  L  SW6      SET COMMA ALLOWABLE FLAG
810  *
811  *          SET UP COUNT
812          SLA  16
813          STO  L  CT          CLEAR CNT OF NO, HOLL CHAR
814  *
815  *          PUT H SPEC,
816          STX  2  JAMEY    SAVE O/P STRING POINTER
817          LD   N5          LD HOLL SPEC MASK
818          OR   L  WW          ADD TO TOTAL FIELD WIDTH
819          BSI  L  PUT          PUT SPEC ON O/P STRING
820  *
821  *          TAG SW11
822          MDX  L  SW11,1    SET H OR QUOTE SPEC FLAG
823  *
824  *          GET X
825  KILO  BSI  L  GET          GET NEXT CHAR
826          STO  L  X          SAVE
827  *
828  *          TEST SW12=QUOTE TYPE SPEC
829          LD   L  SW12      LD SWITCH
830          BSC  L  LOB, -    BR IF NOT QUOTE TYPE SPEC
831  *
832  *          X QUOTE
833          LD   L  X          LD STRING CHAR
834          S    QUOTC        TEST FOR QUOTE CONSTANT
835          BSC  L  LOB3, -   BR IF QUOTE FOUND
836  *
837  *          CHAR NOT QUOTE, PROCESS AS
838  *          HOLLERITH CHAR
839  *          NCNT  0

```

```

840          LD   L   NCNT          TEST NORM COUNT
841          BSC  L   ERR29, - BR IF COUNT  0
842          *
843          LOB2 LD   L   TWO          GET A CONSTANT  2
844          *
845          STO  L   WW          WW 2          SET TOTAL FIELD WIDTH  2
846          *
847          *
848          MDX  L   CT,1          CT CT 1          INCR HOLL CHAR COUNT
849          MDX  L   LOR          BR IO O/P CHAR
850          *
851          *
852          *
853          *
854          *
855          LOB3 LDD  L   SW4          SAVE STATUS OF GET ROUTINE
856          STD  L   SW45H          GET SW4 AND SW5
857          LD   L   NCNT          SAVE IN TEMP STO
858          STO  L   NCNT          GET NORM COUNT
859          STX  1  RXR1H 1          SAVE IN TEMP STO
860          *
861          *
862          BSI  L   GET          GET X
863          *
864          *
865          S     QUOTE          GET ANOTHER CHAR
866          BSC  L   LOB2, -      TEST CHAR FOR QUOTE
867          *
868          *
869          LDD  L   SW45H          RESTORE STATUS OF GET ROUTINE
870          STD  L   SW4          LOAD SW4 AND SW5 FR TEMP
871          LD   L   NCNT          RESTORE SW4 AND SW5
872          STO  L   NCNT          LD TEMP NORM COUNT
873          RXR1H LDX  L1  ***          SAVE
874          *
875          *
876          *
877          LOB1 STX  2  JAMES 1          RESTORE I/P STRING POINTER
878          LDX  12  JAMEY          *MODIFIABLE
879          LD   2  0          PUT OUT SPECIAL QUOTE
880          OR   CT          SAVE O/P STRING POINTER
881          STO  2  0          LD START OF QUOTE AREA PT
882          JAMES LDX  L2  0          LD FIRST WD OF QUOTE AREA
883          *
884          *
885          *
886          LD   CT          ADD NO CHARS
887          BSC  L   NZER,Z          MAY SKIP
888          MDX  2  -1          DECK REDO
889          MDX  L   TABC,-1          MAY SKIP
890          NOP          INCR LOOP
891          MDX  L   REDO,-1          MAY SKIP
892          NOP          MAY SKIP
893          MDX  L   LOOP,1          INCR LOOP
894          NOP          MAY SKIP
895          NZER EQU  *
896          S     C146          SUBTRACT MAX SIZE LITERAL
897          BSC  L   JILL, Z          BR IF LITERAL LT 146
898          *
899          *
          SET UP ERROR 29

```

```

900 ERR29 LD      ER29      LD ERR 29 CON
901      STO L  ERRNO      SAVE IN ERR NO,
902      BSC L  BAKER      BR 10 OUTPUT ERR
903 *
904 *                NORMALIZE SW12
905 JILL  SLA      16      CLEAR ACC
906      STO L  SW12      SET NON-QUOTE TYPE SPEC,
907      MDX      HILO 3    BR 10 CONT WITH STRING
908 *
909 *                OUTPUT HULLERITH CHAPACTERS
910 *                PACKED 2 TO A WORD
911 *                TEST SW7
912 LOB   LD L  SW7      LD LEFT/RIGHT SW
913      BSC L  GAM, -    BR IF LEFT
914 *
915 *                NORMALIZE SW7
916      SLA      16      CLEAR ACC
917      STO L  SW7      SET SW TO LEFT FOR NXT CHA
918 *
919 *                PUT Y X
920      LD      Y      LD LEFT HOLL CHAR TO BE O/
921      SLA      8      SHIFT TO LEFT HALF OF WD
922      OR L  X      ADD PRESENT CHAR RIGHT HAL
923      BSI L  PUT      BR 10 PUT PACKED WD IN O/P
924 *
925 *                WW WW=1
926 HILO  MDX L  WW,-1    DECK TOTAL FIELD WIDTH
927 *
928 *                WW 0
929      MDX      KILO      BR 10 GET NXT CHAR IF ANY
930 *
931 *                TEST SW7
932      LD L  SW7      TEST FOR LEFT CHAR TO O/P
933      BSC L  LIMA, -   BR IF LEFT CHAR TO O/P
934 *
935 *                PUT Y BLANK
936      LD      Y      LD LEFT CHAR
937      SLA      8      SHIFT TO LEFT HALF OF WD
938      OR      BLANK    ADD EBC BLANK IN RIGHT HAL
939      BSI L  PUT      PUT WD ON O/P STRING
940 *
941 *                NORMALIZE SW7
942      SLA      16      CLEAR ACC TO SET LEFT HALF
943      STO L  SW7      *O/P SW FOR NEXT HOLL O/P
944 *
945 *                TAG SW8
946 LIMA  MDX L  SW8,1    SET COMMA MANDATORY SW
947 *
948 *                NORMALIZE SW11
949      SLA      16      CLEAR ACC TO SET NOT H OR
950      STO      SW11     *QUOTE SPEC SW
951 *
952 *                GET X
953      BSI L  GET      GET NXT CHAR
954      STO L  X      SAVE
955      MDX      QUAD     BR 10 PROCESS NEW SPEC
956 *
957 *                Y X
958 GAM  LD L  X      GET STRING CHAR FOR LEFT
959      STO L  Y      *HAND O/P OF PACKED WD

```



```

960 *
961 *          TAG SW7
962 MDX L SW7,1 SET SW FOR RIGHT CHAR NXT
963 MDX HILO CONTINUE PROCESSING STRING
964 *
965 *          TAG SW2
966 OLD MDX L SW2,1 SET I TYPE SPEC SW
967 *
968 *          TAG SW3
969 POP MDX L SW3,1 SET 2ND NO. OF SPEC SW
970 *
971 *          REP SUM
972 LD L SUM LD FIELD WIDTH
973 STO L REP SAVE IN REPEAT
974 *
975 *          CALL NLIZE
976 BSI L NLIZE CLEAR SUM, SW15
977 *
978 *          TAG SW1
979 MDX L SW1,1 SET END OF SPEC SW
980 BSC L DOG BR FOR NEXT SPEC TYPE
981 *
982 *          BUILD SPECIFICATIONS
983 K LD L DD LD DECML FIELD WIDTH
984 SLA 7 LEFT JUSTIFY
985 OR L WW ADD TOT FIELD WIDTH=RIGHT
986 OR COMPT ADD O/P TYPE INDICATOR
987 *
988 *          PUT SPECIFICATION
989 HAN1 BSI L PUT PUT SPEC ON O/P STRING
990 *
991 *          DD 0
992 SLA 16 CLEAR ACC
993 STO L DD CLEAR DECML FIELD WIDTH
994 *
995 *          WW 0
996 STO L WW CLEAR TOTAL FIELD WIDTH
997 *
998 *          REP 0
999 LD L REP TEST REPEAT COUNT
1000 BSC L QUAD, - BR IF 0 FOR NXT SPEC
1001 *
1002 *          REP 1
1003 S L ONE SUBTRACT 1
1004 BSC L ZEBRA, - BR IF 1
1005 *
1006 *          PUT REPEAT
1007 LD L N9 LD REPEAT INDICATOR
1008 OR L REP ADD REPEAT COUNT
1009 BSI L PUT PUT ON O/P STRING
1010 *
1011 *          REP 0
1012 ZEBRA SLA 16 CLEAR ACC
1013 STO L REP CLEAR REPEAT COUNT
1014 *
1015 *          NEW SPECIFICATION TYPE TO BE
1016 *          PROCESSED
1017 *          NORMALIZE SW1
1018 QUAD SLA 16 CLEAR ACC TO SET BEGINNING
1019 STO L SW1 *OF SPEC SW

```

```

1020 *
1021 *           X COMMA
1022         LD   L   X           LD STRING WD
1023         S    L   COMMA      SUBTRACT EBC COMMA
1024         BSC  L   CATT,Z     BR IF COMMA NOT FOUND
1025 *
1026 *           NORMALIZE SW15  DIGIT ENCOUNTERED SW
1027         STO  L   SW15       SET DIGIT NOT ENCOUNTERED
1028 *
1029 *           NORMALIZE SW8   MANDATORY COMMA
1030         STO  L   SW8        SET COMMA MANDATORY
1031         RSC  L   DOG        BR TO CALC FIELD WIDTH
1032 *
1033 *           TEST SW8
1034         CATT LD   L   SW8     LD COMMA MANDATORY SW
1035         BSC  L   ECHO, -     BR IF COMMA MANDATORY
1036 *
1037 *           NORMALIZE SW8
1038         SLA          16       CLEAR ACC
1039         STO  L   SW8        SET COMMA MANDATORY SW
1040         LD   L   X           LD STRING CHAR
1041         BSC  L   DOG 3      BR TO PROCESS
1042 *
1043 *           CONSTANTS
1044         XTYPE DC          /4000  X TYPE SPEC CONSTANT
1045         FTYPE DC          /1000  F TYPE SPEC CONSTANT
1046         ETYPE DC          0      E TYPE SPEC CONSTANT
1047         ITYPE DC          /2000  I TYPE SPEC CONSTANT
1048         ATYPE DC          /3000  A TYPE SPEC CONSTANT
1049         COMTP DC          0      CON INDICATING O/P TYPE
1050 *
1051 *           SET INDICATORS FOR VARIOUS TYPE
1052 *           SPECIFICATIONS
1053 *           INDICATE X TYPE
1054         ZX   LD   XTYPE      LD X TYPE CONSTANT
1055         STO  COMTP          SAVE IN CONSTANT TYPE
1056         BSC  L   F          BR TO CALC FIELD WIDTH
1057 *
1058 *           INDICATE F TYPE
1059         B    LD   FTYPE      LD F TYPE SPEC CONSTANT
1060         STO  COMTP          SAVE CONSTANT TYPE
1061         BB   BSI  L   SUBR    CLEAR SUM AND SW15
1062         MDX          OLD      BR TO CONTINUE
1063 *
1064 *           INDICATE E TYPE
1065         C    LD   ETYPE      LD E TYPE SPEC CONSTANT
1066         STO  COMTP          SAVE CONSTANT TYPE
1067         MDX          BB      BR TO CONTINUE
1068 *
1069 *           INDICATE I TYPE
1070         D    LD   ITYPE      LD I TYPE SPEC CONSTANT
1071         STO  COMTP          SAVE CONSTANT TYPE
1072         DD2  BSI  L   SUBR    CLEAR SW15 AND SUM
1073         MDX          POP      BR TO CONTINUE
1074 *
1075 *           INDICATE A TYPE
1076         I    LD   ATYPE      LD A TYPE SPEC CONSTANT
1077         STO  COMTP          SAVE
1078         MDX          DD2     BR TO CONTINUE
1079 *

```

```

1080 *          SUBROUTINE TO GET A WORD FROM A
1081 *          FORMAT STATEMENT FOR ANALYSIS
1082 *          TAKES PACKED EBC CHARS FROM INPUT
1083 *          STRING AND EMITS ONE CHAR AT A TIME
1084 *          RIGHT JUSTIFIED IN THE ACCUMULATOR.
1085 *
1086 GET      DC      0          BSI ENTRY POINT
1087 *
1088 *          TEST SW5 FOR END OF STMT
1089 GETY     LD      L  SW5      LD END OF STMT SW
1090         BSC     L  MAT, -    BR IF NOT END OF STMT
1091 *
1092 *          TEST SW9
1093         LD      L  SW9      LD REDO SW
1094         BSC     L  FAT, -    BR IF NOT OK TO O/P REDO
1095 *
1096 *          RE-ADJUST ID NORM
1097         STX     2  ZR 1      SAVE O/P STRING POINTER
1098         LDX     I2 VIGG      LD I/D LOC OF O/P STRING
1099         LD      TABC        LD NO. WDS O/P STRING
1100         A      L  TWO       ADD 2 FOR STMT NO. AND ID
1101         SLA     2          SHIFT TO NORM BITS 6-13
1102         STO     TABC        SAVE IN COUNT AREA
1103         LD      2  0        LD ID WD O/P STRING
1104         AND     MATB        MASK OUT PREV STMT NORM
1105         OR      TABC        MASK IN NEW NORM
1106         STO     2  0        SAVE NEW ID WD O/P STRING
1107 ZR      LDX     L2 0        LD ACTUAL O/P STRING PT
1108 *          *MODIFIABLE
1109         STX     L2 NEOFS     SAVE AS END OF STRING
1110         BSC     L  ABEL      BR TO PROCESS NXT STMT
1111 *
1112 *          PUT BLANK IN A REGISTER
1113 FAT      LD      L  BLANK    LD EBC BLANK
1114         BSC     I  GET       RETURN TO CALLING PROGRAM
1115 *
1116 *          TEST SW4
1117 MAT      LD      L  SW4      LD LEFT/RIGHT CHAR SW
1118         BSC     L  LUKE,Z    BR IF RIGHT CHAR
1119 *
1120 *          TAG SW4
1121         MDX     L  SW4,1     SET SW TO PROCESS RIGHT NX
1122 *
1123 *          PUT LEFT CHAR, IN A REGISTER
1124         LD      1  0        LD WD I/P STRING
1125         SRA     8          CHAR IN BITS 8-15
1126         AND     L  MASK      MASK OUT BIS 0-7
1127         STO     WALT        SAVE IN TEMP STORAGE.
1128 *
1129 *          TEST SW11
1130 ABIT     LD      L  SW11     LD QUOTE TYPE SPEC SW
1131         BSC     L  LARR, -    BR IF NOT QUOTE TYPE
1132         LD      WALT        LD O/P WD
1133         MDX     LARRA       BR TO EXIT
1134 *
1135 *          IS CHARACTER BLANK
1136 LARR     LD      WALT        LD CHAR FROM TEMP STD
1137         S      L  BLANK      TEST FOR EBC BLANK
1138         BSC     L  GETY, -    BR IF BLANK TO GET NXT CHA
1139         A      L  BLANK      ELSE, RESTORE CHAR

```

```

1140 LARRA BSC I GET RETURN TO CALLING PROGRAM
1141 *
1142 * NORMALIZE SW4
1143 LUKE SLA 16 SET LEFT/RIGHT CHAR SW FOR
1144 STO L SW4 *LEFT CHAR NEXT TIME
1145 *
1146 * PUT RIGHT CHAR IN A REGISTER
1147 LD 1 0 LD WD FR I/P STRING
1148 AND L MASKK MASK OUT BITS 0-7
1149 STO WALT SAVE IN TEMPORARY STORAGE
1150 *
1151 * MOVE POINTER 1
1152 MDX 1 1 INCR INPUT STRING POINTER
1153 *
1154 * NCNT NCNT=1
1155 MDX L NCNT,-1 DECR COUNT OF WDS IN STMT
1156 *
1157 * NCNT 0
1158 MDX ABIT BR IF COUNT NOT FINISHED
1159 *
1160 * TAG SW5
1161 MDX L SW5,1 SET END OF STMT SW
1162 MDX ABIT BR TO TEST CHAR
1163 *
1164 * CONSTANTS
1165 WALT DC 0 TEMP STU FOR CHAR OBTAINED
1166 VIGG DC 0 LOC OF STMT ID
1167 TABC DC 0 NO. WDS O/P STRING STMT
1168 MATB DC /F803 MASK TO CLEAR NORM ID WD
1169 VAB DC 0 TEMP STU I/P STRING PT
1170 VAB1 DC 0 TEMP STU O/P STRING PT
1171 *
1172 * SUBROUTINE TO PUT WORD ON OUTPUT
1173 * STRING, ACC CONTAINS WORD TO BE
1174 * OUTPUT ON ENTRY TO THIS SUBROUTINE,
1175 *
1176 PUT DC 0 SUBROUTINE ENTRY POINT
1177 *
1178 * PUT A REGISTER IN BUILD BUFFER
1179 STO 2 0 SAVE O/P WD IN STRING
1180 MDX L TABC,1 INCR O/P WD CNT
1181 *
1182 * MOVE POINTER
1183 MDX 2 1 INCR O/P STRING PT
1184 *
1185 * CHECK OVERLAP
1186 STX 1 VAB1 SAVE I/P STRING PT
1187 STX 2 VAB SAVE O/P STRING PT
1188 LD VAB1 COMPARE I/P POINTER
1189 S VAB * WITH OUTPUT POINTER
1190 BSC L VAB2,- BR IF I/P GE O/P
1191 *
1192 * SET JP OVERLAP ERROR
1193 ZAZ MDX L ERROR,1 SET SYSTEM OVERLAP ERR FLA
1194 BSC L OUT BR TO EXIT FR COMPILER
1195 *
1196 * REDO REDO 1
1197 VAB2 MDX L REDO,1 INCR REDO COUNT
1198 *
1199 * LOOP LOOP=1

```

```

1200 MDX L LOOP,-1 INCR LOOP COUNT
1201 BSC I PUT RETURN TO CALLING PROGRAM
1202 *
1203 * THIS SUBROUTINE NORMALIZES SUM AND SW1
1204 *
1205 NLIZE DC 0 SUBROUTINE ENTRY POINT
1206 *
1207 * SUM 0
1208 SLA 16 CLEAR ACC
1209 STO L SUM CLEAR FIELD WIDTH
1210 *
1211 * NORMALIZE SW15
1212 STO L SW15 CLEAR DIGIT ENCOUNTERED SW
1213 BSC I NLIZE RETURN TO CALLING PROG
1214 *
1215 * THIS ROUTINE TESTS SW15 AND SUM
1216 *
1217 SUBR DC 0 SUBROUTINE ENTRY POINT
1218 *
1219 * SW15 TAGED
1220 LD L SW15 TEST FOR DIGIT ENCOUNTERED
1221 BSC - SKIP IF YES
1222 MDX OUT2 RETURN TO CALLING PROG IF
1223 *
1224 * SUM 0
1225 LD L SUM LD FIELD WIDTH
1226 BSC L J, - BR TO ERR IF 0 ERR 27
1227 OUT2 BSC I SUBR RETURN TO CALLING PROG IF
1228 BSS OVERL-***320*3 PHASE=10 PATCH AREA
1229 END START

```