

```

1 * GENERAL AUTOMATION, INC. ALL RIGHTS RESERVED
2 *****
3 *
4 * PROGRAM NAME FPH-11
5 *
6 * MODEL NUMBER 8F011
7 *
8 * PURPOSE FORTRAN PHASE=11
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 ** SUBSCRIPT DECOMPO
22 *****
23 *
24 *
25 *FUNCTION/OPERATION-
26 * * BYPASSES ALL FORMAT, CONTINUE AND ERROR
27 * * STATEMENTS,
28 * * SCANS ALL OTHER STATEMENTS BUT OPERATES ONLY
29 * * ON THOSE STATEMENTS CONTAINING THE SPECIAL
30 * * LEFT PARENTHESIS OPERATOR INSERTED BY PHASE8
31 * * CHECKS SUBSCRIPTING INFORMATION FOR EACH
32 * * VARIABLE FOR VALIDITY,
33 * * CALCULATES THE SUBSCRIPT CONSTANT D4 AND,
34 * * DEPENDING ON THE DIMENSIONING LEVEL, THE
35 * * CONSTANTS D1, D2, AND D3,
36 * * INSERTS THE SUBSCRIPT CONSTANTS INTO THE
37 * * SUBSCRIPT EXPRESSION WITH THE SUBSCRIPT
38 * * INDICES,
39 * * CHANGES THE RIGHT AND LEFT PARENTHESIS
40 * * ENCLOSING THE SUBSCRIPT EXPRESSION TO
41 * * SPECIAL OPERATORS TO BE USED IN A LATER
42 * * PHASE,
43 *
44 *ENTRY POINTS-
45 * * START-PHASE 11 IS READ INTO CORE BY PHASE
46 * * 10 VIA A CALL TO ROLRX, EXECUTION IS
47 * * BEGUN AT LOCATION START,
48 *
49 *INPUT-
50 * * THE STATEMENT STRING
51 * * THE SYMBOL TABLE
52 * * FCOM
53 *
54 *OUTPUT-
55 * * THE MODIFIED STATEMENT STRING
56 * * THE SYMBOL TABLE
57 * * FCOM
58 *
59 *EXTERNAL REFERENCES-

```

```

60 * * SUBROUTINES-
61 *   ROLRX
62 * * OTHER FORTRAN PHASES
63 *   NONE
64 *
65 *EXITS
66 * * NORMAL-
67 *   EXITS VIA A CALL TO THE ROLRX ROUTINE TO
68 *   READ IN PHASE 12.
69 *
70 * * ERROR-
71 *   OVERLAP-PROCESSING IS HALTED, THE ERROR
72 *   WORD IS SET AND THE FOLLOWING
73 *   PHASE IS CALLED VIA ROLRX.
74 *   SYNTAX- ERRORS DETECTED BY THIS ARE NUMBERS
75 *   31,32,33,34 AND 35.
76 *
77 *TABLES/WORK AREAS-
78 * * THE STATEMENT STRING
79 * * THE SYMBOL TABLE
80 * * FCOM
81 *
82 *ATTRIBUTES- NONE,
83 *
84 *NOTES- THE SWITCHES USED IN PHASE 11 FOLLOW, IF
85 *   NON-ZERO, THE SWITCH IS TRANSFER T, IF ZERO,
86 *   THE SWITCH IS NORMAL N,
87 *   SW1 - SIGN CONDITION INDICATOR
88 *     N PLUS SIGN FOUND
89 *     T MINUS SIGN FOUND
90 *   SW2 - SIGN FOUND INDICATOR
91 *     T SIGN FOUND
92 *     N SIGN NOT FOUND
93 *   SW3 - RIGHT PARENTHESIS INDICATOR
94 *     T RIGHT PARENTHESIS FOUND
95 *   SW4 - STRING CLOSE INDICATOR
96 *     T STRING CLOSURE REQUIRED
97 *
98 *****
99   HDNG   MPX FORTRAN ** SUBSCRIPT DECOMPO
100  ABS _____ REF CORE
101 *
102 *   SYSTEM AND FORTRAN EQUATES
103 *
104 MEMRY EQU   4*320 REF CORE   MAXIMUM CORE SIZE
105 PHSIZ EQU   4*320   MAXIMUM PHASE SIZE
106 OVERL EQU   MEMRY-PHSIZ   PHASES 2-29 START
107 FCOM EQU   OVERL-22   FORTRAN COMM, TABLE
108 PHNTB EQU   FCOM-56   PHASE TABLE
109 ROLRX EQU   PHNTB-50   INTERPHASE CALL
110 *
111 *   FORTRAN COMMUNICATION AREA
112 *
113   ORG   FCOM   FORTRAN COMM AREA
114 SOFS BSS   1   START OF STRING
115 EOFS BSS   1   END OF STRING
116 SOFST BSS  1   START OF SYMBOL TABLE
117 SOFNS BSS  1   START OF NON-STMNT NOS,
118 SOFXT BSS  1   START OF SUBSC TEMPS
119 SOFGT BSS  1   START OF GENERATED TEMPS

```

120	EOFST	BSS	1	END OF SYMBOL TABLE
121	COMMON	BSS	1	NEXT AVAILABLE COMMON
122	CSIZE	BSS	1	SIZE OF COMMON
123	ERROR	BSS	1	OVERLAP ERROR
124	FNAME	BSS	2	PROGRAM NAME
125	SORF	BSS	1	SUBR (-) OR FUNC (+)
126	CCWD	BSS	1	CONTROL CARD WORD
127	*			BIT 15 TRANSFER TRACE
128	*			BIT 14 ARITHMETIC TRACE
129	*			BIT 13 EXTENDED PRECISION
130	*			BIT 12 LIST SYMBOL TABLE
131	*			BIT 11 LIST SUBPROGRAM NAMES
132	*			BIT 10 LIST SOURCE PROGRAM
133	*			BIT 9 ONE WORD INTEGERS
134	*			BIT 8 PUNCH
135	*			BIT 7 NONPROCESS PROGRAM
136	IOCS	BSS	1	IOCS CONTROL CARD WORD
137	*			
138	*			SEE PHASE ONE FOR BIT PATTERNS
139	*			
140	DFCNT	BSS	1	FILE TABLE WORD COUNT
141	*			
142	LCOMN	BSS	2	SIZE OF INSKEL COMMON
143	*			
144	ICCR	BSS	2	IOCS CONTROL CARD ERROR
145	*			
146		BSS	2	SYSTEM LOADER USE
147	*			
148	*			
149	*			END OF FORTRAN COMMUNICATION
150	*			AREA
151		HDNG		MPX FORTRAN ** SUBSCRIPT DECOMPO
152		ORG		OVERL PHASE ENTRY
153	*			CHECK FOR OVERLAP ERROR
154	*			
155	START	LD		ERROR IF OVLAP ERROR THEN BR TO
156		BSC	L	*RD IN NXT PH AND EXEC
157	*			
158	*			INITIALIZE PHASE
159	*			
160		LDX	I1	SOFS START OF FIRST STMT ADDR
161	*			
162	*			CHECK STATEMENT TYPE
163	*			
164	TEST	LD	1 0	STMT ID WORD
165		SRA	1	POSITION TO TEST STMT TYP
166		AND		EXTRACT STMT TYPE BITS
167		S		END STMT CONSTANT
168		BSC	L	OUT, - BR IF END TO RD NEXT PHASE
169		S		ERRC ERROR STMT CONSTANT
170		BSC	L	MOVE, - BR IF ERR TO GET NEXT STMN
171		S		FORMC FORMAT STMT CONSTANT
172		BSC	L	MOVE, - BR IF FORMAT TO NEXT STMT
173		S		CONTC CONTINUE STMT CONSTANT
174		BSC	L	MOVE, - BR IF CONTINUE TO GET NEXT
175		S		FIOC FIO STMT CONSTANT
176		BSC	L	MOVE, - BR IF FIO TO GET NEXT STMN
177		S		DFL DEFINE FILE STMT CONSTANT
178		BSC	L	MOVE, - BR IF DEFINE FILE- GET NEX
179		S		CEXT CALL EXIT OR LINK CONSTANT

```

180      BSC  L  MOVE, -   BR 10 GET NEXT IF EXIT=LIN
181      S      DATAS      DATA STMT CONSTANT
182      BSC  L  MOVE, -   BR 10 GET NEXT IF DATA
183      *
184      *      INITIALIZE 10 SCAN STATEMENT
185      *
186      SLA      16      ZERO ACCUMULATOR
187      STO      SW4      CLEAR STRING CLOSE SWITCH
188      STO  L  HOB      STRING CLOSE VALUE
189      STX  1  IDSAV     SAVE STMT START ADDRESS
190      LD  1  0      STMT ID WORD
191      SRA      2      RIGHT JUSTIFY STMT WD CNT
192      AND  L  IDNRM     EXTRACT WORD COUNT
193      STO  L  NRMSV     STMT WORD COUNT TEMPORARY
194      LD  1  0      STMT WORD COUNT
195      BSC      E      SKIP NEXT IF NO STMT NO,
196      MDX  1  1      INCR TO BYPASS STMT NUMBE
197      MDX  1  1      INCR STMT POINTER
198      *
199      *      CHK FOR STMT TERMINATOR SEMICOLON
200      *
201      TEST1 LD  1  0      NEXT STMT WORD
202      BSC  L  0, -     BR IF SEMICOLON
203      *
204      *      CHECK FOR DIMENSION SUBSCRIPTS
205      *
206      S      SJB1      ONE-DIMENSIONAL SUBSCRIPT
207      BSC  L  ABEL, -   BR IF ONE-DIMENSIONAL SIBS
208      S      SUB3      SUBSCRIPT CONSTANT
209      BSC  L  ABEL, -   BR IF TWO-DIMENSIONAL SJBS
210      S      SUB3      SUBSCRIPT CONSTANT
211      BSC  L  ABEL, -   BR IF 3-DIMENSIONAL SUBSC
212      L  MDX  1  1      MOVE STMT POINTER
213      MDX      TEST1     BR 10 TEST FOR SEMICOLON
214      *
215      *      CONSTANTS AND WORK AREA
216      *
217      SW1  DC      0      SIGN CONDITION INDR
218      SW2  DC      0      SIGN FOUND INDR
219      SW3  DC      0      RIGHT PARENTHESIS INDR
220      SW4  DC      0      STRING CLOSE REQUIRED INDR
221      IDTPE DC  /7C00    STMT TYPE MASK
222      ENDC  DC  /0800    END STMT CONSTANT
223      ERRC  DC  /5000-70800  ERROR STMT CONSTANT
224      FORMC DC  /3000-75000  FORMAT STMT CONSTANT
225      CONTC DC  /5800-73000  CONTINUE STMT CONSTANT
226      FIOC  DC  /6C00-75800  FIO STMT CONSTANT
227      DFL  DC  /7800-76C00  DEFINE FILE STMT CON
228      CEXT  DC  /7000-77800  CALL EXIT=LINK STMT CO
229      DATAS DC  /7C00-77000  DATA STMT CONSTANT
230      IDSAV DC  0      STMT ID ADDR TEMPORARY
231      SUB1  DC  /001A     ONE-DIMENSIONAL SUBSC CON
232      SUB3  DC  /0002     DIMENSION SUBSCRIPT CON
233      Y      DC  0      STMT POINTER TEMPORARY
234      MASK  DC  /07FF     SYMBOL TABLE POINTER MASK
235      SAVE  DC  0      SYMBOL TABLE ADDR TEMPORAR
236      THREE DC  /FFFF     MINUS 3 CONSTANT
237      IJMAX DC  0      FIRST DIMENSION FACTOR TEM
238      IMAX  DC  0      2ND DIMENSION FACTOR TEMP
239      BSS  E  0      MAKE ADDRESS EVEN

```

240	TEM	DC	0	1ST WD BINARY CONSTANT TEM
241		DC	0	2ND WD BINARY CONSTANT TEM
242	COT1	DC	/7E00	BINARY CON HIGH DIGIT MASK
243	COT2	DC	/7800	4 BIT DIGIT MASK
244	NINT	DC	/4800	DECIMAL 9 DIGIT LIMIT CON
245	HELP1	DC	0	DIGIT TEMPORARY
246	TEMPT	DC	0	PARTIAL PRODUCT TEMPORARY
247	MASK1	DC	/FFFF	ONE COMPLEMENT MASK
248	ASTK	DC	/0010	ASTERISK CONSTANT
249	SUMM	DC	0	CUMULATIVE CONVERTED SUM
250	ERR32	DC	32	ERROR 32 CONSTANT
251	*			
252	*			ARITHMETIC STATEMENT FUNCTION
253	*			
254	ABEL	LD	I	IDSAV
				STMNT ID WORD
255		SRA		11
				RIGHT JUSTIFY STMNT TYPE
256		S		SUB1
257		BSC	L	ABELL,Z
258		LD		ERR32
				ERR-SUBSC VAR IN STMNT FUN
259		STO	L	ERRNO
260		BSC	L	N
				BR 10 REPLACE STMNT W/ ERR
261	*			
262	*			GET DIMENSION INFORMATION
263	*			
264	ABELL	STX	1	Y
				SAVE STMNT POINTER
265		MDX	1	1
				INCR STMNT POINTER
266		SLA		16
				CLEAR ACCUMULATOR
267		STO		SW1
				CLEAR SIGN CONDITION INDR
268		STO		SW2
				CLEAR SIGN FOUND INDR
269		STO		SW3
				CLEAR RIGHT PAREN INDR
270		STO	L	G0
				CLEAR COMMA COUNT TEMP
271		STO	L	D4
				ZERO SUBSC CONSTANT 4
272		STO	L	CNT
				ZERO SUBSC ELEMENT COUNT
273		STO	L	D1
				ZERO SUBSC CONSTANT 1
274		STO	L	D2
				ZERO SUBSC CONSTANT 2
275		STO	L	D3
				ZERO SUBSC CONSTANT 3
276		STO	L	D4
				ZERO SUBSC CONSTANT 4
277		STO		A
				DIMENSION VALUE CON TEMP
278		STO		B
				DIMENSION VALUE CON TEMP
279		LD	1	-2
				SYMBOL TABLE POINTER WD
280		AND		MASK
				EXTRACT POINTER
281		M		THREE
				WORDS PER ENTRY
282		SLT		16
				MOVE TO ACCUMULATOR
283		A	L	SQFST
				START OF SYMBOL TABLE ADDR
284		STO		SAVE
				SAVE SYMBOL TABLE ADDR
285		LDX	13	SAVE
				XR3 SYMBOL TABLE ADDR
286		LD	3	1
287		STO		IJMAX
				1ST DIMENSION CONSTANT
288		LD	3	2
289		STO		IMAX
				2ND DIMENSION CONSTANT
290		LD	3	3
291		STO	L	HOLD
				3RD DIMENSION CONSTANT
292	*			
293	*			CHK FOR OPERATOR, NAME, OR CONSTANT
294	*			
295	G	LD	1	0
				NEXT STMNT WORD
296		BSC	L	M,-
				BR 10 SET ERR33 IF OPERATO
297		SLA		2
298		BSC	L	* 1,C
				BR 10 TEST SIGN IF CARRY 0
299		MDX		D
				BR IF NOT CARRY = NAME

300	BSC	L	D,-	GO CHK FOR LEGAL NAME
301	LDS		0	RESET STATUS INDRS
302	SLT		32	CLEAR ACC AND EXTENSION
303	STO		SJMM	ACCUMULATED CONVERTED CON
304	LD	1	0	FIRST CONSTANT WORD
305	STD		TEM	CONSTANT TEMPORARY
306	LD	1	1	NEXT STMT WORD
307	BSC	L	COLL,-	BR TO CONVERT TO BINARY
308	SLA		1	SHIFT OFF INDR BIT
309	STO		TEM 1	SECOND CONSTANT WORD
310	LD	1	2	NEXT STMT WORD
311	MDX	1	1	INCR STMT POINTER
312	BSC	L	M, Z	BR IF 3RD WORD = ERPOR 33
313	*			
314	*			COLLECT CONSTANT TO BINARY
315	*			
316	COLL	LD	TEM	FIRST CONSTANT WORD
317		AND	COT1	EXTRACT HIGH CHARACTER
318		SLA	2	
319		BSC	L PILL,C	BR TO TEST SIGN IF CARRY 0
320		BSC	L NEX, -	BR TO CHK SIGN CONDITION S
321		BSC	L M	BR ERROR = NON-DIGIT CHAR
322	PILL	BSC	L M,-	BR ERROR = NON-DIGIT CHAR
323		AND	COT2	EXTRACT BCD DIGIT
324		S	NINT	NINE
325		BSC	L M,-Z	BR IF GREATER THAN NINE-ER
326		A	NINT	RESTORE DIGIT
327		SRA	11	RIGHT JUSTIFY
328		STO	HELP1	UNITS DIGIT TEMPORARY
329		LD	SJMM	CUMULATIVE SUM OF CONSTANT
330		SLA	1	TIMES 2
331		STO	TEMPT	SAVE
332		BSC	L M, Z	BR IF TOO LARGE
333		SLA	2	TIMES 8
334		BSC	L M,C Z	BR IF TOO LARGE
335		A	TEMPT	TIMES 10
336		A	HELP1	UNITS DIGIT
337		BSC	L M,0 Z	BR IF TOO LARGE
338		STO	SJMM	NEW CUMULATIVE SUM
339		LDD	TEM	TWO-WD CONSTANT TEMPORARY
340		SLT	6	SHIFT OFF CONVERTED DIGIT
341		STD	TEM	RESTORE CONSTANT
342		MDX	COLL 1	BR TO CONVERT NEXT DIGIT
343	NEX	LD	L SW1	SIGN CONDITION SWITCH
344		BSC	L C,Z	BR IF MINUS SIGN
345		LD	SJMM	CONVERTED CONSTANT
346		MDX	C 3	BR TO STORE IN B
347	C	LD	SJMM	CONVERTED CONSTANT
348		EOR	MASK1	ONES COMPLIMENT
349		A	ONE	TWOS COMPLIMENT -CONSTAN
350		STO	B	PROPERLY SIGNED CONSTANT
351		SLA	16	
352		STO	L SW1	RESET SIGN CONDITION SWITC
353		MDX	1 1	INCR STMT POINTER
354	*			
355	*			CHECK IF FIRST OR SECOND CONSTANT
356	*			
357		LD	1 0	NEXT STMT WORD
358		S	ASTK	ASTERISK CONSTANT
359		BSC	L F,Z	BR TO CHK IF COMMA

```

360          LD      B          PREVIOUS CON WAS 1ST OF 2
361          STO      A          STORE AS A
362          SLA      16
363          STO      B          CLEAR B
364          MDX      1 1        INCR STMT POINTER
365 *
366 *          CHECK FOR VARIABLE NAME
367 *
368          LD      1 0          NEXT STMT WORD
369          BSC     L M,-        BR ERROR IF NOT NAME
370          SLA      2
371          BSC     L POS,C      BR IO TEST SIGN IF CARRY 0
372          MDX      TV          BR IO TEST SIGN FOUND SW
373 POS       BSC     L TV,-     BR IF ALPHA CHAR = NO ERRO
374          BSC     L M          BR IO SET UP ERROR 33
375 *
376 *          MOVE TO NEXT STMT
377 *
378 MOVE     LD      1 0          NEXT STMT WORD
379          SRA      2
380          AND      IDNRM       EXTRACT STMT WORD COUNT
381          STO      NXID 1     ADDR NEXT INSTRUCTION
382 NXID     MDX     L1 0        INCR TO NEXT STMT ID WORD
383 *          *MODIFIABLE
384          BSC     L TEST      BR IO SCAN NXT STMT
385 *
386 *          CONSTANTS AND WORK AREA
387 *
388 ON       DC      /8000      SIGN BIT CONSTANT
389 B        DC      0          DIMENSION VALUE CON TEMP
390 A        DC      0          DIMENSION VALUE CON TEMP
391 IDNRM    DC      /01FF     STMT WD COUNT EXTRACT MAS
392 ONE      DC      1          DECIMAL ONE CONSTANT
393 WORD1    DC      0          NAME CONVERSION TEMP-1ST W
394 WORD2    DC      0          NAME CONVERSION TEMP-2ND W
395 SAVE1    DC      0          INDEX REG 3 TEMPORARY
396 *
397 *          CHECK FOR LEGAL VARIABLE NAME
398 *
399 D        LD      ONE
400          STO      A          NO CON BEFORE NAME, A 1
401 TV       LD      L SW2      SIGN FOUND SWITCH
402          BSC     L M,Z      BR IF SIGN FOUND
403          LD      ON         PRESET WORD TWO FOR ONE
404          STO      WORD2     WORD NAME,
405          LD      1 0          NEXT STMT WORD
406          STO      WORD1     FIRST NAME WORD
407          LD      1 1          NEXT STMT WORD
408          BSC     L CK1,-     BR IF ONE WORD ONLY
409          STO      WORD2     SECOND NAME WORD
410          MDX      1 1        INCR STMT POINTER
411          LD      1 1          NEXT STMT WORD
412          BSC     L M, Z      BR IF 3RD NAME WD - ERROR
413 *
414 *          CHECK SYMBOL TABLE FOR DUPLICATE NAM
415 *
416 CK1      LDX     I3 SOFNS    START OF NON-STMT NUMBERS
417          MDX      JACK      BR IO INITIATE
418 LPP      LD      3 1        NEXT NAME = FIRST WORD
419          S        WORD1     CHECK FOR DUPLICATE

```

420	BSC	Z		SKIP NEXT IF 1ST WDS EQUAL
421	MDX	EAT		BR TO CHK AGAINST NEXT NAM
422	LD	3 2		COMPARE SECOND WORDS OF
423	S	WORD2		NAME IF FIRST WDS COMPARE,
424	BSC	L CK3, -		BR TO CONVERT IF NAME FOUND
425	EAT	MDX 3 -3		DECR TO NEXT ENTRY
426	JACK	STX 3 SAVE1		XR3 TO TEMPORARY
427		LD SAVE1		CURRENT SYMBOL TBL ADDRESS
428	S	L EOFST		END OF SYMBOL TABLE
429	BSC	Z		SKIP NEXT IF END OF TABLE
430	MDX	LPP		CONTINUE NAME SEARCH
431	LD	ERR34		UNDEFINED VARIABLE IN SUBS
432	STO	ERRNO		SET UP ERROR NO, 34
433	MDX	N		BR TO PUT ERROR ON STRING
434	*			
435	*			CONSTANTS AND WORK AREA
436	*			
437	ERR34	DC 34		ERROR 34 CONSTANT
438	ERRNO	DC 0		ERROR NUMBER TEMPORARY
439	THRE	DC 3		DECIMAL 3 CONSTANT
440	DEE	DC 0		DIMENSIONAL SUBSC NAME TEM
441	EEE	DC 0		DIMENSIONAL SUBSC NAME TEM
442	NAME	DC 0		CONVERTED NAME TEMPORARY
443	CEE	DC 0		DIMENSIONAL SUBSC NAME TEM
444	MARS	DC /0800		DIMENSION INDR BIT MASK
445	PLUSC	DC /000E		PLUS CONSTANT
446	MINC	DC /0020-/000E		MINUS CONSTANT
447	COMMC	DC /002B		COMMA CONSTANT
448	SPECP	DC /0002-/002B		SPECIAL RT PARENTHESIS
449	ERR33	DC 33		ERROR 33 CONSTANT
450	NRMSV	DC 0		STMNT WD COUNT TEMPORARY
451	GETZ	DC 0		NEXT STMNT ADDR TEMPORARY
452	ERRID	DC /A008		ERROR IDENTIFIER CONSTANT
453	RNG	DC 0		STRING CLOSURE RANGE
454	HUB	DC /4000		BIT 1 MASK
455	ERIID	DC /A000		ERROR ID CONSTANT
456	*			
457	*			CONVERT NAME TO POINTER
458	*			
459	CK3	STX 3 SAVE1		CURRENT SYMBOL TABLE ADDR
460		LD L SOFST		START OF SYMBOL TABLE
461		A THRE		ADJUST FOR SEARCH LOOP DEC
462		S SAVE1		ADDR WHERE NAME FOUND
463		SRT 16		MAKE NORMAL DIVIDEND
464		D THRE		FORM SYMBOL TABLE POINTER
465		OR ON		SIGN BIT
466		STO NAME		NAME POINTER
467		LD DEE		
468		STO CEE		PUSH DOWN DIMENSIONAL
469		LD EEE		SUBSCRIPT NAME POINTERS AN
470		STO DEE		ADD THE NAME JUST CONVERTE
471		LD NAME		TO THE TOP OF THE LIST,
472		STO EEE		
473	*			
474	*			CHECK FOR NON-DIMENSIONED NAME
475	*			
476	E	LD I SAVE1		SYMBOL TABLE ID WORD
477		AND MARS		DIMENSION BIT MASK
478		EOR HUB		REAL/INTEGER BIT MASK
479		BSC L N,Z		BR IF NON-DIMENSIONED NAME



```

480 MDX 1 1 INCR STMT POINTER
481 *
482 * CHECK FOR OPERATOR
483 *
484 LD 1 0 NEXT STMT WORD
485 S PLUSC PLUS SIGN CONSTANT
486 BSC L CK4, - BR IF PLUS SIGN
487 S MINC MINUS SIGN CONSTANT
488 BSC L CK5, - BR IF MINUS SIGN
489 F LD 1 0 NEXT STMT WORD
490 S COMMC COMMA CONSTANT
491 BSC L CK6, - BR IF COMMA
492 S SPECPC SPECIAL RIGHT PARENTHESIS
493 BSC L CK7, - BR IF RIGHT PARENTHESIS
494 *
495 * SET UP ERROR 33
496 *
497 M LD ERR33 ERROR 33 CONSTANT
498 STO ERRNO ERROR NUMBER TEMPORARY
499 *
500 * REPLACE STMT WITH ERROR
501 *
502 N LDX I1 IDSAV STMT ID WORD ADDRESS
503 LD 1 0 STMT ID WORD
504 SRA 2 RIGHT JUSTIFY WD COUNT
505 AND IDNRM EXTRACT WORD COUNT
506 STO NRMSV SAVE WORD COUNT
507 A L IDSAV STMT ID WORD ADDRESS
508 STO GETZ ADDR NEXT STMT ID WORD
509 LDX I2 GETZ XR2 NEXT STMT ADDRESS
510 LD 1 0 STMT ID WORD
511 BSC L CALD,E BR IF NUMBERED STMT
512 MDX CALD 4 BR IF UNNUMBERED STMT
513 CALD LD ERIID ERROR ID WORD
514 STO 1 0 STORE AS STMT ID
515 MDX 1 1 INCR STMT POINTER
516 MDX CALDA BR IO SET UP ERR ON STRING
517 LD ERRID ERROR ID WORD
518 STO 1 0 STORE AS STMT ID
519 CALDA LD ERENO ERROR NUMBER
520 STO 1 1 STORE ON STRING
521 MDX 1 1 INCR STMT POINTER
522 LD L EOFS END OF STRING ADDRESS
523 S L IDSAV CURRENT STMT ADDRESS
524 S NRMSV CURRENT STMT WORD COUNT
525 STO RNG
526 LDX I3 RNG RANGE OF STRING CLOSURE
527 STX 1 RNG CURRENT STMT POINTER
528 MDX 3 1 ADJUST RANGE
529 LIM LD 2 0 NEXT WORD TO MOVE
530 STO 1 1 CLOSE UP STRING
531 MDX 1 1 INCR MOVE TO POINTER
532 MDX 2 1 INCR MOVE FROM POINTER
533 MDX 3 -1 DECK NO, WDS TO MOVE
534 MDX LIM BR IO CONTINUE CLOSING
535 STX L1 EOFS END OF STRING ADDRESS
536 LDX I1 RNG XR1 CURRENT STMT POINTER
537 MDX 1 1 INCR STMT POINTER
538 BSC L TEST BR IO TEST NEXT STMT TYPE
539 *

```

```

540 *          SET SWITCHES 10 INDICATE SIGN FOUND
541 *
542 CK4  SLA      16      CLEAR ACCUMULATOR
543      STO L SW1      SET SW1 TO PLUS CONDITION
544      MDX L SW2,1    SET SW2 SIGN FOUND
545      MDX 1 1      INCR STMT POINTER
546      BSC L G       BR 10 CHK FOR OPERATOR
547 CK5  MDX L SW1,1   SET SW1 TO MINUS CONDITION
548      MDX      CK4 3   BR 10 SET SWITCH 2
549 *
550 *          WORK AREA
551 *
552 GO    DC      0      DIMENSION LEVEL TEMPORARY
553 D4    DC      0      DIMENSION CONSTANT TEMP
554 *
555 *          DETERMINE DIMENSIONING LEVEL
556 *
557 CK7  MDX L SW3,1    SET SW3 RIGHT PARENTHESI
558      STX 1 VAMP     SAVE INDEX REG 1
559 CK6  MDX L GO,1    INCR DIMENSION LEVEL CTR
560      LD    GO      DIMENSION LEVEL CTR
561      S     L ONE    DECIMAL ONE
562      BSC L CK8, -   BR IF 1-DIMENSION
563      S     L ONE    DECIMAL ONE
564      BSC L CK9, -   BR IF 2-DIMENSION
565      S     L ONE    DECIMAL ONE
566      BSC L N,Z     BR 10 ERROR IF MORE THAN 3
567      LD    SUBB3   3 DIMENSION INDR CONSTANT
568      STO   TAB     DIMENSION LEVEL INDR
569 *
570 *          CALCULATE D4 D4 IJ
571 *
572      LD    L B      DIMENSION VALUE CONSTANT
573      M     L IJMAX  FIRST DIMENSION FACTOR
574      SLT   16      PRODUCT INTO ACCUMULATOR
575      S     L IJMAX  FIRST DIMENSION FACTOR
576      A     D4      D4 SUBSCRIPT CONSTANT
577      STO   D4      NEW D4
578 *
579 *          CALCULATE A A*IJMAX
580 *
581      LD    L A      DIMENSION VALUE CONSTANT
582      M     L IJMAX  FIRST DIMENSION FACTOR
583      SLT   16      PRODUCT INTO ACCUMULATOR
584      STO   L A      NEW A VALUE
585      MDX   CK10    BR 10 CHECK IF A 0
586 *
587 *          CALCULATE D4 D4 B-1
588 *
589 CK8  LD    L SUB1   ONE-DIMENSIONAL SUBSC CON
590      STO   TAB     DIMENSIONAL LEVEL INDR
591      LD    L D4     D4 SUBSC CONSTANT
592      A     L B      DIMENSION VALUE CONSTANT
593      S     L ONE    DECIMAL ONE
594      STO   D4      NEW D4
595      MDX   CK10    BR 10 CHECK IF A 0
596 *
597 *          CALCULATE D4 D4 IMAX B-1
598 *
599 CK9  LD    SUBB2   TWO-DIMENSIONAL SUBSC CON

```

600		STO		TAB		DIMENSIONAL LEVEL INDR
601		LD	L	B		DIMENSION VALUE CONSTANT
602		M	L	IMAX		2ND DIMENSION FACTOR
603		SLT		16		PRODUCT INTO ACCUMULATOR
604		S	L	IMAX		2ND DIMENSION FACTOR
605		A		D4		D4 SUBSCRIPT CONSTANT
606		STO		D4		NEW D4
607	*					
608	*			CALCULATE	A A*IMAX	
609	*					
610		LD	L	A		DIMENSION VALUE CONSTANT
611		M	L	IMAX		2ND DIMENSION FACTOR
612		SLT		16		PRODUCT INTO ACCUMULATOR
613		STO	L	A		NEW A VALUE
614	*					
615	*			CHECK FOR	A 0	
616	*					
617	CK10	LD	L	A		DIMENSION VALUE CONSTANT
618		BSC	L	J, -		BR IF A 0 TO TEST IF B 0
619	*					
620	*			SET A INTO	NEXT SUBSCRIPT CONSTANT	
621	*					
622		MDX	L	CNT,1		INCR SUBSC ELEMENT COUNTER
623		LD		D2		D1 D2
624		STO		D1		
625		LD		D3		D2 D3
626		STO		D2		
627		LD	L	A		D3 A
628		STO		D3		
629		MDX		JK		BR TO CHK RIGHT PAREN INDR
630	J	LD	L	B		DIMENSION VALUE CONSTANT
631		BSC	L	M, -		BR IF B ZERO
632	*					
633	*			TEST RIGHT	PARENTHESIS INDICATOR	
634	*					
635	JK	LD	L	SK3		RT PARENTHESIS INDICATOR
636		BSC	L	CK11,Z		BR IF RT PARENTHESIS
637		MDX	1	1		MOVE STMT POINTER
638		SLA		16		CLEAR ACCUMULATOR
639		STO	L	SK2		RESET SIGN INDICATOR
640		STO	L	A		CLEAR DIMENSION VALUE CON
641		STO	L	B		CLEAR DIMENSION VALUE CON
642		BSC	L	G		BR TO CHK FOR OPERATOR
643	*					
644	*			CONSTANTS	AND WORK AREA	
645	*					
646	TAB	DC		0		DIMENSIONAL LEVEL INDICATO
647	SUBB2	DC		/001C		2 DIMENSION INDR CONSTANT
648	SUBB3	DC		/001E		3 DIMENSION INDR CONSTANT
649	CNT	DC		0		SUBSCRIPT ELEMENT COUNT
650	ERR35	DC		35		ERROR 35 CONSTANT
651	FOUR	DC		4		DECIMAL 4 CONSTANT
652	OP	DC		0		
653	NO	DC		0		
654	SUBZ	DC		/001B		ZERO SUBSC INDR CONSTANT
655	TWO	DC		2		
656	SPARR	DC		/0024		SPECIAL RT PARENTHESIS CON
657	VAMP	DC		0		
658	D1	DC		0		
659	D2	DC		0		SUBSC CONSTANT D2

```

660 D3 DC 0 SUBSC CONSTANT D3
661 EQVIC DC /15 EQUIVALENCE STMT CONSTANT
662 *
663 * DO SUBSCRIPTS AGREE
664 *
665 CK11 LD I Y STMT SUBSC WORD
666 S TAB DIMENSION LEVEL INDICATOR
667 BSC L CK12, - BR IF SUBSCRIPTS AGREE
668 *
669 * EQUIVALENCE STATEMENT
670 *
671 LD I IDSAV STMT ID WORD
672 SRA 11 RT JUSTIFY STMT TYPE
673 S EQVIC EQUIVALENCE STMT CONSTANT
674 BSC L CKKK,Z BR IF NOT EQUIVALENCE STMT
675 *
676 * CHECK IF 1 DIMENSIONAL STMT
677 *
678 LD L TAB DIMENSION LEVEL INDICATOR
679 S L SUB1 ONE-DIMENSIONAL SUBSC CON
680 BSC L CK12, - BR IF ONE DIMENSION SUBSC
681 LDX 2 31 XR2 31
682 STX L2 ERRNO SET UP ERROR NO, 31
683 MDX CKKK 4 BR TO REPLACE STMT W/ ERR
684 CKKK LD L ERR35 ERROR 35 CONSTANT
685 STO L ERRNO SET UP ERROR NO, 35
686 BSC L N BR TO REPLACE STMT W/ ERR
687 CK12 LD CNT SUBSCRIPT ELEMENT COUNT
688 M FOUR CNT CNT*4
689 SLT 16 PRODUCT INTO ACCUMULATOR
690 STO CNT RESIORE CNT
691 *
692 * COMPUTE SIZE OF SUBSCRIPT EXPRESSION
693 *
694 LD L VAMP STMT POINTER END
695 S L Y STMT POINTER BEGINNING
696 STO NO STORE SIZE-1
697 MDX L NO,1 SUBSCRIPT EXPRESSION SIZE
698 *
699 * COMPUTE RANGE OF STMT CLOSURE
700 *
701 LD CNT SUBSCRIPT ELEMENT COUNT
702 S NO SUBSCRIPT EXPRESSION SIZE
703 A L THREE THREE
704 STO OP RANGE OF CLOSURE
705 BSC L CK13, Z BR IF RANGE IS NEGATIVE
706 BSC L CK14,Z BR IF RANGE IS POSITIVE
707 K LDX I1 Y STMT PI Y IF RANGE ZERO
708 *
709 * COMPUTE NO. OF WORDS PER ENTRY
710 *
711 LD HOLD 3RD DIMENSION CONSTANT
712 SLA 1 REAL-INTEGER INDR INTO SIG
713 BSC L HOLD1,- BR IF REAL
714 LD L CCWD CONTROL CARD WORD
715 SLA 9 ONE-WD INTEGER BIT TO SIGN
716 BSC Z SKIP NEXT IF NOT ONE WORD
717 MDX HLD2 BR IF ONE WORD INTEGER MOD
718 HOLD1 LD L CCWD CONTROL CARD WORD
719 SLA 13 EXTENDED PREC BIT INTO SIG

```

```

720      BSC      Z      SKIP NEXT IF STANDARD PREC
721      MDX      ZZZZ-2  BR IF EXTENDED PRECISION
722      LD       TWO
723      MDX      ZZZZ      BR 10 SET WDS PER ENTRY 2
724  HLD2  LD      L  ONE      BR 10 SET WDS PER ENTRY 1
725      MDX      ZZZZ      WORDS PER ENTRY 3
726      LD      L  THREE     NO. OF WORDS PER SUBSCRIPT
727  ZZZZ  STO      SES
728  *
729  *      SET UP D1, D2, D3, AND D4
730  *
731      LDX      2 3      LOOP COUNTER
732      LD      L2 D1-1    D-FACTOR
733      M        SES      NO. OF WORDS PER SUBSCRIPT
734      SLT      16      PRODUCT INTO ACCUMULATOR
735      STO      L2 D1-1    COMPUTED DIMENSION INDEX
736      MDX      2 -1      DECR COUNTER
737      MDX      ZZZZ 2    BR 10 COMPUTE NEXT D FACTO
738      LD      D1      TEST IF D1, D2, D3
739      OR      D2      *ARE ALL ZERO
740      OR      D3      *
741      BSC      L  D4OK,Z  BRANCH IF NOT ALL ZERO
742  *
743  *      TEST FOR SUBSCRIPT SIZE ERROR
744  *
745      LD      I  SAVE     LOAD MAXIMUM SUBSCRIPT
746      S      L  D4      SUBTRACT THIS SUBSCRIPT
747      BSC      L  D4OK,-Z  BRANCH IF SIZE OK
748      LD      ERR35     LOAD ERROR CODE
749      BSC      L  H 1    GO INDICATE ERROR
750  D4OK  EQU      *      CONTINUE
751  *
752  *      PUT D4 ON STRING AT POINTER 1
753  *
754      LD      L  D4      D-FACTOR
755      M        SES      NO. OF WORDS PER SUBSCRIPT
756      SLT      16      PRODUCT INTO ACCUMULATOR
757      OR      L  ON      SIGN BIT
758      STO      L  D4      COMPUTED DIMENSION INDEX
759      STO      1 1      PLACE ON STRING
760  *
761  *      PLACE DIMENSION SUBSC ON OUTPUT STRIN
762  *
763      LD      CNT      SUBSC ELEMENT COUNT
764      BSC      L  Z,Z    BR IF NOT ZERO
765      LD      SUBZ     ZERO SUBSC INDR
766      STO      1 0      PLACE ON STRING
767  Z3    LD      L  Y      COMPUTE POINTER Y CNT 2
768      A      CNT      ELEMENT COUNT
769      A      TWO      DECIMAL TWO
770      STO      TAT 1    ADDR OF LDX INSTRUCTION
771  TAT   LDX      L1 0    POINTER Y CNT 2
772  *      *MODIFIABLE
773  *
774  *      PUT SPECIAL R) PARENTHESIS ON STRING
775  *
776      LD      SPARR     SPECIAL RT PARENTHESIS
777      STO      1 0      PLACE ON STRING
778      BSC      L  L      BR 10 CHK FOR END OF STYNT
779  *

```

```

780 *          CONSTANTS AND WORK AREA
781 *
782 EIGHT DC      8          CONSTANT
783 HOLD DC       0          3RD DIMENSION FACTOR TEMP
784 SES DC        0          NO. OF WORDS PER SUBSC TEM
785 COMS DC       /0012     COMMA CONSTANT
786 *
787 *          CHECK FOR D1 0
788 *
789 Z           LD          D1          D-FACTOR 1
790           BSC L Z1,Z          BR IF NOT ZERO
791           LD L D2          D-FACTOR 2
792           STO L D1          D1 D2
793           LD L D3          D-FACTOR 3
794           STO L D2          D2 D3
795           LD L DEE
796           STO L CEE        C D
797           LD L EEE
798           STO L DEE        D E
799           MDX           Z          BR TO CHK D1 FOR ZERO
800 Z1          LD L CNT          ELEMENT COUNT
801           S L FOUR        DECIMAL FOUR
802           BSC L Z4,Z          BR TO Z4 IF GREATER THAN 4
803           LD L SUB1        1 DIMENSIONAL SUBSC INDR
804           STO 1 0          PLACE ON STRING
805           MDX 1 2          INCR POINTER BY 2
806 *
807 *          PLACE ,C,D1 ON STRING
808 *
809 Z5          LD          COMS        COMMA CHARACTER CONSTANT
810           STO 1 0          FIRST COMMA
811           STO 1 2          SECOND COMMA
812           LD L D1
813           STO 1 3          D1
814           LD L CEE
815           STO 1 1          C
816           MDX           Z3        BR TO COMPUTE POINTER
817 Z4          LD          EIGHT      DECIMAL EIGHT
818           S L CNT          ELEMENT COUNT
819           BSC L Z6,Z          BR TO Z6 IF GREATER THAN 8
820           LD L SUB2        2 DIMENSIONAL SUBSC INDR
821           STO 1 0          PLACE ON STRING
822           MDX 1 6          INCR POINTER BY 6
823 *
824 *          PUT ,D,D2 ON STRING
825 *
826 Z7          LD          COMS        COMMA CONSTANT
827           STO 1 0          FIRST COMMA
828           STO 1 2          SECOND COMMA
829           LD L D2
830           STO 1 3          D2
831           LD L DEE
832           STO 1 1          D
833           MDX 1 -4          MOVE POINTER BY =4
834           MDX           Z5        BR TO PUT ,C,D1 ON STRING
835 Z6          LD L SUB3        3 DIMENSIONAL SUBSC INDR
836           STO 1 0          PLACE ON STRING
837           MDX 1 10         INCR POINTER BY 10
838 *
839 *          PUT ,E,D3 ON STRING

```

840	*							
841		LD		CONS			COMMA CHARACTER CONSTANT	
842		STO	1	0			FIRST COMMA	
843		STO	1	2			SECOND COMMA	
844		LD	L	D3				
845		STO	1	3			D3	
846		LD	L	EEE				
847		STO	1	1			E	
848		MDX	1	-4			MOVE POINTER BY -4	
849		MDX		Z7			BR IO PUT ,D,D2 ON STRING	
850	*							
851	*						COMPUTE STRING CLOSE VALUE	
852	*							
853	CK13	LD		HQB			STRING CLOSE VALUE	
854		S	L	OP			STMT CLOSURE RANGE	
855		STO		HQB			ADJUSTED STRING CLOSE VALU	
856		MDX	L	S#4,1			SET STRING CLOSE SWITCH ON	
857	*							
858	*						CLOSE STMT OF WORDS	
859	*							
860		LD	L	Y			ADDR BEGIN SUBSCRIPT STMT	
861		S	L	OP			RANGE OF STMT CLOSURE	
862		STO		* 1				
863		LDX	L2	0			MOVE FROM ADDRESS	
864	*						*MODIFIABLE	
865		LDX	I1	Y			MOVE TO ADDRESS	
866	CLOS	LD	2	1			NEXT WORD TO MOVE	
867		STO	1	1			CLOSE UP STMT	
868		BSC	L	* 3, -			BR IF END OF STMT FOUND	
869		MDX	1	1			INCR MOVE TO ADDR	
870		MDX	2	1			INCR MOVE FROM ADDR	
871		MDX		CLOS			BR IO CONTINUE CLOSING	
872		BSC	L	K			BR IO SET STMT POINTER Y	
873	*							
874	*						OPEN STRING OF WORDS	
875	*							
876	CK14	STX	1	CRIK			CURRENT STMT POINTER	
877		LDX	I1	EOFS			END OF STRING ADDRESS	
878		LDX	I2	EOFS			END OF STRING ADDRESS	
879		MDX	I2	OP			NEW END OF STRING ADDRESS	
880		LD	L	EOFS			END OF STRING ADDRESS	
881		S		CRIK			CURRENT STMT POINTER	
882		STO		CRIK1 1			DIFFERENCE AS ADDR NEXT	
883	CRIK1	LDX	L3	0			MODIFIABLE	
884		MDX	3	1			NO. OF WORDS TO MOVE	
885		STX	L2	EOFS			NEW END OF STRING ADDRESS	
886	LAMB	LD	1	0			NEXT WORD TO MOVE	
887		STO	2	0			OPEN STRING	
888		MDX	1	-1			DECR MOVE FROM ADDRESS	
889		MDX	2	-1			DECR MOVE TO ADDRESS	
890		MDX	3	-1			DECR NO. OF WORDS TO MOVE	
891		MDX		LAMB			BR IO CONTINUE MOVE	
892		LDX	I1	IDS AV			ADDR OF STMT ID WORD	
893		LD	1	0			STMT ID WORD	
894		AND		NAZZ			EXTRACT STMT WORD COUNT	
895		STO		CRIK			SAVE ID WITHOUT WORD COUNT	
896		LD	L	NRMSV			OLD WD COUNT	
897		A	L	OP			INCR WD COUNT BY RANGE	
898		STO	L	NRMSV			NEW WD COUNT	
899		SLA		2			POSITION FOR ID WD FORMAT	

900	OP			CRIK	NEW WD COUNT IN ID WORD
901	STO	1	0		RESTORE ID WITH NEW WD CNT
902	LD	L		EOFST	LOAD END OF SYMBOL TABLE
903	S	L		EOFS	SUBTRACT END OF STRING
904	BSC	L		K,-Z	BRANCH IF NO OVERLAP
905	MDX	L		ERPOR,1	SET OVERLAP INDICATOR ON
906	MDX			OUT	EXIT
907	*				
908	*				CONSTANTS AND WORK AREA
909	*				
910	HOB	DC		0	STRING CLOSE VALUE
911	CRIK	DC		0	STMNT POINTER TEMPORARY
912	MAZZ	DC		/F803	STMNT ID WD EXTRACT MASK
913	ZAP1	DC		0	STMNT WORD COUNT TEMPORARY
914	ZAP	DC		0	INDEX REG 1 TEMPORARY
915	*				
916	*				CHECK SWITCH 4 FOR STRING CLOSURE
917	*				
918	0	LD	L	SW4	STRING CLOSE SWITCH
919		BSC	L	ZAPZ,Z	BR IF STRING CLOSE REQUIRE
920	ZAP6	MDX	1	1	INCR STMNT POINTER
921		BSC	L	TEST	GO TO NEXT STMNT
922	*				
923	*				CLOSE STRING BY HUB WORDS
924	*				
925	ZAPZ	STX	1	ZAP	SAVE CURRENT STMNT POINTER
926		LDX	I1	IDSAV	ADDR OF STMNT ID WORD
927		LD	1	0	STMNT ID WORD
928		AND		MAZZ	EXTRACT STMNT WORD COUNT
929		STO		CRIK	SAVE ID WITHOUT WORD COUNT
930		LD	L	NRMSV	OLD STMNT WORD COUNT
931		S		HOB	STRING CLOSURE RANGE
932		SLA		2	POSITION FOR ID WORD FORMA
933		OR		CRIK	NEW STMNT WORD CNT IN ID.W
934		STO	1	0	RESTORE ID WITH NEW WD CNT
935		LD	L	NRMSV	OLD WORD COUNT
936		A	L	IDSAV	ADDRESS STMNT ID WORD
937		STO		ZAP1	
938		LDX	I2	ZAP1	MOVE FROM ADDRESS
939		LD	L	EOFS	END OF STRING ADDRESS
940		S		ZAP1	MOVE FROM ADDRESS
941		STO		ZAP2 1	
942	ZAP2	LDX	L3	0	RANGE=1 OF MOVE, MODIFIABL
943		MDX	3	1	NO, OF WORDS TO MOVE
944		LDX	I1	ZAP	MOVE TO ADDRESS
945	ZAP4	LD	2	0	NEXT WORD TO MOVE
946		STO	1	1	CLOSE UP STRING
947		MDX	1	1	INCR MOVE TO ADDRESS
948		MDX	2	1	INCR MOVE FROM ADDRESS
949		MDX	3	-1	DECR NO, OF WDS TO MOVE
950		MDX		ZAP4	BR TO CONTINUE MOVE
951		LDX	I1	ZAP	RESTORE INDEX REG 1
952		MDX		ZAP6	BR TO INCR STMNT POINTER
953	*				
954	*				READ IN AND GO TO NEXT PHASE
955	*				
956	OUT	BSI	L	ROLRX	CALL DOWN PHASE 12
957		DC		12	NEXT PHASE NUMBER
958		BSS		OVERL-++320*3	PHASE-11 PATCH AREA
959		END		START	