

◀◀◀◀◀◀◀◀

OTL

INDEX

Address	Hex	Assembly	Page
7	000000		1
8	000000	INDEX	1
9	000000	ENTRY POINTS AND EXTERNALS	2
10	000000	FOR STMT	4
11	000000	TO, STEP STMTS	5
12	000000	NEXT STMT	7
13	000000	ON STMT	9
14	000000	ON KEY# OFF KEY#	10
15	000000	PRINT STMT	10
16	000000	ROUTINE CVNUM	12
17	000000	TAB	26
18	000000	RESTORE STMT	27
19	000000	READ STMT	28
20	000000	ROUTINE GETNAM	29
21	000000	SUB PARAM PASS	30
22	000000	SUB PASS CNTL	30
23	000000	SUB RETURN	31
24	000000	STR CONCATENATION	34
25	000000	FIND	36
26	000000	CHAIN STMT	36

>>>> ENTRY POINTS AND EXTERNALS <<<<<<<<

1	000000	ENT ARITH	
29	000000	ENT CHAIN.	
30	000000	ENT CHAIN+	Post-processing (cf MS rom)
31	000000	ENT CKLOOP	
32	000000	ENT COMMA*	
33	000000	ENT COMMA.	
34	000000	ENT CONCA.	
35	000000	ENT CVNUM	
36	000000	ENT CVNUMG	
37	000000	ENT DISP.	
38	000000	ENT DISP.+	
39	000000	ENT EXHXT	
40	000000	ENT FOR.	
41	000000	ENT FORMG	
42	000000	ENT FORMN	
43	000000	ENT FORMN+	
44	000000	ENT GCHAR-	
45	000000	ENT INIPGM	
46	000000	ENT KCLR	
47	000000	ENT KCLR-	
48	000000	ENT LEGDUN	
49	000000	ENT NEXT.	
50	000000	ENT OFFER.	
51	000000	ENT OFKEY.	
52	000000	ENT ON.	
53	000000	ENT ONERR.	
54	000000	ENT ONKEY.	
55	000000	ENT PEOL	
56	000000	ENT PRINT.	
57	000000	ENT PRIN.+	
58	000000	ENT PRLINE	
59	000000	ENT PRLIN2	
60	000000	ENT PRNFMT	
61	000000	ENT READN.	
62	000000	ENT READS.	
63	000000	ENT READ.	
64	000000	ENT RESTL.	
65	000000	ENT RESTO.	
66	000000	ENT RESUL.	
67	000000	ENT ROUND	
68	000000	ENT RSTO..	
69	000000	ENT SCANLX	
70	000000	ENT SEMIC*	
71	000000	ENT SEMIC.	
72	000000	ENT SENDCR	
73	000000	ENT SRCHTK	
74	000000	ENT STEP.	
75	000000	ENT TO.	
76	000000	ENT ULIN#.	
77	000000	ENT USING.	
78	000000	ENT USINL.	
79	000000		
80	000000		

* EXTERNALS

ENTRY POINTS AND EXTERNALS

<<<<<<<<

81	000000	EXT ADD9
82	000000	EXT ALINI+
83	000000	EXT ASIGNL
84	000000	EXT PRNT#5
85	000000	EXT COPSUB
86	000000	EXT PRNT#N
87	000000	EXT READ#N
88	000000	EXT READ#5
89	000000	EXT ROMINI
90	000000	EXT RESFIL
91	000000	EXT RESFL1
92	000000	EXT RM+SET
93	000000	EXT RSETG-
94	000000	EXT RSPTRS
95	000000	EXT CONINT
96	000000	EXT DCSLOP
97	000000	EXT DCSTR1
98	000000	EXT DIGIT
99	000000	EXT DRV12.
100	000000	EXT EMOVDN
101	000000	EXT EMOVUP
102	000000	EXT ERROR
103	000000	EXT ERROR+
104	000000	EXT FETADD
105	000000	EXT FETSYA
106	000000	EXT FNDL2
107	000000	EXT FTADR
108	000000	EXT FNUM
109	000000	EXT HLFLIN
110	000000	EXT ICONST
111	000000	EXT INFR4
112	000000	EXT INTCON
113	000000	EXT INTIUL
114	000000	EXT MATCH?
115	000000	EXT HXTLIN
116	000000	EXT NO.CRS
117	000000	EXT ONEI
118	000000	EXT ONE9
119	000000	EXT ONER
120	000000	EXT RELNEM
121	000000	EXT RESNEM
122	000000	EXT SCONST
123	000000	EXT SEP10
124	000000	EXT SET240
125	000000	EXT SETTR1
126	000000	EXT SKPNM+
127	000000	EXT SKPNM1
128	000000	EXT SKF2NA
129	000000	EXT STOST
130	000000	EXT STOSV
131	000000	EXT SUBROI
132	000000	EXT SVPTRS
133	000000	EXT TRAFIC

```

=====
KARMA 03/14/81=====
ITEM   LOC   OBJECT CODE SRC=ORNMRT OBJ=GEZNM3  9/11/1981 12:08 PM PG  4
=====
>>>>  ENTRY POINTS AND EXTERNALS                                     <<<<<<<
1  000000      EXT TRVRBL
135 000000     EXT TSTEND
136 000000     EXT VAL#.
137 000000     EXT WARN

```

```

>>>>>>>  RUNTIME "FOR" ROUTINE  GEMINI 9-80  <<<<<<<<
  3 000000  *****
139 000000  *
140 000000  *  INTERNAL REPRESENTATION:  *
141 000000  *  *
142 000000  *  STEP <expr> TO <expr> STORE-Token <numval> <SNV> FOR  *
143 000000  *  *
144 000000  *  FOR-Stack entry format:  Incr. mem.  *
145 000000  *  *
146 000000  *    Address after STEP (PTR1)    3 BYTES    !  *
147 000000  *    Loop address (PCR of FOR)    3 BYTES    !  *
148 000000  *    STEP value                    8 BYTES    !  *
149 000000  *    TO value                      8 BYTES    v  *
150 000000  *    FOR variable address          3 BYTES    *
151 000000  *    -----  *
152 000000  *    31 BYTES (octal)  *
153 000000  *  *
154 000000  *  Register use:  *
155 000000  *    PTR1..... substitutes R10 as prog. pointer  *
156 000000  *    PTR2..... FOR-LOOP stack + 1 pointer (LAVAIL)  *
157 000000  *  *
158 000000  *    R10..... used instead of PTR2  *
159 000000  *    R14..... used instead of PTR2-  *
160 000000  *          only within the FOR-NEXT routines  *
160 000000  *  *
160 000000  *  *****
161 000000  *
162 000000  *  ***  FOR ATTRIBUTES TABLE  *****
163 000000 341  OCT 341
164 000001  *
165 000001 230  *
166 000002 123 270 311  FOR,  BIN
166 000005 377  LDBI R23,=PTR1-  load SNV token
167 000006 120 251 377  LDM R20,=FTADR  SNV runtime routine
167 000011 377  *
168 000012 316 377 377  JSB =FTADR  get address in R75
169 000015  *
169 000015  *  *****
170 000015  *  Space saving version uses  *
171 000015  *  register indirect addressing instead of  *
172 000015  *  literal  "  "  *
173 000015  *  *
174 000015  *  LDM R10,=PTR2  load address of PTR2  *
175 000015  *  LDM R14,=PTR2-  also the one of PTR2-  *
175 000015  *  *
175 000015  *  *****
176 000015  *
177 000015 316 231 000  JSB =GETFC  get FOR-loop cnt in R20
178 000020  ORP !20
179 000020 000 242  STB R20,R0  save it
180 000022  *
181 000022 155 261 025  LDMD R55,=LAVAIL  LAVAIL --> to FOR-stack+1
181 000025 200  *
182 000026 263 314 377  STMD R55,=PTR2  set pointer
  
```

```

KARMA 03/14/81=====
ITEM    LOC    OBJECT CODE  SRC=ORNMRT OBJ=GENNM3   9/11/1981 12:08 PM PG 6
=====
>>>>    RUNTIME "FOR" ROUTINE  GENINI 9-80          <<<<<<<<
1 000031 006 345          PUMD R55,+R6          save position
184 000033 100 212      FLOOP DCB R0          decrement count
185 000035 155          DRP 55
186 000036 364 072          JNG FOR1          jif no more
187 000040 271 317 377      LDMD R55,=PTR2-+    get next FOR-var addr.
188 000043 145 261 314      LDMD R45,=PTR2
189 000046 377
189 000047 315 031 000      SBM R45,=31,0,0    move to next entry + 1
189 000052 000
190 000053 263 314 377      STMD R45,=PTR2    update PTR2
191 000056 155 075 301      CMM R55,R75      abs. addresses match?
192 000061 266 350          JNZ FLOOP        jif no
193 000063 261 310 377      LDMD R55,=PTR1    a second pointer for ENOVDN
194 000066 035 243          STM R55,R35      save it
195 000070 006 343          POMD R55,-R6      get old LAVAIL
196 000072 145 313 031      ADM R45,=31,0,0
196 000075 000 000
197 000077 263 310 377      STMD R45,=PTR1    LWA source
197 000102 215          TCM R45
199 000103 055 303          ADM R45,R55      #bytes to move
200 000105 155 315 031      SBM R55,=31,0,0    trash one entry
200 000110 000 000
201 000112 263 025 200      STMD R55,=LAVAIL    new LAVAIL
202 000115 006 345          PUMD R55,+R6      push back for coming pop
203 000117 120 212          DCB R20          decrement count
204 000121 316 377 377      JSB =ENOVDN
204 000124 155 035 241      LDM R55,R35      get saved pointer
204 000127 263 310 377      STMD R55,=PTR1    restore PTR1
207 000132
208 000132          FOR1 DRP 155          save a DRP
209 000132 006 343          POMD R55,-R6      get actual LAVAIL
210 000134 313 071 000      ADM R55,=71,0,0    reserve space + 40
210 000137 000
211 000140 321 022 200      CMMD R55,=NXTMEM    out of room?
212 000143 373 054          JCY ERR1          jif yes
213 000145 315 040 000      SBM R55,=40,0,0    only 31 needed
213 000150 000
213 000151 120 210          ICB R20          incr. FOR-loop count
213 000153 373 050          JCY ERR2          jif too many
216 000155 021 242          STB R20,R21      save new count
217 000157 316 231 000      JSB =GETFC      position pointer
218 000162          * STBI R21,=PTR2      update count
219 000162
220 000162          *
220 000162          *
221 000162          * Fix for calculator FOR loops 7/16/81          *
221 000162          *
221 000162          *
222 000162 121          DRP 21
223 000163 316 315 002      JSB =UPFOR
224 000166
225 000166 155 263 025      STMD R55,=LAVAIL    update FOR-stack
225 000171 200

```

```

KARMA 03/14/81=====
ITEM    LOC    OBJECT CODE SRC=ORNMRT OBJ=GE2NM3  9/11/1981 12:08 PM PG 7
=====
>>>>>>>  RUNTIME "FOR" ROUTINE  GEMINI 9-80  <<<<<<<<
227 000172 263 314 377      STMD R55,=PTR2      and stack pointer
227 000175 175 273 315      STMI R75,=PTR2-      store FOR-vbl abs addr
227 000200 377
228 000201 150 223          CLM R50
229 000203 273 315 377      STNI R50,=PTR2-      leave room for T0 value
230 000206 154 251 377      LDM R54,=377,1,0,0 Tagged integer 1
230 000211 001 000 000
231 000214 150 273 315      STMI R50,=PTR2- STEP value
231 000217 377
232 000220 236              RTN
233 000221
234 000221 316 377 377 ERR1  JSB =ERROR+
235 000224 023                OCT 19D                out of memory
236 000225 316 377 377 ERR2  JSB =ERROR+
237 000230 021                OCT 17D                > 255 active FOR loops
238 000231 165 261 006 GETFC  LDMD R65,=FWCURRE      load 1st pgrm word + 1
238 000234 200
239 000235 315 026 000        SBM R65,=P,FCNG      subtract offset
240 000240 000                OCT 0
241 000241 263 314 377      STMD R65,=PTR2      load ext mem pointer
242 000244          *        LDBI R20,=PTR2-      get count
243 000244
244 000244 235              CLE                set CALC fors
245 000245 116 220          TSB R16                RUN?
246 000247 120              ORP 20
247 000250 367 002          JZR CLCFC                jif idle
248 000252 363 006          JEV PGMFC                jif not calc
249 000254 120 260 372 CLCFC  LDBD R20,=CALFOR
249 000257 206
250 000260 360 004          JMP GETFEX
251 000262          PGMFC      ORP 120
252 000262 270 315 377      LDBI R20,=PTR2-
253 000265 234              ICE
254 000266 236          GETFEX RTN
255 000267

```

>>>> RUNTIME TO/STEP Gemini 9-80 <<<<<<<<

7	000267						
258	000267			ERTOK	EQU 360		
259	000267			ENDTOK	EQU 16		
260	000267			HXTTK.	EQU 217		
261	000267			ATTOK	EQU 100		
262	000267			*			
263	000267			* ENTRY TO.			
264	000267			*			
265	000267			**** TO ATTRIBUTES TABLE ****			
266	000267	041		OCT 41			
267	000270			*****			
268	000270	316	206	001	TO, JSB =GTFOR	release memory, get FOR ptr	
269	000273	150	012	343	POMD R50,-R12	get real or integer	
270	000276	273	315	377	STMI R50,=PTR2-	store final value (TO)	
271	000301	146	270	313	LDBI R46,=PTR1-+	fetch next token	
271	000304	377					
272	000305	316	377	377	JSB =TSTEND	CR,0,!	
273	000310	371	017		JEZ CKDON	jif yes	
274	000312	236			RTN		
275	000313						
276	000313						
277	000313						
278	000313						
279	000313						
280	000313	041			**** STEP ATTRIBUTES TABLE ****		
281	000314				OCT 41		
282	000314	316	206	001	STEP, JSB =GTFOR	release memory, get FOR ptr	
283	000317	150	271	315	LDMI R50,=PTR2-	pop final value (TO)	
284	000322	377					
284	000323	140	012	343	POMD R40,-R12	pop increment from R12 stack	
285	000326	273	317	377	STMI R40,=PTR2-+	store incr, don't advance	
286	000331	140	271	315	CKDON LDMI R40,=PTR2-	R40=incr, step over it	
286	000334	377					
287	000335	145	261	115	LDMD R45,=PCR	get addr of program line	
287	000340	202					
288	000341	273	315	377	STMI R45,=PTR2-	stack it after (bef) STEP	
289	000344	261	310	377	LDMD R45,=PTR1	get program pointer	
290	000347	273	315	377	STMI R45,=PTR2-	stack it after (bef) PCR	
291	000352	175	263	314	STMD R75,=PTR2 ?	set address	
291	000355	377					
292	000356	143	271	314	LDMI R43,=PTR2	FNUM requires flags in R46	
292	000361	377					
293	000362	316	377	377	JSB =FNUM	numval to R60	
294	000365	231			BCD		
295	000366	140	060	241	LDM R40,R60	make copy of R60	
296	000371	316	371	001	JSB =CKDONE	go see if done	
297	000374	371	270		JEZ GETFEX	jif done	
298	000376	230			BIN		
299	000377	116	220		TSB R16	are we in CALC mode?	
300	000401	363	006		JEV SKPF-	jif no	
301	000403	222			CLB R16	set IDLE mode	
302	000404	316	377	377	JSB =SET240	set IMMEDIATE mode	
303	000407	360	171		JMP STPRT+	clean up	

```

KARMA 03/14/81=====
ITEM      LOC      OBJECT CODE  SRC=QRNMRT OBJ=GE2NM3  9/11/1981 12:08 PM  PG  9
=====
>>>>      RUNTIME TO/STEP      Gemini 9-80      <<<<<<<<
004 000411
305 000411 316 377 377 SKPF-      JSB =SETTR1      set trace 'jumped' flag
306 000414 316 341 002      JSB =LY-PT2      get start of FOR-stack
307 000417      DRP 165      and set PTR2
308 000417
308 000417      *****
309 000417      *
310 000417      * Following changes were made to share a search
311 000417      * token routine
312 000417      * Original code put back on 6/10/81 R.O.
313 000417      *
314 000417      * STMD R65,=PTR2      point to variable value
315 000417      * LDMI R65,=PTR2+      get address of name
316 000417      * ADMD R65,=FWCURR      complement it
317 000417      * STMD R65,=PTR2      set pointer to object
318 000417      * LDM R26,=217,21      left token, right token
319 000417      * JSB =SRCHTK      do the search
320 000417      * JEZ ENDPGM      jif not found
321 000417      * STMD R55,=PTR1      set to EOL of NEXT-stmt
322 000417      *
322 000417 052 243
323 000421 136 270 311 SKPLOP      STM R65,R52      FOR-var in R52 for comp
323 000424 377      LDBI R36,=PTR1-      get next token
324 000425 310 016      CMB R36,=16      EOL?
325 000427 366 042      JNZ TSTNXT      jif not EOL
326 000431 165 261 115      LDMD R65,=PCR      current line address
326 000434 202
327 000435 263 314 377      STMD R65,=PTR2
328 000440 144 271 315      LDMI R44,=PTR2-      R44=byte#, R45=line#
328 000443 377
329 000444 165 223      CLM R65
330 000446 044 240      LDB R65,R44      byte count
331 000450 323 310 377      ADMD R65,=PTR1      are we back to PTR2
332 000453 321 314 377      CMMD R65,=PTR2      at STSIZE?
333 000456 366 341      JNZ SKPLOP      jif not real EOL
334 000460 145 263 237      STMD R45,=ALINE#
335 000463 210
335 000464 316 377 377      JSB =NXTLIN      else next line
336 000467 371 126      JEZ ENDPGM      jif end
337 000471 360 326      JMP SKPLOP      else loop
338 000473      TSTNXT      DRP 136
339 000473 310 217      CMB R36,=NXTTK.      check if NEXT
340 000475 366 322      JNZ SKPLOP      jif no
341 000477 155 261 310      LDMD R55,=PTR1      save PC
341 000502 377
342 000503 006 345      PUMD R55,=R6)
343 000505 123 270 312 GSNVLP      LDBI R23,=PTR1+      get one byte
343 000510 377
344 000511 310 021      CMB R23,=21      SNV token?
345 000513 366 370      JNZ GSNVLP      jif no
346 000515
347 000515

```

```

KARMA 03/14/81=====
ITEM   LOC   OBJECT CODE SRC=ORNMRT OBJ=GEZHM3  9/11/1981 12:08 PM PG 10
=====
>>>      RUNTIME TO/STEP      Gemini 9-80      <<<<<<<
0 000515      *****
348 000515      *
349 000515      * Updated Aug. 31, 1981      *
349 000515      *
349 000515      *****
350 000515 175 223      CLM R75      null addr if exit
351 000517 155 261 310      LDMD R55,=PTR1      pointer
351 000522 377
352 000523 321 115 202      CMMD R55,=PCR      gone into previous line
353 000526 373 037      JCY RESTPC      jif yes
354 000530 263 314 377      STMD R55,=PTR2      copy pointer
355 000533 164 271 315      LDMD R64,=PTR2-      R64=len, if right tkn
355 000536 377
356 000537 316 377 377      JSB =SKP2NA      skip the name
357 000542 123 270 315      LDBI R23,=PTR2-      was it really a var?
357 000545 377
358 000546 310 217      CMB R23,=217      NEXT if yes
359 000550 366 333      JNZ GSNVLP      skip more if not
360 000552
361 000552 270 311 377      LDBI R23,=PTR1-      load token
362 000555 120 251 377      LDM R20,=FTADR      SNV runtime routine
362 000560 377
363 000561 316 377 377      JSB =FTADR      abs address to R75
364 000564 161 012 343      POND R61,-R12      undo what FTADR did
365 000567 155 006 343 RESTPC POND R55,-R6      get PC back
366 000572 263 310 377      STMD R55,=PTR1      restore it
367 000575 175 052 301      CMM R75,R52      see if right NEXT
368 000600 366 217      JNZ SKPLOP      jif no
369 000602
370 000602
371 000602 316 275 002 STPR+ JSB =EXNXT      reset LAVAIL & count
372 000605 236 STPRTH RTN      got to next line
373 000606
374 000606 316 377 377 GTFOR JSB =RELMEM      release temporary memory
375 000611 316 341 002 JSB =LV-PI2      start of FOR-stack to PTR2
376 000614      DRP 165
377 000614 075 243      STM R65,R75      get FOR-vbl's abs addr
378 000616 236      RTN
379 000617
380 000617 316 377 377 ENDPGM JSB =ERROR+
381 000622 056 OCT 46D      FOR w/o matching NEXT
382 000623
383 000623 230 OKLOOP BIN
384 000624 150 271 315      LDMD R50,=PTR2-      load final value (T0)
384 000627 377
385 000630 012 345      PUMD R50,+R12      save it
386 000632 271 315 377      LDMD R50,=PTR2-      load increment
387 000635 175 345      PUMD R75,+R12      save cv address
388 000637 263 314 377      STMD R75,=PTR2      set to fetch cv
389 000642 271 314 377      LDMD R75,=PTR2      get name ptr
390 000645 323 006 200      ADMD R75,=FWCURR      make it absolut
391 000650 345      PUMD R75,+R12      push it
392 000651 146 344      PUBD R46,+R12      and bits for store

```

```

KARMA 03/14/81=====
ITEM    LOC    OBJECT CODE  SRC=ORNMRT OBJ=GEZNM3  9/11/1981 12:08 PM  PG 11
=====
>>>>>>>  RUNTIME TO/STEP      Gemini 9-80      <<<<<<<<
000653 316 377 377      JSB =FNUM      current value to R60
394 000656 231      BCD
395 000657 160 040 243      STM R60,R40      copy in R40
396 000662      *****
396 000662      *
397 000662      *   Add increment to current value
397 000662      *
397 000662      *****
398 000662 316 141 002      JSB =MATH2      prepare for addition
399 000665 371 010      JEZ ADDR      jif both int
400 000667 316 377 377 ADDR      JSB =ADD9      add reals
401 000672 140 012 343      POMB R40,-R12      get new current value
402 000675 360 024      JMP ADDR1
403 000677 145 072 243 ADDR      STM R45,R72      save R45, don't kill R75
404 000702 055 303      ADM R45,R55      cv + incr (as integers)
405 000704 374 015      JLZ ADDR1      jif no overflow
406 000706 147 310 220      CMB R47,=90C
407 000711 374 010      JLZ ADDR1
408 000713 145 072 241      LDM R45,R72      restore for real addition
409 000716 316 155 002      JSB =R01      R40 & R50 to stack
410 000721 360 344      JMP ADDR8
411 000723 137 012 343 ADDR1      POMB R37,-R12      bits (trace,real,etc)
412 000726 036 243      STM R37,R36      save a copy
413 000730 317 062      ANM R37,=62      type & trace
414 000732 367 017      JZR ADDR1A      jif real & not trace
415 000734 136 012 344      PUBD R36,+R12      push back bits
416 000737 140 345      PUMB R40,+R12      push value
417 000741 006 345      PUMB R40,+R6      save value
418 000743 316 377 377      JSB =STOSV
419 000746 140 006 343      POMB R40,-R6      restore current value
420 000751 360 013      JMP ADDR1B
421 000753 175 012 343 ADDR1A      POMB R75,-R12      trash name ptr
422 000756 343      POMB R75,-R12      get cv address
423 000757 263 314 377      STMD R75,=PTR2      set pointer
424 000762 140 273 315      STMI R40,=PTR2-      update current value
424 000765 377
425 000766      *****
425 000766      *
426 000766      *   Calculate cv - limit
426 000766      *
426 000766      *****
427 000766 150 012 343 ADDR1B      POMB R50,-R12      pop limit
428 000771 316 141 002 CKDONE      JSB =MATH2      set up subtraction
429 000774 371 010      JEZ SUBR      jif both integer
430 000776 316 377 377 SUBRR1      JSB =SUBR01      subtract reals
431 001001 140 012 343      POMB R40,-R12      R40 = cv - limit
432 001004 360 032      JMP NEXT1
433 001006 155 215      SUBR      TCM R55      complement R55
434 001010 145 075 243      STM R45,R75      save R45
435 001013 055 303      ADM R45,R55      cv - limit (if integers)
436 001015 374 021      JLZ NEXT1      jif no overflow
437 001017 147 310 220      CMB R47,=90C
438 001022 374 014      JLZ NEXT1

```

```

KARMA 03/14/81=====
ITEM    LOC    OBJECT CODE  SRC=ORNMRT OBJ=GE2NM3  9/11/1981 12:08 PM  PG 12
=====
>>>>      RUNTIME TO/STEP      Gemini 9-80      <<<<<<<<
9 001024 145 075 241      LDM R45,R75      else restore R45
440 001027 316 155 002      JSB =R01      R40 & R50 to stack
441 001032 316 377 377      JSB =ADD9
442 001035 140 012 343      FOMD R40,-R12      R40 = cv - limit
443 001040      *****
443 001040      *
444 001040      * if (CV-LIM)*SIGN(INC)>=0 then
445 001040      * exit FOR loop
445 001040      *
445 001040      *****
446 001040
447 001040      +NEXT1      LDM R55,R25      get incr's addr
448 001040      *      STMD R55,=PTR2      set pointer
449 001040      *      LDMI R50,=PTR2-      get increment
450 001040 230      NEXT1      BIN
451 001041 155 261 025      LDMD R55,=LAVAIL      start of FOR-stack
451 001044 200
452 001045 315 013 000      SBM R55,=13,0,0      skip FOR-var + TO
452 001050 000
453 001051 263 314 377      STMD R55,=PTR2
454 001054 150 271 315      LDMI R50,=PTR2-      get increment
454 001057 377
455 001060 231      BCD
456 001061 141 204      LLB R41      sign upper if real
457 001063 100 250 042      LDB R0,=42
458 001066 144 310 377      CMB R44,=377
459 001071 372 013      JNC TSTR1      jif real
460 001073 100 250 045      LDB R0,=45
461 001076 141 222      CLB R41      sign pos
462 001100 147 220      TSB R47      test neg
463 001102 374 002      JLZ TSTR1      jif pos
464 001104 141 212      DCB R41      sign neg
465 001106 151 204      TSTR1      LLB R51      sign upper
466 001110 154 310 377      CMB R54,=377
467 001113 372 010      JNC TSTR2      jif real
468 001115 151 222      CLB R51      sign pos
469 001117 157 220      TSB R57      test neg
470 001121 374 002      JLZ TSTR2      JIF POS
471 001123 151 212      DCB R51      SET SIGN NEG
472 001125 235      TSTR2      CLE
473 001126 151 041 226      XRB R51,R41      SIGNS SAME ?
474 001131 364 005      JNG TSRTN      JIF NO
475 001133 101 221      TSM R*      R42 or R45
476 001135 367 001      JZR TSRTN      jif yes
477 001137 233      DCE
478 001140 236      TSRTN      RTN
479 001141
480 001141 235      NATH2      CLE      clear real flag
481 001142 144 310 377      CMB R44,=377      tagged integer?
482 001145 372 006      JNC R01      jif no
483 001147 154 310 377      CMB R54,=377
484 001152 372 001      JNC R01
484 001154 236      RTN

```

```

=====
KARMA 03/14/81=====
ITEM   LOC   OBJECT CODE   SRC=ORNMRT OBJ=GEZNM3   9/11/1981 12:08 PM PG 13
=====
>>>>  RUNTIME TO/STEP      Gemini 9-80      <<<<<<<<
.86 001155 235      ROI      CLE      clear real flag
487 001156 140 012 345      PUMD R40,+R12  push count variable
488 001161 150 345      PUMD R50,+R12  push limit
489 001163 233      DCE      flag real
490 001164 236      RTN

```

```

>>>>  RUNTIME "NEXT"      12-15-76      <<<<<<<<
2 001165
492 001165
493 001165
494 001165
495 001165
496 001165
497 001165
498 001165
499 001165
499 001165
499 001165
500 001165
501 001165
502 001165 341
503 001166
504 001166 230
505 001167
505 001167
507 001167
508 001167
509 001167
510 001167
510 001167
510 001167
511 001167
512 001167 316 231 000
513 001172 367 141
514 001174
515 001174 146 012 342
516 001177 175 343
517 001201 343
518 001202
519 001202 316 231 000 FNDVLP
520 001205 367 126
521 001207 165 261 025
521 001212 200
522 001213 263 314 377
523 001216 271 315 377
524 001221 075 301
525 001223 366 103
526 001225 316 223 001
527 001230 370 043
528 001232 316 377 377
529 001235 175 261 310
529 001240 377
530 001241 155 271 315
530 001244 377
531 001245 263 115 202
532 001250 271 315 377
533 001253 263 310 377
534 001256 116 220
535 001260 363 012

* Internal representation:
* NEXT-tok / varnam / length / addr (3) / NV-tok
* Ext memory pointer PTR1 scans it from right to left
* ENTRY NEXT.

*** NEXT ATTRIBUTES TABLE ***
OCT 341
NEXT. BIN
* R10 = addr of PTR2
* R14 = addr of PTR2- to save literal addressing
LDM R10,=PTR2
LDM R14,=PTR2-

JSB =GETFC
JZR NXERR

POBD R46,-R12 get bits
POMD R75,-R12 trash name pointer
POMD R75,-R12 get value pointer

JSB =GETFC get FOR count in R20
JZR NXERR jif no active FOR loops
LDMD R65,=LAVAIL set FOR-stack ptr

STMD R65,=PTR2
LDMI R65,=PTR2- fetch FOR-var
CMM R65,R75 matching FOR-NEXT
JNZ NXTST jif no
JSB =CKLOOP incr cv & cif done
JEN EXNXT jif done FOR loop
JSB =SETTR1 set trace 'jumped' flag
LDMD R75,=PTR1 save old PC

LDMI R55,=PTR2- get PCR

STMD R55,=PCR and update it
LDMI R55,=PTR2- get PC
STMD R55,=PTR1 update PTR1
TSB R16
JEV NEXTEX jif RUN
  
```

```

KARMA 03/14/81:=====
ITEM    LOC    OBJECT CODE SRC=ORNMRT OBJ=GEKMM3  9/11/1981 12:08 PM PG 15
=====
>>>>>  RUNTIME "NEXT"  12-15-76  <<<<<<<<
536 001262 155 321 022      CMMD R55,=NXTMEM  PTR1 => NXTMEM?
537 001265 200
538 001266 373 045      JCY  NXERR
539 001270 075 301      CMM  R55,R75
539 001272 372 041      JNC  NXERR
540 001274 236      NEXTEX RTN      jif bad direction
541 001275 230      EXNXT  BIN      go to top of loop
542 001276 155 261 025      LDMD R55,=LAVAIL  start of FOR-stack
542 001301 200
543 001302 315 031 000      SBM  R55,=31,0,0  trash FOR-stack entry
543 001305 000
544 001306 263 025 200      STMD R55,=LAVAIL  reset FOR-stack
545 001311 316 231 000      JSB  =GETFC  get FOR-loop count)
546 001314      DRP  !20
547 001314 212      DCB  R20
548 001315      *      STBI R20,=PTR2  update FOR count
549 001315      *      RTN      and exit
550 001315
551 001315 371 005      UPFOR  JEZ  UPCALF
552 001317 272 314 377      STBI R#,=PTR2
553 001322 360 003      JMP  UPEX
554 001324 262 372 206  UPCALF STBD R#,=CALFOR
555 001327 236      UPEX  RTN
556 001330
557 001330
558 001330 316 275 002 HXTST JSB  =EXNXT  advance to next FOR
559 001333 366 245      JNZ  FNDVLP  jif if some left
560 001335
561 001335 316 377 377 NXERR JSB  =ERROR+
562 001340 057      OCT  47D      NEXT w/o matching FOR
563 001341
564 001341 165 261 025 LV-PT2 LDMD R65,=LAVAIL  get start of FOR-stack
564 001344 200
565 001345 263 314 377      STMD R65,=PTR2  set pointer
566 001350 271 315 377      LDMI R65,=PTR2-  fetch FOR-var
567 001353 236      RTN
568 001354

```

TOKEN SEARCH ROUTINE

<<<<<<<

```

0 001354
570 001354
571 001354
572 001354
573 001354
574 001354
575 001354
576 001354
577 001354
578 001354
579 001354
580 001354
581 001354
581 001354
581 001354
582 001354
583 001354 165 006 345 SRCHTK PUMD R65,+R6 save start of name
584 001357 263 310 377 STMD R65,=PTR1 set pointer
585 001362 261 314 377 LOMD R65,=PTR2 save PTR2
586 001365 345 PUMD R65,+R6
587 001366 316 377 377 JSB =MATCH? find a matching name
588 001371 165 006 343 POND R65,-R6
589 001374 263 314 377 STMD R65,=PTR2
590 001377 343 POND R65,-R6 back up to name strt
591 001400 370 010 JEH SRCHMO compare tokens
592 001402 165 213 HXT DCM R65
593 001404 321 014 200 CMMD R65,=BOVAR end of program?
594 001407 373 343 JCY SRCHTK jif no
595 001411 236 RTN
596 001412
597 001412 175 270 311 SRCHMO LOBI R75,=PTR1- token after name
597 001415 377
598 001416 026 300 CMB R75,R26 match?
599 001420 366 360 JNZ NXT jif no
600 001422 155 261 310 LOMD R55,=PTR1 save EOL position
600 001425 377
601 001426 165 263 310 STMD R65,=PTR1 set to start of name
601 001431 377
602 001432 175 271 312 LDMI R75,=PTR1+ skip address
602 001435 377
603 001436 270 310 377 LOBI R75,=PTR1 token before name
604 001441 027 300 CMB R75,R27 match?
605 001443 366 335 JNZ NXT jif no
606 001445 236 RTN
607 001446
608 001446

```

67553

```

>>>>>>> ON KEY# OFF KEY# Gemini 9-80 <<<<<<<<
10 001446 *****
610 001446 *
611 001446 * Internal representation:
612 001446 *
613 001446 * line# (3) / GOTO-Token
614 001446 * lblname / length / addr (3) / GOTO-token
615 001446 *
616 001446 * Register use:
617 001446 * R24 = LEGEND or LEGCAL
618 001446 * R33 = Interrupt/Definition mode
619 001446 * R46 = Key number (also R76)
620 001446 * R32,R34,R36,R53,R63 Scratch
620 001446 *
620 001446 *****
621 001446
622 001446 LENKLB EQU 36 max length of cmd$
623 001446
624 001446 *** ON KEY# ATTRIBUTES TABLE *****
625 001446 241 OCT 241
626 001447 *****
627 001447 230 ONKEY. BIN check if string present
628 001450 106 263 066 STMD R6,=SAVER6 to restore, if errors
628 001453 210
629 001454 133 222 CLB R33 preset interrpt skays
630 001456 134 012 241 LDM R34,R12 get Top Of Stack
631 001461 315 010 000 SBM R34,=10,0 - key#
632 001464 325 344 203 SBMD R34,=TOS - base
633 001467 310 012 CMB R34,=12 two strings ?
634 001471 366 115 JNZ RUKEYS jif no, no skay def
635 001473
636 001473 *****
636 001473 *
637 001473 * Softkey definition with ON KEY# statement
638 001473 *
639 001473 * Syntax:
640 001473 * ON KEY# <expr>,<label string>,<command string>
640 001473 *
641 001473 *****
642 001473 133 212 DCB R33 set skays definition
643 001475 163 343 POMD R63,-R12 add + len of cmd$
644 001477 153 343 POMD R53,-R12 add + len of lbl$
645 001501 006 345 PUMD R53,+R6 save them
646 001503 163 345 PUMD R63,+R6
647 001505 134 345 PUMD R34,+R6 save R34 too
648 001507 316 330 003 JSB =KEY2 number to R46 & R76
649 001512 134 006 343 POMD R34,-R6 restore R34
650 001515 176 002 243 STM R76,R2 save key#
651 001520 213 DCM R76 for indexing
652 001521 166 223 CLM R66
653 001523 250 036 LDB R66,=LENKLB entry size
654 001525 316 377 377 JSB =INTMUL * key#
655 001530 173 006 343 POMD R73,-R6 get cmd$ back

```

```

KARMA 03/14/81=====
ITEM LOC OBJECT CODE SRC=ORNMRT OBJ=GEZNM3 9/11/1981 12:08 PM PG 18
=====
>>>> ON KEY# OFF KEY# Gemini 9-80 <<<<<<<
3 001533 136 073 241 LDM R36,R73 put length in 2 bytes
657 001536 311 035 000 CNM R36,=35,0 > 28 chars?
658 001541 373 035 JCY KEYERR jif yes
659 001543 130 054 241 LDM R30,R54 offset
660 001546 313 210 215 ADM R30,=SKYTXT add base
661 001551 136 030 345 PUMD R36,+R30 first 2 are cmd$ len
662 001554 316 377 377 JSB =DCSTR1 copy at R30
663 001557 155 006 343 POMD R55,-R6 lbl$ addr
664 001562 263 314 377 STMD R55,=PTR2 set pointer
665 001565 134 343 POMD R34,-R6 lbl$ length
666 001567 176 002 241 LDM R76,R2 restore orig key#
667 001572 124 251 005 LDM R24,=LEGCAL update legend
667 001575 203
668 001576 360 053 JMP KCLR-
669 001600
670 001600 106 261 066 KEYERR LDMD R6,=SAVER6 clean up
670 001603 210
671 001604 316 377 377 JSB =ERROR+
671 001607 131 OCT 89D
671 001610
674 001610
674 001610
675 001610
676 001610
677 001610
678 001610
678 001610
678 001610
679 001610
680 001610 134 221 RUKEYS TSM R34
681 001612 367 010 JZR ONK1
682 001614 155 012 343 POMD R55,-R12 string addr
683 001617 263 314 377 STMD R55,=PTR2 save it
684 001622 134 343 POMD R34,-R12 string length
685 001624 ONK1 DRP !34
686 001624 006 345 PUMD R34,+R6
687 001626 316 330 003 JSB =KEY2 get number of key
687 001631 134 006 343 POMD R34,-R6
687 001634 165 261 310 LDMD R65,=PTR1 get this address
689 001637 377
690 001640 325 006 200 SBMD R65,=FWCURR make it relative
691 001643 046 267 016 STMD R65,X46,KEYTAB set ON KEY address
691 001646 204
692 001647 124 251 145 LDM R24,=LEGEND
692 001652 202
693 001653 316 034 004 KCLR- JSB =KCLR clear old label
694 001656 134 311 013 CNM R34,=13,0 keylabel len <= 11?
694 001661 000
695 001662 372 003 JNC SZOK jif yes
696 001664 251 012 000 LDM R34,=12,0 else make it 10
697 001667 SZOK DRP !34
698 001667 221 TSM R34
698 001670 367 014 JZR .LEGDU- jif no legend

```

```

KARMA 03/14/81=====
ITEM   LOC   OBJECT CODE SRC=ORNMRT OBJ=GEWNN3  9/11/1981 12:08 PM PG 19
=====
>>>>      ON KEY# OFF KEY#      Gemini 9-80      <<<<<<<<
700 001672 132 270 315 KSETLP  LDBI R32,=PTR2-  get one char from str
700 001675 377
701 001676 312 200      ADB R32,=200      invert field
702 001700 076 344      PUBD R32,=R76      next char to legend
703 001702 134 213      DCM R34      decr count
704 001704 366 364      JNZ KSETLP      loop if more
705 001706 133 220      LEGDU-  TSB R33      skeys definition?
706 001710 364 015      JNG SKPLB      jif yes
707 001712 230      LEGDUN  BIN      for add
708 001713 174 271 311      LDMI R74,=PTR1-  GOTO-tok, 3 bytes addr
708 001716 377
709 001717 177 310 262      CMB R77,=262      GOTO/GOSUB label token?
710 001722 372 003      JNC SKPLB      jif no
711 001724 316 377 377      JSB =SKFNM+      skip label
712 001727 236      SKPLB  RTN
713 001730
714 001730 316 377 377 KEY2  JSB =ONEB      get key# in R46 & R76
715 001733 146 221      TSM R46
716 001735 367 241      JZR KEYERR      jif bad number
717 001737 310 017      CMB R46,=15D      =>15D ?
718 001741 373 235      JCY KEYERR      jif yes
719 001743 146 213      KTRIPL DCM R46      key-1
720 001745 002 243      STM R46,R2      save a copy
721 001747 205      LLM R46      (key-1)*2
722 001750 303      ADM R46,R2      triple for indexing
723 001751 236      RTN
724 001752
725 001752      **** OFF KEY# ATTRIBUTES TABLE *****
726 001752 241      OCT 241
727 001753      *****
728 001753 230      OFKEY.  BIN
729 001754 124 251 145      LDM R24,=LEGEND  prgm labels area
729 001757 202
730 001760 134 012 241      LDM R34,R12      check stack
731 001763 325 344 203      SBMD R34,=TOS
732 001766 367 007      JZR OFALL      jif no param
733 001770 310 010 000      CMB R34,=10,0
734 001773 367 026      JZR OFONE      jif if one number
735 001775 360 201      JMP KEYERR      garbage otherwise
736 001777
737 001777 136 251 016 OFALL  LDM R36,=140,0  key#
737 002002 000
738 002003 136 076 243 OFALL+  STM R36,R76      make a copy
739 002006 046 243      STM R36,R46
740 002010 316 343 003      JSB =KTRIPL      triple for indexing
741 002013 316 026 004      JSB =OFONE+
742 002016 136 213      DCM R36
743 002020 366 361      JNZ OFALL+      jif while keys left
744 002022 236      RTN
745 002023
746 002023 316 330 003 OFONE  JSB =KEY2
747 002026 165 223      OFONE+  CLM R65
748 002030 046 267 016      STND R65,X46,KEYTAB clear ON KEY addr

```

>>>>> ON KEY# OFF KEY# Gemini 9-80 <<<<<<<<

3	002033	204					
749	002034	145	251	005	KCLR	LDM	R45,=5,12,240 empty\fieldlen\whites
749	002037	012	240				
750	002041	166	223			CLM	R66
751	002043	046	240			LDB	R66,R46
752	002045	176	312	006		ADB	R76,=6
753	002050	310	016			CMB	R76,=16
754	002052	372	006			JNC	CALCAD
755	002054	145	314	003		SBB	R45,=3
756	002057	176	314	016		SBB	R76,=16
757	002062				CALCAD	DRP	176
758	002062	056	242			STB	R76,R56
759	002064				MLTLOP	ARP	156
760	002064	176	302			ADB	R76,R56
761	002066	166	212			DCB	R66
762	002070	366	372			JNZ	MLTLOP
763	002072	176	045	302		ADB	R76,R45
764	002075	024	303			ADM	R76,R24
765	002077	056	243			STM	R76,R56
766	002101				KCLRRLP	ARP	156
767	002101	147	344			PUBD	R47,+R56
768	002103	146	212			DCB	R46
769	002105	366	372			JNZ	KCLRRLP
770	002107	236				RTN	
771	002110						
772	002110						

***** ON ERROR ATTRIBUTES TABLE *****

OCT 241

774	002111						
775	002111	230			ONERR.	BIN	
776	002112	155	261	310		LDMD	R55,=PTR1
776	002115	377					get PC
777	002116	325	006	200		SBMD	R55,=FWCURR
778	002121	263	111	200		STMD	R55,=ERGOTO
779	002124	104	251	311		GTO	LEGDUN
779	002127	003					skip GOTO
780	002130						

*****OFF ERROR ATTRIBUTE TABLE*****

OCT 241

783	002131						
784	002131	165	223		OFFER.	CLM	R65
785	002133	263	111	200		STMD	R65,=ERGOTO
786	002136	236				RTN	

store zero
in ON ERROR location
done

>>>> ON GOTO/GOSUB. Gemini 9-80 <<<<<<<<

```

38 002137
789 002137
789 002137
790 002137
791 002137
792 002137
793 002137
794 002137
795 002137
796 002137
797 002137
798 002137
799 002137
800 002137
801 002137
801 002137
801 002137
802 002137
803 002137
804 002137
805 002137
806 002137
807 002137
808 002137 230
809 002140
810 002140 316 377 377 ON,
811 002143 145 213
812 002145 364 154
813 002147 211
814 002150 136 251 132
814 002153 262
815 002154
816 002154 135 270 313
816 002157 377
817 002160 036 300
818 002162 367 021
819 002164 037 300
820 002166 367 015
821 002170 136 251 133
821 002173 270
822 002174
823 002174 135 036 300
824 002177 367 004
825 002201 037 300
826 002203 366 116
827 002205 155 261 310 ONLPI
827 002210 377
828 002211 165 270 311
828 002214 377
829 002215 036 300
830 002217 366 005
831 002221 271 311 377
832 002224 360 013

```

```

* Syntax:
*
* EOL /.../ lblname / len / GLBL-tok/...
* .../ line# / GLN#-tok / ON-tok / <expr>
*
* Register use:
* R36 = GOTOK / GOSTOK
* R37 = GOLBL / GOSLBL
* R35 = first token after expression
* R45 = ordinal
* R55-7= to update PTR1 & PTR2
* R63-7= element of GOTO / GOSUB-list
*
*****

```

```

GOTOK EQU 132
GOSTOK EQU 133
GOLBL EQU 262
GOSLBL EQU 270
***** ON ATTRIBUTE TABLE *****
OCT 230
*****

```

JSB =ONEI	GET #
DCM R45	decrement it
JNG INVAL	jif < 0
ICM R45	restore count
LDM R36,=132,262	preset GOTO line#
* and GOTO label	
LDBI R35,=PTR1-+	get jump token
CMB R35,R36	GOTO line# token?
JZR ONLPI	jif yes
CMB R35,R37	GOTO label token?
JZR ONLPI	jif yes
LDM R36,=133,270	load GOSUB line#
* and GOSUB label	
CMB R35,R36	GOSUB line# token?
JZR ONLPI	jif yes
CMB R35,R37	GOSUB label token?
JNZ INVAL	jif no
LDMD R55,=PTR1	save PTR1 for exit
LDBI R65,=PTR1-	get G...token
CMB R65,R36	GOTOK / GOSTOK?
JNZ GLBLI?	jif no
LDHI R65,=PTR1-	trash line#
JMP DCR45	

```

=====
>>>>      ON GOTO/GOSUB      Gemini 9-80      <<<<<<<<
3 002226      GLBL1?  ORP  165
834 002226 037 300      CMB  R65,R37      GOSBL / GOSLBL?
835 002230 266 071      JNZ  INVAL      invalid token
836 002232 271 311 377      LDHI R65,=PTR1- skip the address
837 002235 316 377 377      JSB  =SKPNM1 set bin & skip the label
838 002240 231      BCD      restore mode
839 002241 145 213      DCR45  DCM  R45
840 002243 266 340      JNZ  ONLP1      jif not the one
841 002245 165 261 310      LDMD R65,=PTR1
841 002250 377
842 002251 263 314 377      STMD R65,=PTR2      set to next in list
843 002254 155 263 310      STMD R55,=PTR1      set to right one
843 002257 377
844 002260 165 270 315 ONLP2  LDBI R65,=PTR2-      get next list token
844 002263 377
845 002264 036 300      CMB  R65,R36      G-- line#
846 002266 266 005      JNZ  GLBL2?      jif no
847 002270 271 315 377      LDHI R65,=PTR2-      trash line#
847 002273 360 363      JMP  ONLP2
849 002275      GLBL2?  ORP  165
850 002275 037 300      CMB  R65,R37      jump to label token?
851 002277 266 010      JNZ  SETONF      jif no
852 002301 271 315 377      LDHI R65,=PTR2-      skip the address
853 002304 316 377 377      JSB  =SKP2NA      skip the label
854 002307 260 347      JMP  ONLP2
855 002311      SETONF  ORP  165
855 002311 272 316 377      STBI R65,=PTR2+      restore byte (EOL)
857 002314 261 314 377      LDMD R65,=PTR2      addr after last good jmp
858 002317 263 065 200      STMD R65,=ONFLAG      store return addr
859 002322 236      RTN
860 002323
861 002323 316 377 377 INVAL  JSB  =ERROR+
862 002326 013      OCT  110      jump out of range

```

UNFORMATTED PRINT 02-17-77

((((((((

```

864 002327          ZONELN  EQU 210
865 002327
866 002327          *****
866 002327          *
867 002327          * ENTRY PRINT.
868 002327          *
869 002327          * ENTRY DISP.
869 002327          *
869 002327          *****
870 002327
871 002327          ***** DISP ATTRIBUTES TABLE *****
872 002327 241          OCT 241
873 002330          *****
874 002330 140 261 212 DISP. LDMD 40,=CS.C.      set CRT select code
874 002333 200
875 002334 150 251 142 DISP.+ LDM 50,=DISBUF      set pointers
875 002337 201
876 002340 060 200          DEF DISPTR            display pointer to R52
877 002342 121 202          DEF DSFLAG            " flag to R54
878 002344 136 202          DEF DISPLN            " length to R56
879 002346 360 017          JMP DOCOM            set the flag
880 002350
881 002350
882 002350          ***** PRINT ATTRIBUTES TABLE *****
883 002350 241          OCT 241
884 002351          *****
885 002351 140 261 222 PRINT. LDMD 40,=PS.C.      set printer S.C.
885 002354 200
886 002355 150 251 054 PRIN.+ LDM 50,=PRTBUF      set pointers
886 002360 217
887 002361 062 200          DEF PRTPTR            print pointer to R52
888 002363 120 202          DEF PRFLAG            " flag to R54
889 002365 137 202          DEF PRNTLN            " length to R56
890 002367 263 306 203 DOCOM STMD R#, =P.BUFF      do common stuff
891 002372 222          CLB R#
892 002373 262 064 200      STBD R#, =USING?      clear USING!!
893 002376 210          ICB R#
894 002377 272 312 203      STBI R#, =P.FLAG      set flag
895 002402 140 263 321      STMD R40, =SCTEMP      set SC for traffic cop
895 002405 203
896 002406 236          RTN

```

```

=====
>>>> PRINT LINE <<<<<<<
3 002407 *****
898 002407 *
899 002407 * PRINT LINE ENTRY *
900 002407 *
901 002407 * INPUT: R36 = character count *
902 002407 * R54 = " " *
903 002407 * PTR2 = points to line to be output *
904 002407 * OUTPUT: R30 = print buffer *
905 002407 *
906 002407 * Note: expects the text in line backwards and *
907 002407 * assumes there was a call to RESFIL *
907 002407 *
908 002407 *****
909 002407 136 311 001 PRNFMT CMM R36,=1,0 character count?
909 002412 000
910 002413 366 012 JNZ NTPRL jif not eol
911 002415 134 270 317 LDBI R34,=PTR2-+ get from buffer
911 002420 377
912 002421 310 015 CMB R34,=CR carriage return?
913 002423 366 002 JNZ NTPRL jif not eol
914 002425 360 167 JMP PEOL jif end line
915 002427 122 271 310 NTPRL LDMI R22,=P.PTR
915 002432 203
916 002433 367 030 JZR NOTOVR jif buffer empty
917 002435 136 022 303 ADM R36,R22 char count + p.ptr
917 002440 134 223 CLM R34
919 002442 270 314 203 LDBI R34,=LINELN device line length
920 002445 036 301 CMM R34,R36 LINELN-(PRTPTR+BYTES)
921 002447 365 014 JPS NOTOVR jif it will fit
922 002451 177 260 064 LDBD R77,=USING? using flag set?
922 002454 200
923 002455 366 006 JNZ NOTOVR jif yes
924 002457 316 367 005 JSB =WRTLIN go write it
925 002462 136 054 241 LDM R36,R54 no. of bytes
926 002465 130 271 310 NOTOVR LDMI R30,=P.PTR
926 002470 203
928 002471 323 306 203 ADMD R30,=P.BUFF current location
928 002474 122 270 310 LDBI R22,=P.PTR current pointer
928 002477 203
929 002500 154 221 TSM R54
930 002502 367 036 JZR MOVEDN jif none to move
931 002504 136 270 315 MYLUP LDBI R36,=PTR2- get from buffer
931 002507 377
932 002510 030 344 PUBD R36,+R30 to print buffer
933 002512 122 210 ICB R22 inc pointer
934 002514 316 242 005 JSB =TSTEOL end of line?
935 002517 154 213 MYCKD DCM R54 decrement count
936 002521 366 361 JNZ MYLUP jif some left
937 002523 316 242 005 JSB =TSTEOL cif at eol
938 002526 DRP 122
939 002526 220 TSB R22 at begin of line?
1 002527 367 060 JZR PRFEK jif yes

```

```

KARMA 03/14/81=====
ITEM LOC OBJECT CODE SRC=ORNMRT OBJ=GEZNM3 9/11/1981 12:08 PM PG 25
=====
>>>>> PRINT LINE <<<<<<<<
941 002531 136 022 240 LDB R36,R22
942 002534 ZNTST DRP 136
943 002534 314 025 SBB R36,=ZONELN
944 002536 367 051 JZR PRFEX jif begin zone
945 002540 365 372 JPS ZNTST loop if not current zone
946 002542 120 220 MOVEDN TSB R20 semi?
947 002544 366 043 JNZ PRFEX jif yes
948 002546 * ELSE STEP TO NEXT ZONE
949 002546 136 022 240 LDB R36,R22
950 002551 NXLOOP DRP 136
951 002551 314 025 SBB R36,=ZONELN
952 002553 365 374 JPS NXLOOP loop if not this zone
953 002555 214 TCB R36 make it positiv
954 002556 176 250 040 LDB R76,=BLANK
955 002561 123 222 CLB R23 for multi
956 002563 102 261 306 NXLOP1 LDMD R2,=P.BUFF buffer pointer
956 002566 203
957 002567 022 303 ADM R2,R22 current pointer
958 002571 176 002 246 STBD R76,R2 move blank
959 002574 316 242 005 JSB =TSTEOL see if eol
960 002577 367 055 JZR NXZRTN jif yes
961 002601 DRP 122
962 002601 210 ICB R22 inc current pointer
963 002602 136 212 DCB R36 decrement count
964 002604 366 355 JNZ NXLOP1 loop if not done
965 002606 316 242 005 JSB =TSTEOL
966 002611 122 272 310 PRFEX STBI R22,=P.PTR store new value
966 002614 203
967 002615 236 RTN
968 002616
969 002616 122 270 310 PEOL LDBI R22,=P.PTR get print pointer
969 002621 203
970 002622 366 027 JNZ BUFF jif buff not empty
971 002624 160 250 040 LDB R60,=BLANK
972 002627 272 306 203 STBI R60,=P.BUFF
973 002632 122 270 312 LDBI R22,=P.FLAG
973 002635 203
974 002636 367 117 JZR CLEAN jif already printed
975 002640 360 011 JMP BUFF go write the line
976 002642
977 002642 102 270 314 TSTEOL LDBI R2,=LINELN get end of line
977 002645 203
978 002646 122 002 300 CMB R22,R2 are we there?
979 002651 372 003 JNC NXZRTN jif not
980 002653 316 367 005 BUFF JSB =WRTLIN
981 002656 236 NXZRTN RTN
982 002657
983 002657 **** PRINT EOL ATTRIBUTES TABLE *****
984 002657 035 OCT 35
985 002660 *****
986 002660 230 PRLINE BIN
987 002661 * LDBD 54,=SCT+7 INTERCEPT FOR TAPE?
988 002661 * CMB 54,=377

```

```

>>>>> PRINT LINE
9 002661 * JNZ PRLIN1
990 002661 * TSB R16
991 002661 * JEV BUFFEX
992 002661 * JSB =BUFDMP
993 002661 +UFFEX RTN
994 002661 122 260 064 PRLIN1 LDMD R22,=USING?
994 002664 200
995 002665 367 047
996 002667 130 261 366
996 002672 203
997 002673 175 261 310
997 002676 377
998 002677 006 345
999 002701 261 370 203
1000 002704 072 243
1001 002706 263 310 377
1002 002711 261 363 203
1003 002714 301
1004 002715 366 006
1005 002717 122 270 312
1005 002722 203
1006 002723 367 003
1007 002725 316 172 011 PRLIN3
1008 002730 155 006 343 PRLIN2-
1009 002733 263 310 377
1010 002736 154 223 PRLIN2
1011 002740 211
1012 002741 316 377 377
1013 002744 370 011
1014 002746 136 250 015
1015 002751 272 317 377
1016 002754 316 377 377
1017 002757 316 377 377 CLEAN
1018 002762 112 261 344
1018 002765 203
1019 002766 236
1020 002767
1021 002767 154 006 345 WRTLIN
1022 002772 136 223
1023 002774 022 240
1024 002776 126 261 306
1024 003001 203
1025 003002 316 377 377
1026 003005 130 261 306
1026 003010 203
1027 003011 154 006 343
1028 003014 122 223
1029 003016 272 312 203
1030 003021 273 310 203
1031 003024 236
1032 003025 *

```

<<<<<<<<

CALC mode?
 exit if no
 dump buffers
 get flag
 jif not using
 get curr image len
 get image pointer
 save it
 current image addr
 make a copy
 set image pointer
 start image addr
 CIA = SIA?
 jif no
 first print token?
 jif no
 restore pointer
 byte count for RESFIL
 res memory & set ptr2
 jif no room
 load eol & push it
 don't move pointer
 release memory
 reset R12
 save character count
 send the message
 and power up CRT
 restore count
 clear flag
 reset printer pointer

```

KARMA 03/14/81=====
ITEM   LOC   OBJECT CODE SRC=ORNMRT OBJ=GEWMM3  9/11/1981 12:08 PM PG 27
=====
>>>> USING <<<<<<<

1034 003025      *
1035 003025      INDTOK EQU 367
1036 003025      LBLTOK EQU 257
1037 003025      IMTOK EQU 216
1038 003025      USINGL EQU 345      USING <label ref>-token
1039 003025      **** USING ATTRIBUTES TABLE ****
1040 003025 041      OCT 41
1041 003026      ****
1042 003026 230      USING. BIN
1043 003027 173 012 343      POMB R73,-R12      line# or string addr
1044 003032 173 263 361 USEXIT STMD R73,=IMLEN      set start length
1044 003035 203
1045 003036 263 366 203      STMD R73,=IMLEN      set current length
1046 003041 177 250 177      LDB R77,=177
1047 003044 262 064 200      STBD R77,=USING?      set USING flag
1048 003047 236      RTN
1049 003050      ****
1049 003050      *
1050 003050      * USING LABEL ATTRIBUTES
1050 003050      *
1050 003050      ****
1051 003050 027      OCT 27
1052 003051 230      USINL. BIN
1053 003052 123 250 345      LDB R23,=USINGL      USING LABEL token
1054 003055 316 071 006      JSB =USCOM      get & set image addr
1055 003060 316 377 377      JSB =SKPNM+      skip label name
1056 003063 236      RTN
1057 003064      )
1058 003064      ****
1058 003064      *
1059 003064      * USING LINE # ATTRIBUTES
1059 003064      *
1059 003064      ****
1060 003064 027      OCT 27
1061 003065
1062 003065 230      ULIN#. BIN
1063 003066 123 250 347      LDB R23,=347      USING <line#>-token
1064 003071 316 377 377 USCOM JSB =FETADD      get address of image
1065 003074 165 323 006      ADMD R65,=FWCURR      and make it absolut
1065 003077 200
1066 003100 263 314 377      STMD R65,=PTR2      set 2nd pointer
1067 003103 271 315 377      LDHI R65,=PTR2-      trash line number
1068 003106 250 216      LDB R65,=IMTOK      token to look for
1069 003110 316 133 006      JSB =FINDTK
1070 003113 366 162      JNZ FMTER1
1071 003115 173 223      CLM R73
1072 003117 270 315 377      LDBI R73,=PTR2-      trash unq-string token
1073 003122 270 315 377      LDBI R73,=PTR2-      string length (one byte)
1074 003125 175 261 314      LDMD R75,=PTR2      string addr
1074 003130 377
1075 003131 360 277      JMP USEXIT
1076 003133
1077 003133 136 271 315 FINDTK LDHI R36,=PTR2-      get 2 bytes after line#

```

```

KARMA 03/14/81=====
ITEM    LOC    OBJECT CODE SRC=ORMMRT OBJ=GENHM3   9/11/1981 12:08 PM PG 28
=====
X>>>>      USING                                     <<<<<<<<
1  7 003136 377
1078 003137 310 367          CMB R36,=INDTOK      indentation?
1079 003141 366 003          JNZ LBLTK?           jif no
1080 003143 271 315 377      LDMI R36,=PTR2-      indent blanks & nxt tok
1081 003146          LBLTK?  DRP 136
1082 003146 310 257          CMB R36,=LBLTOK      label?
1083 003150 366 007          JNZ TOK65?          jif no
1084 003152 316 377 377      JSB =SKP2NA         skip label
1085 003155 136 270 315      LDBI R36,=PTR2-      get next one
1085 003160 377
1086 003161          TOK65?  DRP 136
1087 003161 065 300          CMB R36,R65          compare w. input token
1088 003163 236          RTN
1089 003164          * PRINT STRING
1090 003164          *
1091 003164          **** COMMA$ ATTRIBUTES TABLE ****
1092 003164 036          OCT 36
1093 003165          ****
1094 003170 200          COMMA$ LDBD R20,=USING?    SEMI IF USING/ELSE COMMA
1095 003171 360 004          JMP STR
1096 003173
1097 003173          **** SEMIC$ ATTRIBUTES TABLE ****
1098 003173 036          OCT 36
1099 003174          ****
1100 003174 120 222          SEMIC$ CLB R20
1101 003176 210          ICB
1102 003177 143 012 343 STR  POMD R43,-R12        set string flag
1103 003202 316 371 006      JSB =CALSCN        R43=$len, R45=$addr
1104 003205 373 062          JCY P$RTN          SAVE REG AND SCAN
1105 003207 230          BIN
1106 003210          DRP 154
1107 003210 221          TSM R54
1108 003211 366 012          JNZ DUFMT           jif no
1109 003213 043 241          LDM R54,R43        length of string
1110 003215 156 223          CLM R56           only two bytes
1111 003217 154 221          TSM R54
1112 003221 367 043          JZR COMRTN         jif null string
1113 003223 173 212          DCB R73           string format
1114 003225 173 220          DUFMT  TSB R73
1115 003227 365 046          JPS F$NTER1        jif not string format
1116 003231 316 377 377      JSB =RESFIL        reserve space
1117 003234 370 033          JEN P$RTN         jif no room
1118 003236 316 377 377      JSB =SVPTRS       save PTR1 & PTR2
1119 003241          DRP 145
1120 003241 263 310 377      STMD R45,=PTR1      source = $ addr
1121 003244 223          CLM R45               to fill w/count
1122 003245 122 043 241      LDM R22,R43        $ len
1123 003250 054 301          CNM R22,R54        1t char count?
1124 003252 372 001          JNC DUFMT1        jif data < image
1125 003254 241          LDM R22,R54          else truncate
1126 003255 122 045 243 DUFMT1 STM R22,R45      # bytes to move
1  7 003260 316 377 377      JSB =EMOVUP       copy string into buffer

```

```

=====
>>> USING <<<<<<<
1128 003263 316 377 377 JSB =RSPTRS restore PTR1 & PTR2
1129 003266 316 377 377 COMRTN JSB =TRAFIC call traffic cop
1130 003271 122 222 P$RTH CLB R22
1131 003273 272 312 203 STBI R22,=P.FLAG set flag
1132 003276 236 RTN
1133 003277
1134 003277 316 377 377 FMTER1 JSB =ERROR+
1135 003302 065 OCT 53D
1136 003303
1137 003303
1137 003303
1138 003303 * Print Numeric
1138 003303
1138 003303
1139 003303
1140 003303
1141 003303 036 ***** COMMA. ATTRIBUTES TABLE *****
1142 003304 OCT 36
1143 003304 120 260 064 ***** COMMA. LDBD R20,=USING? semi if USING else comma *****
1143 003307 200
1144 003310 360 004 JMP TSTUS
1145 003312
1146 003312 ***** SEMIC. ATTRIBUTES TABLE *****
1147 003312 036 OCT 36
1148 003313 *****
1149 003313 120 222 SEMIC. CLB R20
1150 003315 210 ICB R20 set string flag
1151 003316 316 377 377 TSTUS JSB =ONER get real value
1152 003321 316 371 006 JSB =CALSCH save reg and scan
1153 003324 373 343 JCY P$RTH
1154 003326 DRP !54
1155 003326 221 TSM R54
1156 003327 366 014 JNZ UFMT jif not
1157 003331 316 200 007 KFMT JSB =ARITH
1158 003334 230 BIN
1159 003335 170 220 TSB R70 test sign
1160 003337 364 002 JNG KFMT1 jif K format
1161 003341 154 211 ICM R54 one for trailing space
1162 003343 166 223 KFMT1 CLM R66
1163 003345 173 220 UFMT TSB R73
1164 003347 364 326 JNG FMTER1 jif not arith fmt
1165 003351 166 006 345 PUMD R66,+R6
1166 003354 316 377 377 JSB =RESFIL reserve and zero fill
1167 003357 166 006 343 POMD R66,-R6
1168 003362 370 305 JEN P$RTH jif not enough room
1169 003364 316 070 007 JSB =FORMN
1170 003367 360 275 JMP COMRTN
1171 003371
1172 003371 140 006 345 CALSCH PUMD R40,+R6 save it
1173 003374 120 345 PUMD R20,+R6 save R20
1174 003376 316 377 377 JSB =SVPTRS save PTR1 & PTR2
1175 003401 316 044 010 JSB =SCANN go scan image
1176 003404 316 377 377 JSB =RSPTRS restore pointers
=====

```

KARMA 03/14/81=====

ITEM	LOC	OBJECT CODE	SRC=QRNMRT	OBJ=GEZNM3	9/11/1981	12:08 PM	PG 30
------	-----	-------------	------------	------------	-----------	----------	-------

=====

> >>>> USING <<<<<<<<

1	7	003407		ARP	16		
1178		003407	120 343	POMD	R20,-R6	restore R20	
1179		003411	140 343	POMD	R40,-R6	also R40	
1180		003413	117 310 300	CMB	R17,=300	errors?	
1181		003416	154	ORP	54	default format?	
1182		003417	236	RTN			
1183		003420					

```

KARMA 03/14/81=====
ITEM LOC OBJECT CODE SRC=ORNMRT OBJ=GEZNM3 9/11/1981 12:08 PM PG 31
=====
>>>> FORMN/CVNUM <<<<<<
1185 003420 *****
1185 003420 *
1186 003420 * CVNUM: Formats a floating point number into
1187 003420 * ASCII for printing,
1188 003420 * IN: R30 = output buffer
1189 003420 * R40 = floating point #
1190 003420 * OUT: R26 = FWA converted number
1191 003420 *
1192 003420 * Note: the formatting is done in a reserved
1193 003420 * area, possibly in extended memory, and
1194 003420 * then copied to the output buffer
1194 003420 *
1194 003420 *****
1195 003420
1196 003420 130 026 243 CVNUM STM R30,R26 save for CRT output
1197 003423 316 050 007 JSB =CVNUMG call GEMINI routine
1198 003426 176 054 241 FORMDC LDM R76,R54 byte count
1199 003431 316 377 377 JSB =DCSLQP copy from resmem
1200 003434 316 377 377 JSB =RELMEM
1201 003437 236 RTN
1202 003440
1203 003440 *****
1203 003440 *
1204 003440 * FORMN+: entry point for external routines.
1205 003440 *
1206 003440 * IN: R30 = output buffer
1207 003440 * R54 = byte count
1208 003440 * PTR1 = data to be formatted
1209 003440 * OUT: R26 = FWA of formatted data
1209 003440 *
1209 003440 *****
1210 003440
1211 003440 130 026 241 FORMN+ LDM R30,R26 get output buffer
1212 003443 316 056 007 JSB =FORMG call GEMINI routine
1213 003446 360 356 JMP FORMDC copy and exit
1214 003450
1215 003450 170 222 CVNUMG CLB R70
1216 003452 212 DCB R70 SIGN FLAG NEG
1217 003453 316 200 007 JSB =ARITH
1218 003456 316 377 377 FORMG JSB =RM+SET reserve space
1219 003461 370 020 JEN FORMOV jif no room
1220 003463 316 377 377 FORMG+ JSB =RESFL1 blank it
1221 003466 166 223 CLM R66 no C or P
1222 003470 172 220 FORMN TSB R72 I/O image?
1223 003472 365 010 JPS FORMNC jif no
1224 003474 102 251 003 LDM R2,=3,0 tell em what we expect
1224 003477 000
1225 003500 316 324 207 JSB =IMERR
1226 003503 236 FORMOV RTN
1227 003504
1228 003504 231 FORMNC BCD
1229 003505 316 377 377 JSB =SVPTRS save PTR2
1230 003510 ARP 16

```

```

>>>> FORMN/CVNUM
1 1 003510 154 345 PUMD R54,+R6 save length
1232 003512 166 345 PUMD R66,+R6 save CP count
1233 003514 316 377 377 JSB =SEP10
1234 003517 316 263 013 JSB =ROUND
1235 003522 173 220 TSB R73
1236 003524 366 006 JNZ FRMNI jif E format
1237 003526 316 377 377 JSB =INFR4
1238 003531 136 034 241 LDM R36,R34 exp to R36
1239 003534 166 006 343 FRMNI POMD R66,-R6 restore C/P count
1240 003537 316 355 013 JSB =OUTC/P go output them
1241 003542 316 107 015 JSB =SIGNH
1242 003545 316 044 014 JSB =LDIGIT
1243 003550 316 167 015 JSB =RADIX
1244 003553 316 177 015 JSB =RDIGIT
1245 003556 316 245 015 JSB =EFORM handle rest of #
1246 003561 230 BIN THESE 3 SUBROUTINES CAN BE
1247 003562 154 006 343 POMD R54,-R6 COMBINED IF NECESSARY
1248 003565 316 377 377 JSB =RSPTRS restore count
1249 003570 * DECR COUNT BY R2 SET IN EFORM restore pointers
1250 003570 174 223 CLM R74
1251 003572 002 240 LDB R74,R2
1252 003574 154 074 305 SBM R54,R74
1253 003577 236 RTN
1254 003600
1254 003600
1255 003600
1256 003600
1257 003600
1258 003600
1259 003600
1260 003600
1261 003600
1261 003600
1261 003600
1262 003600
1263 003600 171 250 004 ARITH LDB R71,=4 FILL IS ZEROS
1264 003603 172 223 CLM R72
1265 003605 250 056 LDB R72,=PERIOD
1266 003607 136 040 241 LDM R36,R40 EXP/SIGN
1267 003612 034 243 STM R36,R34 DUP IT
1268 003614 231 BCD
1269 003615 137 206 LRB R37 TRASH SIGN
1270 003617 162 042 241 LDM R62,R42 MANTISSA
1271 003622 366 004 JNZ TSEXP JIF NON ZERO
1272 003624 176 211 ICM R76 CHR LT = 1
1273 003626 360 022 JMP INT? JMP TO SETUP EXIT
1274 003630 137 310 005 TSEXP CMB R37,=5
1275 003633 365 136 JPS NGEXP JIF NEG EXP
1276 003635 136 311 022 CNM R36,=120,0
1276 003640 000
1277 003641 365 113 JPS NR3 JIF EXP > 11
1278 003643 176 036 240 LDB R76,R36
1  ) 003646 211 ICM R76

```

```

KARMA 03/14/81=====
ITEM   LOC   OBJECT CODE   SRC=ORNMRT  OBJ=GEZNM3   9/11/1981 12:08 PM  PG 33
=====
>>>>>> FORMN/CVNUM <<<<<<<<
1280 003647 316 016 010 NR12 JSB =CHRRT
1281 003652 136 074 241 INT? LDM R36,R74 CHR RT
1282 003655 366 002 JNZ SETCNT SET UP RETURN
1283 003657 172 222 CLB R72
1284 003661 156 074 241 SETCNT LDM R56,R74 CRT RT
1285 003664 076 303 ADM R56,R76 + CHR RT
1286 003666 170 220 TSB R70 TEST SIGN
1287 003670 364 002 JNG INCS?
1288 003672 366 004 JNZ IHCS GO INC IF # 0
1289 003674 141 220 IHCS? TSB R41
1290 003676 376 002 JRZ NOINCS JIF POS
1291 003700 156 211 IHCS ICM R56
1292 003702 172 220 NOINCS TSB R72
1293 003704 367 002 JZR NODEC JIF NO DECIMAL
1294 003706 156 211 ICM R56 +DEC
1295 003710 173 220 NODEC TSB R73
1296 003712 367 004 JZR ARDON+ JIF NOT E
1297 003714 156 313 005 ADM R56,=5,0 FOR EXP
1298 003717 000
1299 003720 156 054 243 ARDON+ STM R56,R54
1300 003723 223 CLM R56
1301 003724 170 006 345 ARDON PUMD R70,+R6
1302 003727 166 345 PUMD R66,+R6 SAVE C/P COUNT
1303 003731 160 223 CLM R60
1304 003733 250 007 LDB R60,=7
1305 003735 164 054 241 LDM R64,R54
1306 003740 316 377 377 JSB =CONINT CONV TO BIN
1307 003743 154 223 CLM R54
1308 003745 176 054 243 STM R76,R54
1309 003750 166 006 343 PUMD R66,-R6 RESTORE C/P COUNT
1310 003753 170 343 PUMD R70,-R6
1311 003755 236 RTN
1312 003756 162 042 241 NR3 LDM R62,R42 MANTISSA AGAIN
1313 003761 176 211 ICM R76 CHR LT = 1
1314 003763 316 016 010 JSB =CHRRT
1315 003766 173 250 002 LDB R73,=2 E=2
1316 003771 360 266 JMP SETCNT
1317 003773 136 217 NGEXP HCM R36 MAKE EXP POS
1318 003775 137 317 017 ANM R37,=17 # LEAD ZEROS TO BRING
1319 004000 136 221 TSM R36
1320 004002 367 243 MYAGN JZR NR12 JIF NONE
1321 004004 167 207 LRM R67
1322 004006 161 220 TSB R61
1323 004010 375 344 JLN NR3 JIF SHIFTED OUT NON ZERO
1324 004012 136 213 DCM R36
1325 004014 360 364 JMP MYAGN
1326 004016 134 251 022 CHRRT LDM R34,=120,0 MAX CHR RT
1327 004021 000
1328 004022 076 305 SBM R34,R76 MAX RT - LT = MAX
1329 004024 367 012 JZR CHRTEX JIF ZERO
1330 004026 162 220 SHFAGN TSB R62
1331 004030 377 006 JRN CHRTEX JIF NEXT DIGIT # 0
1332 004032 167 207 LRM R67

```

KARMA 03/14/81=====

ITEM	LOC	OBJECT CODE	SRC=ORNMRT	OBJ=GE2NM3	9/11/1981	12:08 PM	PG 34
=====							
>>>>		FORMN/CVNUM			<<<<<<<<		
1	004034	134	213	DCM	R34		
1332	004036	366	366	JNZ	SHPAGN		
1333	004040	134	074 243	CHRTX	STM R34,R74		
1334	004043	236		RTN			

```

>>>> SCANN - SCAN FOR FORMAT ITEM <<<<<<
1336 004044 *****
1336 004044 *
1337 004044 * SCANN - scan for format item
1338 004044 *
1339 004044 * ENTRY
1340 004044 * PTR1- CURRENT IMAGE POINTER
1341 004044 * R30 - CHAR IN IMAGE
1342 004044 *
1343 004044 * EXIT
1344 004044 * R54 - CHAR CNT (<=0 IF DEFAULT FORMAT)
1345 004044 * R70 - SIGN
1346 004044 * -1 - IF NEG ELSE NULL
1347 004044 * 0 - IN DIG LEFT IF NEG ELSE NULL
1348 004044 * 1 S - IF NEG ELSE + (DEFAULT)
1349 004044 * 2 M - IF NEG ELSE BLANK
1350 004044 * R71 - FILL
1351 004044 * -1 DEFAULT
1352 004044 * 0 DZ
1353 004044 * 1 D
1354 004044 * 2 *Z
1355 004044 * 3 *
1356 004044 * 4 Z
1357 004044 * R72 - DECIMAL (ZERO, COMMA, OR PERIOD)
1358 004044 * R73 - E FORMAT/A FORMAT
1359 004044 * -1 A
1360 004044 * 0 NONE
1361 004044 * 1 E FORMAT (IMAGE)
1362 004044 * 2 E FORMAT (SUPPRESS +, -, AND LEAD ZEROS)
1363 004044 * R74 - CHR RT
1364 004044 * R76 - CHR LT
1365 004044 * R66 - C/P CNT
1365 004044 *
1365 004044 *****
1366 004044

```

ITEM	LOC	OBJECT CODE	SRC=ORNMRT	OBJ=GE2NN3	9/11/1981	12:08 PM	PG 35
1367	004044	230	SCANN	BIN			
1368	004045	154 223		CLM R54		clear character count	
1369	004047	146 260 064		LDBD R46,=USING?		get USING flag	
1370	004052	200					
1371	004053	266 005		JNZ SCANUS		jif USING	
1372	004055	170 223		CLM R70		clear SCANN flags	
1373	004057	250 002		LDB R70,=2		sign (M)	
1374	004061	236		RTN			
1375	004062	316 157 011	SCANUS	JSB =SCANL		scan past leading garbage	
1376	004065	166 223		CLM R66		C/P count	
1377	004067	170 223		CLM R70		SCANN flags	
1378	004071	171 212		DCB R71		default fill	
1379	004073	154 223		CLM R54		clear character count	
1380	004075	370 005		JEN SCANL1		jif not wraparound	
1381	004077	360 163		JMP FMTER2		else error	
1382	004101	316 074 013	SCANLP	JSB =GETREP		get replication in R44	
1383	004104	316 361 012	SCANL1	JSB =FCHAR+		get chr flag in R36	
1384	004107	136 205		LLM R36		times 2	

```

KARMA 03/14/81=====
ITEM LOC OBJECT CODE SRC=ORNMRT OBJ=GEZNM3 9/11/1981 12:08 PM PG 36
=====
>>>>> SCANN - SCAN FOR FORMAT ITEM <<<<<<<
1 5 004111 036 265 126 LDMD R36,X36, TYTAB address
1385 004114 010
1386 004115 306 000 000 JSB X36,ZRO go doit
1387 004120 117 310 300 CMB R17,=300 errors
1388 004123 372 354 JNC SCANLP loop if no
1389 004125 236 RTN else exit
1390 004126
1391 004126 044 011 TYTAB DEF IMERR? 0 END
1392 004130 141 011 DEF STRNG 1 A
1393 004132 162 010 DEF ZEE 2 Z
1394 004134 251 010 DEF STARR 3 *
1395 004136 266 010 DEF DEE 4 D
1396 004140 306 010 DEF RADX 5 .
1397 004142 303 010 DEF RADXR 6 R
1398 004144 324 010 DEF SIGNF 7 S
1399 004146 324 010 DEF SIGNF 10 M
1400 004150 356 010 DEF SEP 10 C
1401 004152 347 010 DEF SEPS 11 P
1402 004154 012 011 DEF EEE 11 E
1403 004156 030 011 DEF KAA 12 K
1404 004160 044 011 DEF IMERR? 13 ILLEGAL OR SEP
1405 004162
1406 004162 132 250 004 ZEE LDB R32,=4 fill code
1407 004165 171 220 TSB R71
1408 004167 364 042 JNG SFILL jif no fill chr yet
1409 004171 032 300 CMB R71,R32
1410 004173 367 040 JZR SFILL1 jif fill type 4
1411 004175 144 213 DCM R44
1412 004177 366 063 JNZ FMTER2 error if cnt<>1
1413 004201 211 ICM R44
1414 004202 120 310 056 CMB R20,=PERIOD
1415 004205 367 020 JZR DCFILL jif next .
1416 004207 310 122 CMB R20,=122 R?
1417 004211 367 014 JZR DCFILL jif R
1418 004213 316 201 013 JSB =CKSEP see if separator
1419 004216 371 044 JEZ FMTER2 jif not
1420 004220 316 227 010 JSB =DCFILL go update count
1421 004223 104 251 116 GTO DONE? and exit since sep
1422 004226 011
1422 004227 171 212 DCFILL DCB R71
1423 004231 360 002 JMP SFILL1
1424 004233 032 240 SFILL LDB R#,R32 fill to R71
1425 004235 172 220 SFILL1 TSB R72
1426 004237 366 023 JNZ FMTER2 jif decimal seen
1427 004241 231 CHRAD- BCD
1428 004242 176 044 303 ADM R76,R44 CHR LT + REPL
1429 004245 154 044 303 CHRAD ADM R54,R44 CHR + REPL
1430 004250 236 CHRADX RTN
1431 004251
1432 004251 132 250 003 STARR LDB R32,=3 FILL CODE
1433 004254 171 220 STARR1 TSB R71
1434 004256 364 353 JNG SFILL JIF FILL NEG
1435 004260 032 300 CMB R71,R32 FILL MATCH?

```

```

KARMA 03/14/81=====
ITEM      LOC      OBJECT CODE  SRC=ORNMRT OBJ=GEZNM3   9/11/1981 12:08 PM PG 37
=====
J >>>> SCANN - SCAN FOR FORMAT ITEM
1436 004262 367 351 JZR SFILL1 JIF YES
1437 004264 360 167 FMTER2 JMP FMTER
1438 004266
1439 004266 132 250 001 DEE LDB R32,=1 FILL
1440 004271 172 220 TSB R72
1441 004273 367 357 JZR STAR1 JIF NO DECIMAL
1442 004275 231 BCD
1443 004276 174 044 303 ADM R74,R44 CHR RT + REPL
1444 004301 360 342 JMP CHRAD GO ADD CHRS
1445 004303
1446 004303 114 250 054 RADXR LDB R14,=COMMA
1447 004306 172 220 RADX TSB R72
1448 004310 366 143 JNZ FMTER JIF DECIMAL SEEN
1449 004312 014 240 LDB R72,R14 SET , OR COMMA
1450 004314 231 CREP1 BCD
1451 004315 144 213 DCM R44
1452 004317 366 134 JNZ FMTER JIF REPL # 1
1453 004321 211 ICM R44
1454 004322 360 321 JMP CHRAD GO ADD CHRS
1455 004324
1456 004324 170 220 SIGNF TSB R70
1457 004326 366 125 JNZ FMTER JIF SIGN SEEN
1458 004330 154 221 TSM R54
1459 004332 366 121 JNZ FMTER JIF NOT FIRST CHAR
1460 004334 114 310 123 CNB R14,=123
1461 004337 367 002 JZR INC1 JIF S
1462 004341 170 210 ICB R70
1463 004343 170 210 INC1 ICB R70
1464 004345 360 345 JMP CREP1
1465 004347
1466 004347 316 374 010 SEPS JSB =SEP1 GO CONVERT TO BIN
1467 004352 134 215 TCM R34 NEG TO FLAG PERIOD
1468 004354 360 003 JMP SEP2 GO FINISH UP
1469 004356 316 374 010 SEP JSB =SEP1 GO CONV TO BIN
1470 004361 134 012 345 SEP2 PUMD R34,+R12 PUSH CHR CNT
1471 004364 166 211 ICM R66 C/P CNT
1472 004366 176 221 TSM R76 CHR LT 0?
1473 004370 367 063 JZR FMTER ERROR IF SO
1474 004372 360 320 JMP CREP1
1475 004374 154 006 345 SEP1 PUMD R54,+R6 SAVE 54
1476 004377 316 324 007 JSB =ARDON GO CONVERT
1477 004402 134 054 241 LDM R34,R54 RESULT TO 34
1478 004405 154 006 343 PUMD R54,-R6 RESTORE 54
1479 004410 230 BIN
1480 004411 236 SPRTN RTN AND EXIT
1481 004412 *
1482 004412 173 220 EEE TSB R73
1483 004414 366 037 JNZ FMTER JIF E OR A SEEN
1484 004416 210 ICB R73
1485 004417 231 BCD
1486 004420 144 213 DCM R44
1487 004422 366 031 JNZ FMTER JIF REPL # 1
1488 004424 250 005 LDB R44,=5 REPL = 4

```

```

KARMA 03/14/81=====
ITEM   LOC   OBJECT CODE SRC=ORNMRT OBJ=GE2NM3  9/11/1981 12:08 PM PG 38
=====
>>>>>  SCANN - SCAN FOR FORMAT ITEM                                     <<<<<<<
1  3 004426 360 215                                JMP CHRAD
1490 004430                                         *
1491 004430 170 212      KAA      DCB R70
1492 004432 231                                BCD
1493 004433 154 221                                TSM R54
1494 004435 366 016                                JNZ FMERR      JIF CHR CNT # 0
1495 004437 144 213                                DCM R44
1496 004441 366 012                                JNZ FMERR      JIF REPL # 1
1497 004443 236                                RTN
1498 004444                                         *
1499 004444 316 201 013 IMERR? JSB =CKSEP      CIF SEPERATOR
1500 004447 371 010      JEZ CKCLO      JIF NOT SEP
1501 004451 144 213                                DCM R44
1502 004453 367 042                                JZR DONE?      JIF REPL = 1
1503 004455 316 377 377 FMERR JSB =ERROR+
1504 004460 064                                OCT 520
1505 004461 120 310 051 CKCLO  CMB R20,=CLOSE
1506 004464 367 023                                JZR CLSFND      Changed 8 23 78
1507 004466 102 223                                CLM R2
1508 004470 211                                ICM
1509 004471 211                                ICM
1510 004472 316 324 207      JSB =IMERR      TELL EM WHAT WE EXPECT
1511 004475 102 221                                TSM R2
1512 004477 364 004                                JNG CKCLOR      FOUND SEP
1513 004501 316 377 377      JSB =ERROR      JIF YES
1514 004504 064                                OCT 520      SET ERROR ROM MAY CLEAR
1515 004505 106 213      CKCLOR DCM R6
1516 004507 213                                DCM R6
1517 004510 236                                RTN      TRASH 1 RETURN
1518 004511
1519 004511 120 272 312 CLSFND STBI R20,=PTR1+      so SCANL sees ')'
1519 004514 377
1520 004515 130 211                                ICM R30
1521 004517 130 263 366 DONE? STMD R30,=IMCLEN      make count right
1521 004522 203                                LDMD R45,=PTR1      current image len
1522 004523 145 261 310
1522 004526 377                                STMD R45,=IMCADR      get position
1523 004527 263 370 203                                CLM R45      current image addr
1524 004532 223                                JSB =ARDON      CONV TO BIN IN 54
1525 004533 316 324 007                                BIN
1526 004536 230                                JMP CKCLOR
1527 004537 360 344
1528 004541
1529 004541 173 220      STRNG  TSB R73
1530 004543 366 310                                JNZ FMERR      jif E or A seen
1531 004545 212                                DCB R73      set A format
1532 004546 154 044 303      ADM R54,R44      store count in R54
1533 004551 236                                RTN
1534 004552      *****
1534 004552      *
1535 004552      * SCANL - scan leading garbage
1536 004552      *
1537 004552      * ENTRY

```

```

KARMA 03/14/81=====
ITEM LOC OBJECT CODE SRC=ORNMRT OBJ=GEZNM3 9/11/1981 12:08 PM PG 39
=====
>>>>>>> SCANN - SCAN FOR FORMAT ITEM <<<<<<<<
1538 004552 * R65 - ACTIVE REPL CNT
1539 004552 * PTR1- CURRENT IMAGE POINTER
1540 004552 * R30 - CHAR IN IMAGE
1541 004552 * R20 - NEXT CHAR
1542 004552 * EXIT
1543 004552 * PTR1- CURRENT IMAGE POINTER
1544 004552 * R30 - CHAR IN IMAGE
1544 004552 *
1544 004552 *****
1545 004552
1546 004552 130 042 050 LTABL ASC 5,X"(<>? garbage table
1546 004555 051 057
1547 004557 130 261 366 SCANL LDMD R30,=IMCLEN image len
1547 004562 203
1548 004563 175 261 370 LDMD R75,=IMCADR image addr
1548 004566 203
1549 004567 263 310 377 STMD R75,=PTR1 set pointer
1550 004572 230 SCANLX BIN
1551 004573 175 263 373 STMD R75,=IMWADR wrap address
1551 004576 203
1552 004577 316 152 013 SCANLGO JSB =GCHAR# next char to R20
1553 004602 371 023 JEZ SCANLS jif end
1554 004604 316 257 011 SCANLR JSB =WTEST check wrap
1555 004607 316 074 013 JSB =GETREP get replication
1556 004612 134 251 152 LDM R34,=LTABL addr of char table
1556 004615 011
1557 004616 124 251 006 LDM 24,=6,0 # chars in table
1557 004621 000
1558 004622 316 371 012 JSB =FCHAR find inchar in table
1559 004625 370 050 JEN PROCHR jif found
1560 004627 316 257 011 SCANLS JSB =WTEST test wrap
1561 004632 316 201 013 JSB =CKSEP separator?
1562 004635 371 005 JEZ SCLRT+ jif no
1563 004637 316 257 011 JSB =WTEST test wrap
1564 004642 360 333 JMP SCANLGO and loop
1565 004644 102 223 SCLRT+ CLM R2 tell em what we expect
1566 004646 316 324 207 JSB =IMERR image error hook
1567 004651 102 221 TSM R2
1568 004653 364 327 JNG SCANLR jif if found one
1569 004655 234 ICE clear wraparound
1570 004656 236 SCLRTX RTN
1571 004657
1572 004657 235 WTEST CLE
1573 004660 230 BIN
1574 004661 130 221 TSM R30
1575 004663 365 371 JPS SCLRTX jif count positiv
1576 004665 175 261 373 LDMD R75,=IMWADR wrap addr
1576 004670 203
1577 004671 321 363 203 CMMO R75,=IMSADR = start addr?
1578 004674 367 207 JZR CKCLOR jif wrap
1579 004676 236 RTN else return
1580 004677 136 205 PROCHR LLM R36 double character flag
1581 004701 036 265 316 LDMD R36,X36,LJTAB get jump addr

```

```

KARMA 03/14/81=====
ITEM LOC OBJECT CODE SRC=ORNMRT OBJ=GEZNM3 9/11/1981 12:08 PM PG 40
=====
>>>>>> SCANN - SCAN FOR FORMAT ITEM <<<<<<<<
1 004704 011
1582 004705 306 000 000 JSB X36,ZR0 go doit
1583 004710 117 310 300 CMB R17,=300 errors?
1584 004713 372 267 JNC SCANLR loop if no
1585 004715 236 RTN else exit
1586 004716 120 012 LJTAB DEF FMTER4
1587 004720 332 011 DEF EX X
1588 004722 372 011 DEF QLIT "
1589 004724 054 012 DEF LTRAC (
1590 004726 124 012 DEF RTBRAC )
1591 004730 177 012 DEF SENDCR /
1592 004732 155 044 241 EX LDM R55,R44 repl (BCD)
1593 004735 154 250 377 LDB R54,=377 flag
1594 004740 150 012 345 PUMD R50,+R12 put integer to stack
1595 004743 316 377 377 JSB =ONEB conv. to int in R40
1596 004746 154 223 CLM R54
1597 004750 146 054 243 STM R46,R54 bint in R54 for blanking
1598 004753 316 377 377 JSB =RESFIL
1599 004756 * JSB =RM+SET reserve buffer and fill
1600 004756 370 276 JEN SCLRTX jif out of room
1601 004760 130 012 345 SPECOT PUMD R30,+R12 save for next time
1602 004763 * LDMD R75,=PTR1 get pointer
1603 004763 * PUMD R75,+R12 and save it
1604 004763 316 377 377 JSB =TRAFFIC call traffic cop
1605 004766 * POMD R75,-R12
1606 004766 * STMD R75,-PTR1 restore pointer
1607 004766 130 012 343 POMD R30,-R12 restore count
1608 004771 236 RTN
1609 004772
1610 004772 +QLIT LDMD R45,=PTR1 save start
1611 004772 * ICM R45 start+1 ("back up" one)
1612 004772 * STMD R45,=PTR2
1613 004772 * DCM R45
1614 004772 +QLCOM CMB R20,=QUOTE restore
1615 004772 * JZR SMOV end-quote?
1616 004772 * JSB =GCHAR- next non blank char
1617 004772 * JEZ FMTER4 jif none
1618 004772 * JMP QLCOM loop
1619 004772 +QMOV SBMD R45,=PTR1 byte count
1620 004772 * STM R45,R54 save two byte count
1621 004772 * CLM R56
1622 004772 * JSB =GCHAR# get next character
1623 004772 * JMP SPECOT
1624 004772 *
1625 004772 *****
1625 004772 *
1626 004772 * FIX FOR LABEL USING IN GRAPHICS
1626 004772 *
1626 004772 *****
1627 004772 145 261 310 QLIT LDMD R45,=PTR1 save start
1627 004775 377
1628 004776 211 ICM R45 move up one
1629 004777 065 243 STM R45,R65 make a copy

```

```

>>>>> SCANN - SCAN FOR FORMAT ITEM
1630 005001 120 310 042 9LCOM CMB R20,=QUOTE second quote?
1631 005004 267 007 JZR QMOV jif yes
1632 005006 316 165 013 JSB =GCHAR- get non-blank
1633 005011 371 105 JEZ FMTER4
1634 005013 260 364 JMP 9LCOM
1635 005015 145 325 310 QMOV SBMD R45,=PTR1 calc length
1635 005020 377
1636 005021 213 DCM R45
1637 005022 054 243 STM R45,R54 for reserving
1638 005024 156 223 CLM R56
1639 005026 165 263 310 STMD R65,=PTR1 restore PTR1
1639 005031 377
1640 005032 316 377 377 JSB =RESFIL reserve R54 bytes
1641 005035 316 377 377 JSB =EMOVUP copy $ pgm->resvd
1642 005040 165 263 314 STMD R65,=PTR2 set to start of $
1642 005043 377
1643 005044 270 311 377 LDBI R65,=PTR1- skip second quote
1644 005047 316 152 013 JSB =GCHAR# get char after "
1645 005052 260 304 JMP SPECOT
1646 005054
1647 005054 151 006 345 LTBRAC PUMD R51,+R6 save R51
1648 005057 175 261 310 LDMD R75,=PTR1 get pointer
1648 005062 377
1649 005063 211 ICM R75 make it right
1650 005064 051 243 STM R75,R51 image pointer
1651 005066 * DCM R75 don't need it
1652 005066 154 030 241 LDM R54,R30 char count
1653 005071 211 ICM R54
1654 005072 156 044 241 LDM R56,R44 repl
1655 005075 151 012 345 PUMD R51,+R12 push iptr, chr cnt & repl
1656 005100 112 321 044 CMMD R12,=FWBIN out of room?
1656 005103 200
1657 005104 373 007 JCY LTBOU jif overflow
1658 005106 165 260 064 LDBD R65,=USING?
1658 005111 200
1659 005112 210 ICB R65 incr repl count
1660 005113 266 032 JNZ RTBREX
1661 005115 151 006 343 LTBOU POMD R51,-R6 restore stak
1662 005120 316 377 377 FMTER4 JSB =ERROR+ image error
1663 005123 064 OCT 520
1664 005124 165 260 064 RTBRAC LDBD R65,=USING?
1664 005127 200
1665 005130 265 366 JPS FMTER4 jif repl cnt 0
1666 005132 151 006 345 PUMD R51,+R6 save R52
1667 005135 012 343 POMD R51,-R12 pop current repl
1668 005137 231 BCD
1669 005140 156 213 DCM R56 decr count
1670 005142 230 BIN
1671 005143 266 011 JNZ RSET jif not done this repl
1672 005145 165 212 DCB R65 decr replicate count
1673 005147 262 064 200 RTBREX STBD R#,=USING? update repl count
1674 005152 151 006 343 RTBRX1 POMD R51,-R6 restore R51
1675 005155 236 RTN
  
```

```

KARMA 03/14/81=====
ITEM    LOC    OBJECT CODE  SRC=ORNMRT OBJ=GEZNM3  9/11/1981 12:08 PM  PG 42
=====
>>>>>  SCANN - SCAN FOR FORMAT ITEM  <<<<<<<
1  6 005156 151 012 345 RSET      PUMD R51,+R12
1677 005161 130 054 241          LDM R30,R54      push it back
1678 005164 165 051 241          LDM R65,R51      char count
1679 005167 263 310 377          STMD R65,=PTR1
1680 005172 316 152 013          JSB =GCHAR#      set 1st pointer
1681 005175 260 353              JMP RTBRX1      get first char
1682 005177
1683 005177 121 250 015 SENDCR    LDB R21,=15      CR
1684 005202 316 327 012 SENDCC    JSB =SC12?      select code 1 or 2?
1685 005205 371 060              JEZ HTSTD        jif not SC 1 or 2
1686 005207 136 270 310          LDBI R36,=P.PTR
1686 005212 203
1687 005213 266 020              JNZ SENDC1      jif buff not empty
1688 005215 117 310 300 SENDC2    CMB R17,=300      errors?
1689 005220 373 122              JCY SC12RT      exit if so
1690 005222 136 222              CLB R36
1691 005224 210
1692 005225 272 310 203          STBI R36,=P.PTR    buff cnt = 1
1693 005230 250 040          LDB R36,=BLANK
1694 005232 272 306 203          STBI R36,=P.BUFF    blank char
1695 005235 144 006 345 SENDC1    PUMD R44,+R6      save repl
1696 005240 154 223          CLM R54
1697 005242 211          ICM R54
1698 005243 316 377 377          JSB =RM+SET      reserve one byte
1699 005246 370 007          JEN SENDCL      jif out of memory
1700 005250 121 272 317          STBI R21,=PTR2-+    store CR (don't move)
1701 005253 377
1701 005254 316 360 011          JSB =SPECOT
1702 005257 144 006 343 SENDCL    PUMD R44,-R6      get repl
1703 005262 231          BCD
1704 005263 213          DCM
1705 005264 266 327          JNZ SENDC2      loop if more to send
1706 005266 236          RTN
1707 005267 154 044 241 HTSTD      LDM R54,R44      repl
1708 005272 316 324 007          JSB =ARDON      convert to binary
1709 005275 316 377 377          JSB =RM+SET      reserve byte, set ptr
1710 005300 370 042          JEN SC12RT      jif out of ROOM
1711 005302 231          BCD
1712 005303 165 261 314          LDMD R65,=PTR2    save start
1712 005306 377
1713 005307 121 272 315 HTSTL      STBI R21,=PTR2-    push a CR
1713 005312 377
1714 005313 144 213          DCM R44      dec count
1715 005315 266 370          JNZ HTSTL      loop if more
1716 005317 165 263 314          STMD R65,=PTR2    restore pointer
1716 005322 377
1717 005323 104 251 357          GTO SPECOT
1717 005326 011
1718 005327 140 006 345 SC12?      PUMD R40,+R6      protect R40
1719 005332 316 377 377          JSB =COPSUB      get select code
1720 005335 235          CLE      false flag
1721 005336 367 001          JZR POPR40      jif 10
1  2 005340 234          ICE      true flag

```

```

KARMA 03/14/81=====
ITEM   LOC   OBJECT CODE SRC=ORNMRT OBJ=GENNM3   9/11/1981 12:08 PM PG 43
=====
>>>> SCANN - SCAN FOR FORMAT ITEM
1723 005341 140 006 343 POPR40  POMD R40,-R5          restore R40
1724 005344 236          SC12RT  RTN
1725 005345          *
1725 005345          *
1726 005345          * FCHAR - IDENTIFY FRMAT CHAR
1727 005345          *
1728 005345          * ENTRY
1729 005345          *   PTR1-- IMAGE POINTER
1730 005345          *   R20 - NEXT CHAR
1731 005345          *   R30 - CHAR IN IMAGE
1732 005345          *   R44 - REPL
1733 005345          *
1734 005345          * EXIT
1735 005345          *   PTR1-- IMAGE POINTER
1736 005345          *   R20 - NEXT CHAR
1737 005345          *   R30 - CHAR IN IMAGE
1738 005345          *   R44 - REPL
1739 005345          *   R14 - CHAR
1740 005345          *   R36 - CHAR FLAG
1741 005345          *
1742 005345 101 132 052 CTABL  ASC 14,AZ*D.RSMCPEK
1742 005350 104 056 122
1742 005353 123 115
1743 005355 103 120 105
1743 005360 113
1744 005361 134 251 345 FCHAR+  LDM R34,=CTABL
1744 005364 012
1745 005365 124 251 015          LDM R24,=15,0
1745 005370 000
1746 005371          *
1746 005371          * RETURNS E = 0 IF NOT FOUND
1747 005371          *
1747 005371          *
1748 005371          *
1749 005371 230          FCHAR  BIN
1750 005372 235          CLE
1751 005373 136 223          CLM R36          SET NOT FOUND
1752 005375 130 221          TSM R30          BYTES REMAINING
1753 005377 364 072          JNG FCRTN
1754 005401 120 014 242          STB R20,R14          CHAR TO 14
1755 005404 310 054          CMB R20,=COMMA
1756 005406 367 063          JZR FCRTN          EXIT IF SEP
1757 005410 316 130 013          JSB =IDCHAR          GO IDENTIFY CHAR
1758 005413 371 056          JEZ FCRTN          JIF NOT FOUND
1759 005415 316 165 013          JSB =GCHAR-          NEXT CHAR
1760 005420 371 047          JEZ FCRT1          JIF NO MORE
1761 005422          * DONT REPEAT ( ) OR "
1762 005422 114 310 050          CMB R14,=OPEN
1763 005425 367 044          JZR FCRTN
1764 005427 310 051          CMB R14,=CLOSE

```

```

KARMA 03/14/81=====
ITEM LOC OBJECT CODE SRC=GRNMPT OBJ=GENNM3 11/1981 12:08 PM PG 44
=====
>>>> SCANN - SCAN FOR FORMAT ITEM <<<<<<<<
1 005431 367 040 JZR FCRTN
1766 005433 310 042 CMB R14,=QUOTE
1767 005435 367 034 JZR FCRTN
1768 005437 120 310 040 CMB R20,=BLANK
1769 005442 266 005 JNZ TSTRP1 JIF NOT BLANK
1770 005444 316 152 013 TSTRP JSB =GCHAR# NEXT NON BLANK
1771 005447 371 020 JEZ FCRT1 JIF NO MORE
1772 005451 114 020 300 TSTRP1 CMB R14,R20
1773 005454 266 015 JNZ FCRTN JIF NO MATCH
1774 005456 231 BCD
1775 005457 144 211 ICM R44 INCR REPL
1776 005461 146 221 TSM R46 >9999
1777 005463 267 357 JZR TSTRP JIF NO
1778 005465 144 213 DCM R44 ELSE RESET
1779 005467 260 353 JMP TSTRP LOOP
1780 005471 234 FCRT1 ICE
1781 005472 230 BIN
1782 005473 236 FCRTN RTN
1783 005474
1784 005474
1784 005474
1784 005474
1785 005474
1786 005474 144 223 GETREP CLM R44
1787 005476 231 GLOP BCD
1788 005477 316 377 377 JSB =DIGIT DIGIT?
1789 005502 371 015 JEZ GLOPT JIF NO
1790 005504 145 220 TSB R45 4 DIGITS
1791 005506 375 004 JLN GLOPL JIF YES
1792 005510 120 202 ERB R20 TO E
1793 005512 144 201 ELM R44 THEN 44
1794 005514 316 152 013 GLOPL JSB =GCHAR# NEXT CHAR
1795 005517 260 355 JMP GLOP
1796 005521 144 221 GLOPT TSM R44
1797 005523 266 001 JNZ FND1 JIF NOT ZERO
1798 005525 211 ICM R# ELSE MAKE IT ONE
1799 005526 230 FND1 BIN
1800 005527 236 RTN
1801 005530
1801 005530
1802 005530
1803 005530
1804 005530
1804 005530
1804 005530
1805 005530
1806 005530 235 IDCHAR CLE
1807 005531 136 223 CLM R36
1808 005533 136 211 IDCHR1 ICM R36
1809 005535 024 301 CNM R36,R24
1810 005537 367 010 JZR FCRTN JIF NONE
1811 005541 132 034 340 POBD R32,+R34 NEXT CHAR FROM TABL

```

```

PARMA 03/14/81=====
ITEM    LOG    OBJECT CODE  SRC=ORNMRT OBJ=GENHMS  3/11/1981 12:00 PM  PG 45
=====
>>>>>>>  SCANN - SCAN FOR FORMAT ITEM
1812 005544 014 300          CMB R32,R14          IS THIS IT????
1813 005546 266 263          JNZ IDCHR!          JIF NOT FOUND
1814 005550 234              ICE                  SET FOUND
1815 005551 236              FCRTN  RTN
1816 005552
1816 005552
1817 005552
1818 005552
1818 005552
1818 005552
1819 005552
1820 005552 316 165 013 GCHAR# JSB =GCHAR-          GET NEXT CHAR
1821 005555 371 021          JEZ GCRTN          JIF NONE
1822 005557 120 310 040          CMB R20,=BLANK
1823 005562 267 266          JZR GCHAR#          LOOP IF BLANK
1824 005564 236              RTN
1825 005565 230              GCHAR-  BIN
1826 005566 235              CLE
1827 005567 130 213          DCM R30
1828 005571 264 005          JNG GCRTN          EXIT IF NEG
1829 005573 120 270 311          LDB( R20,=PTR1-  get next char
1829 005576 377
1830 005577 234              ICE
1831 005600 236              GCRTN  RTN
1832 005601
1833 005601
1833 005601
1834 005601
1835 005601
1836 005601
1836 005601
1836 005601
1837 005601 230          CKSEP  BIN
1838 005602 130 221          TSM R30          end of image?
1839 005604 265 034          JFS CKRST          jif no
1840 005606 261 361 203          LDMD R30,=IMLEN          else reset to start
1841 005611 165 006 345          FUMD R65,+R6          save C/P count
1842 005614 261 263 203          LDMD R65,=IMSADR          image start address
1843 005617 263 310 377          STMD R65,=PTR1          set pointer
1844 005622 343          FOMD R65,-R6          rest C/P count (R66)
1845 005623 235              CKINC  CLE
1846 005624 234              ICE                  FOUND SEP FLAG
1847 005625 236              RTN
1848 005626 102 223          CKRTN  CLM R2
1849 005630 211              ICM R2          TELL EM WHAT WE EXPECT
1850 005631 316 324 207          JSB =IMERR
1851 005634 102 221          TSM R2
1852 005636 264 263          JNG CKINC          JIF FOUND ONE
1853 005640 235              CLE
1854 005641 236              RTN
1855 005642 120 310 054 CKRST  CMB R20,=COMMA
1856 005645 267 354          JZR CKINC          JIF COMMA
1857 005647 310 057          CMB R20,=SLASH

```

```

FORMA 03/14/81=====
ITEM LOC OBJECT CODE SRC=GRNNRT DBJ=GENNMB 9/11/1981 12:08 PM PG 40
=====
>>>> SCANN - SCAN FOR FORMAT ITEM
1859 005651 766 353 JNZ OKRTN JIF NOT
1859 005653 120 272 312 SPSEP STBL R20,=PTR1+ BACK UP POINTER
1859 005656 377
1860 005657 130 211 ICM R30 INC COUNT
1861 005661 760 340 JMP OKINC SET FLAG AND EXIT
1862 005663
*****
1862 005663
*
1863 005663
* ROUND - ROUND NUMERIC VALUE
1864 005663
*
1865 005663
* EXP + DIGRT + 1
1866 005663
* IF NEG OR > 11 THEN EXIT, ELSE
1867 005663
* SHIFT 5 RT TO POSITION AND ADD
1868 005663
*
1869 005663
* ENTRY
1870 005663
* R32 - SIGN
1871 005663
* R36 - EXP
1872 005663
* R40 - MANTISSA W/ TRAILING ZEROS
1873 005663
* R74 - CHR RT (BCD)
1874 005663
*****
1874 005663 140 221 ROUND TSN R40
1875 005665 367 065 JZR RNDEX DONT ROUND ZERO
1876 005667 150 223 CLM R50
1877 005671 157 250 120 LDB R57,=500 FORM OF .5
1878 005674 134 036 241 LDM R34,R36 EXP
1879 005677 211 ICM R34 EXP + 1
1880 005700 173 220 TSB R73
1881 005702 367 003 JZR NRE JIF NOT E
1882 005704 134 076 241 LDM R34,R76 DIG LT
1883 005707 134 074 303 NRE ADM R34,R74 + DIG RT
1884 005712 372 010 JNC COMPR OK IF NO CARRY
1885 005714 173 220 TSB R73
1886 005716 766 034 JNZ RNDEX EXIT IF E
1887 005720 137 220 TSB R37 EXP SIGN
1888 005722 374 030 JLZ RNDEX JIF EXP POS
1889 005724 134 311 022 COMPR CMM R34,=120,0
1889 005727 000
1890 005730 372 003 JNC POS.5L JIF < 12
1891 005732 236 RTN
1892 005733 157 207 POS.5 LRM R57
1893 005735 134 212 POS.5L DCB R34
1894 005737 373 372 JCY POS.5
1895 005741 142 052 303 ADD.5 ADM R42,R52
1896 005744 372 006 JNC RNDEX
1897 005746 147 207 LRM R47
1898 005750 312 020 ACB R47,=100
1899 005752 136 211 ICM R36 INC EXP
1900 005754 236 RNDEX RTN
1901 005755
*****
1901 005755
*
1902 005755
* OUTC/P - OUTPUT COMMA/PERIOD SEPERATORS
1903 005755
*
1903 005755
* ENTRY

```

```

MARMA 03/14/81=====
ITEM LOC 06JSDT CODE SRC=ORNMRT 06J-GENNM3 9/11/1981 12:03 PM PG 47
=====
/ >>>> SCANN - SCAN FOR FORMAT ITEM <<<<<<<<
1905 005755 * PTR2 - BUFFER FWA
1906 005755 * R66 - C/P COUNT
1907 005755 * R12 - STACK POINTER
1908 005755 * C/P CHR POS,C IF POS ELSE P
1908 005755 *
1909 005755 *****
1910 005755 166 221 OUTC/P TSM R66
1911 005757 367 062 JZR OUTCEX jif none
1912 005761 230 BIN
1913 005762 145 006 345 PUMD R45,+R6 save R45
1914 005765 155 345 PUMD R55,+R6 save R55
1915 005767 261 314 377 LDMD R55,=PTR2 copy of FWA
1916 005772 134 250 054 OUTCLP LDB R34,=COMMA
1917 005775 124 012 343 PUMD R24,-R12
1918 006000 365 004 JPS PUIT jif period
1919 006002 215 TCM R24 make it negativ
1920 006003 134 250 056 LDB R34,=PERIOD
1921 006006 145 223 PUIT CLM R45
1922 006010 124 045 243 STM R24,R45 dist from FWA
1923 006013 145 215 TCM R45 complement it
1924 006015 055 303 ADM R45,R55 make it absolute
1925 006017 263 314 377 STMD R45,=PTR2
1926 006022 134 272 315 STBI R34,=PTR2- push a comma/a period
1926 006025 377
1927 006026 166 213 DCM R66 DECREMENT CNT
1928 006030 366 340 JNZ OUTCLP JIF NOT DONE
1929 006032 155 263 314 STMD R55,=PTR2
1929 006035 377
1930 006036 006 343 PUMD R55,-R6 restore R55
1931 006040 145 343 PUMD R45,-R6 restore R45
1932 006042 231 BCD
1933 006043 236 OUTCEX RTN
1934 006044 *****
1934 006044 *
1935 006044 * LDIGIT - OUTPUT DIGITS LEFT OF DECIMAL
1936 006044 *
1937 006044 * ENTRY
1938 006044 * PTR2 - OUTPUT BUFFER POINTER
1939 006044 * R76 - DIGITS LEFT
1940 006044 * R50 - DIGITS JUST RT
1941 006044 * R36 - EXP
1941 006044 *
1942 006044 *****
1943 006044 173 220 LDIGIT TSB R73
1944 006046 367 003 JZR LD1 JIF NOT E
1945 006050 150 040 241 LDM R50,R40. VAL TO 52
1946 006053 150 221 LD1 TSM R50
1947 006055 375 007 JLN NORM JIF NORMALIZED
1948 006057 366 002 JNZ RSHFT JIF NON ZERO
1949 006061 360 003 JMP NORM ELSE DONT NORM ZERO
1950 006063 205 RSHFT LLM R# R#=R50

```

```

KARMA 03/14/81=====
ITEM   LOC   OBJECT CODE SRC=ORNMRT OBJ=GENNM3  9/11/1981 12:08 PM PG 43
=====
>>>>> SCANN - SCAN FOR FORMAT ITEM <<<<<<<
1 1 006064 374 375 JLZ NSHFT JIF NOT NORMALIZED
1952 006066 134 076 241 NORM LDM R34,R76 CHT LT (BIN)
1953 006071 173 220 TSB R73
1954 006073 367 004 JZR NOTE JIF NOT E
1955 006075 136 305 SBM R36,R76 EXP-CHR LT
1956 006077 360 074 JMP OUTE
1957 006101 136 024 243 NOTE STM R36,R24 EXP
1958 006104 375 003 JLN FILLIT JIF NEG EXP
1959 006106 134 305 SBM R34,R24
1960 006110 213 DCM R34 CHRLT-(EXP+1)
1961 006111 134 221 FILLIT TSM R34
1962 006113 364 124 JNG OVFLD JIF OVERFLOW
1963 006115 367 031 JZR SETFIL JIF NO LEAD ZEROS
1964 006117 316 267 014 JSB =FILL
1965 006122 172 221 TSM R72 CIF DONE
1966 006124 367 103 JZR LDEXIT JIF YES
1967 006126 * looking for C or P
1968 006126 123 270 317 LDBI R23,=PTR2-+ GET ONE MORE
1969 006131 377
1970 006132 310 040 CMB R23,=BLANK IS IT BLANK?
1971 006134 367 012 JZR SETFIL JIF YES
1972 006136 122 310 060 CMB R22,=60 IS FILL ZEROS?
1973 006141 366 002 JNZ PUFILL JIF NO
1974 006143 023 240 LDB R22,R23 SAME CHAR READ
1975 006145 272 315 377 PUFILL STBI R#,=PTR2- ELSE FILL IT
1976 006150 171 310 002 SETFIL CMB R71,=2 BLANK FILL?
1977 006153 373 003 JCY SETFL1 JIF NO
1978 006155 316 011 015 JSB =SIGNFF GO DO FLOATING SIGN
1979 006160 171 250 004 SETFL1 LDB R71,=4 FILL IS ZEROS
1980 006163 134 036 241 LDM R34,R36 EXP
1981 006166 375 041 JLN LDEXIT JIF NEG EXP
1982 006170 311 022 000 CMM R34,=12C,0
1983 006173 365 014 JPS OUTMO JIF EXP > 11
1984 006175 211 OUTE ICM R#
1985 006176 134 221 TSM R34
1986 006200 367 027 JZR LDEXIT JIF NONE TO OUTPUT
1987 006202 375 035 JLN OVFLD OR NEG
1988 006204 316 250 014 JSB =OUTDIG OUT EXP + 1 DIGITS
1989 006207 360 020 JMP LDEXIT
1990 006211 251 022 000 OUTMO LDM R#,=12C,0
1991 006214 316 250 014 JSB =OUTDIG OUTPUT 12 DIGITS
1992 006217 134 036 241 LDM R34,R36 EXP
1993 006222 211 ICM R34
1994 006223 315 022 000 SBM R34,=12C,0 OUTPUT (EXP+1)-12 ZEROS
1995 006226 316 267 014 JSB =FILL
1996 006231 173 220 LDEXIT TSB R73
1997 006233 366 003 JNZ LDEX1 JIF E
1998 006235 150 040 241 LDM R50,R40
1999 006240 236 LDEX1 RTN
2000 006241 316 377 377 OVFLD JSB =ERROR OVERFLOW WARNING
2001 006244 002 OCT 20 OVERFLOW
2002 006245 174 223 CLM R74 SO RDIGIT WON'T TANK
2 006247 236 RTN

```

```

>>>> SCANN - SCAN FOR FORMAT ITEM
2003 006250 150 205 OUTOIG LLN R50
2004 006252 122 222 CLB R22
2005 006254 200 ELB R22
2006 006255 230 BIN
2007 006256 312 060 HDB R22,=60 MAKE IT ASCII
2008 006260 231 BCD
2009 006261 316 333 014 JSB =OUT1 GO OUTPUT ONE DIGIT
2010 006264 366 362 JNZ OUTOIG LOOP IF NOT DONE
2011 006266 236 RTN
2012 006267 122 250 040 FILL LDB R22,=BLANK
2013 006272 171 310 002 CMB R71,=2
2014 006275 364 013 JNG TSTZ FILL IS BLANK
2015 006277 122 250 052 LDB R22,=STAR
2016 006302 171 310 004 CMB R71,=4
2017 006305 364 003 JNG TSTZ FILL IS *
2018 006307 122 250 060 LDB R22,=60 FILL IS ZERO
2019 006312 176 213 TSTZ DCM R76
2020 006314 366 007 JNZ OUTIC JIF NOT LAST
2021 006316 171 220 TSB R71
2022 006320 362 003 JGD OUTIC JIF NOT DZ OR *Z OR Z
2023 006322 122 250 060 LDB R22,=60 ZERO FILL THIS CHAR
2024 006325 316 333 014 OUTIC JSB =OUT1 GO OUTPUT ONE DIGIT
2025 006330 366 360 JNZ TSTZ LOOP IF MORE
2026 006332 236 RTN
2027 006333 100 222 OUT1 CLB R0 CHAR TO OUTPUT
2028 006335 123 270 317 LODI R23,=PTR2-+ LOAD NEXT
2029 006341 310 040 CMB R23,=BLANK BLANK?
2030 006343 367 024 JZR OUT1X JIF YES
2031 006345 134 211 ICM R34 ELSE DONT COUNT IT
2032 006347 230 BIN
2033 006350 122 310 040 CMB R22,=BLANK BLANK FILL?
2034 006353 367 006 JZR OUT1X- JIF BLANK
2035 006355 310 052 CMB R22,=STAR * FILL?
2036 006357 367 002 JZR OUT1X- JIF * FILL
2037 006361 100 210 ICB R0 ELSE PUSH OUT SAME CHAR
2038 006363 231 OUT1X- BCD
2039 006364 316 371 014 JSB =OUT1X GO DOIT
2040 006367 360 342 JMP OUT1 AND LOOP
2041 006371
2042 006371 100 220 OUT1X TSB R0 R22 or R23?
2043 006373 366 004 JNZ LD23
2044 006375 022 240 LDB R0,R22
2045 006377 360 002 JMP STOR0
2046 006401 LD23 ORP !0
2047 006401 023 240 LDB R0,R23
2048 006403 STOR0 ORP !0
2049 006403 272 315 377 STBI R0,=PTR2-
2050 006406 134 213 DCM R34
2051 006410 236 RTN
2052 006411
2052 006411
2053 006411

```

* SIGN - OUTPUT + OR - OR BLANK OR NOTHING

TARNA 03/14/81=====

ITEM	LOC	OBJECT CODE	SRC=ORNMRT	OBJ=GEZNN3	9/11/1981	12:08 PM	PG 50
------	-----	-------------	------------	------------	-----------	----------	-------

=====

SCANN - SCAN FOR FORMAT ITEM

```

2 4 006411          *
2055 006411        * ENTRY
2056 006411        * R70 - AS DESCRIBED IN SCANN
2057 006411        * R32 - SIGN
2058 006411        * R30 - OUTPUT POINTER
2058 006411        *
2058 006411        *
2059 006411        *
*****

2060 006411 316 135 015 SIGNFF JSB =SIGNX      now do sign
2061 006414 124 271 316      LDMI R24,=PTR2+    last 2 bytes
2061 006417 377
2062 006420 165 006 345      PUMD R65,+R6      save register
2063 006423 261 221 210      LDMD R65,=TMPTR2    get FWA
2064 006426 230      BIN
2065 006427 321 314 377      CMMO R65,=PTR2      cif within bounds
2066 006432 372 012      JNC SIGNFX              jif not
2067 006434 125 310 060      CMB R25,=60          zero?
2068 006437 366 005      JNZ SIGNFX              jif no
2069 006441 124 025 242      STB R24,R25          sign to R25
2070 006444 250 060      LDB R24,=60            zero to R24
2071 006446 124 273 315 SIGNFX STMI R24,=PTR2-    restore it
2071 006451 377
2072 006452 165 006 343      POND R65,-R6
2073 006455 176 311 002      CMM R76,=2,0        cif 2 bytes left
2073 006460 000
2074 006461 372 022      JNC SIGNF1              jif no
2075 006463 124 250 040      LDB R24,=BLANK
2076 006466 125 270 317      LDBI R25,=PTR2++     next a blank?
2076 006471 377
2077 006472 024 300      CMB R25,R24
2078 006474 367 007      JZR SIGNF1              jif yes
2079 006476 270 316 377      LDBI R25,=PTR2+     pop sign/fill
2080 006501 124 273 315      STMI R24,=PTR2-     out with leading blank
2080 006504 377
2081 006505 231      SIGNF1 BCD
2082 006506 236      RTN
2083 006507 132 220      SIGNN TSB R32            SIGN
2084 006511 376 006      JRZ SIGN1              JIF POS
2085 006513 170 220      TSB R70                SIGN REQUESTED?
2086 006515 366 002      JNZ SIGN1              JIF YES
2087 006517 176 213      DCM R76                ELSE DECR DIGITS LEFT
2088 006521 173 220      SIGN1 TSB R73           E?
2089 006523 366 010      JNZ SIGNX              JIF YES
2090 006525 171 220      TSB R71                FILL
2091 006527 364 004      JNG SIGNX              JIF DEFAULT FILL
2092 006531 310 002      CMB R71,=2             BLANK FILL
2093 006533 372 031      JNC SGNEX              JIF YES
2094 006535 134 250 055 SIGNX LDB R34,=MINUS
2095 006540 132 220      TSB R32
2096 006542 377 016      JRN OUTSGN             NEG ,SO OUTPUT IT
2097 006544 134 250 053      LDB R34,=PLUS
2098 006547 230      BIN
2 1 006550 170 212      DCB R70

```

```

>> >>> SCANH - SCAN FOR FORMAT ITEM
2100 006552 231 BCD
2101 006553 364 011 JNG SGNEK NO SIGN SO EXIT
2102 006555 367 003 JZR OUTSGN S ,SO OUTPUT +
2103 006557 134 250 040 LDB R34,=BLANK
2104 006562 134 272 315 OUTSGN STBI R34,=PTR2- OUTPUT SIGN
2104 006565 377
2105 006566 236 SGNEK RTN
2106 006567
2106 006567
2107 006567
2108 006567
2109 006567
2110 006567
2111 006567
2111 006567
2111 006567
2112 006567
2113 006567 172 220 RADIX TSB R72
2114 006571 367 003 JZR RADEX JIF ZERPD
2115 006573 272 315 377 STBI R72,=PTR2- OUTPUT RADIX
2116 006576 236 RADEX RTN
2117 006577
2118 006577
2118 006577
2119 006577
2120 006577
2121 006577
2122 006577
2123 006577
2124 006577
2124 006577
2124 006577
2124 006577
2125 006577
2126 006577 171 250 004 RDIGIT LDB R71,=4 FILL IS ZEROS
2127 006602 176 074 241 LDM R76,R74 DIGITS RT
2128 006605 367 035 JZR RDEXIT JIF NONE
2129 006607 173 220 TSB R73
2130 006611 366 021 JNZ RDLPI JIF E
2131 006613 134 036 241 LDM R34,R36 EXP
2132 006616 374 014 JLZ RDLPI JIF EXP POS
2133 006620 217 NCM R34
2134 006621 076 301 CMM R34,R76
2135 006623 372 001 JNC RDLPI2 JIF EXP < RDIGIT
2136 006625 241 LDM R34,R76 MAX FILL
2137 006626 221 RDLPI2 TSM R#
2138 006627 367 003 JZR RDLPI JIF NO FILL
2139 006631 316 267 014 JSB =FILL
2140 006634 134 076 241 RDLPI LDM R34,R76
2141 006637 367 003 JZR RDEXIT JIF NONE LEFT
2142 006641 316 250 014 JSB =OUTDIG
2143 006644 236 RDEXIT RTN
2144 006645
2144 006645
  
```

```

>>>> SCANN - SCAN FOR FORMAT ITEM
2 3 006645          * EFORM - E FORMAT          <<<<<<<<
2146 006645          *
2147 006645          * ENTRY
2148 006645          *   PTR2 - OUTPUT BUFFER POINTER
2149 006645          *   R73 - E FORMAT
2150 006645          *   -1 A
2151 006645          *   0 NOT A OR E
2152 006645          *   1 E+-NNN
2153 006645          *   2 E SUPPRESS LEAD ZEROS,-,+
2154 006645          *   R36 - EXPONENT
2154 006645          *
2154 006645          *
2155 006645          *
2156 006645 102 222    EFORM   CLB R2
2157 006647 173 220    TSB R73
2158 006651 367 056    JZR EXPEX
2159 006653 122 250 105    LDB R22,=105
2160 006656 272 315 377    STBI R22,=PTR2-
2161 006661 250 373    LDB R22,=373
2162 006663 136 201    ELM R36
2163 006665 371 011    JEZ SGN+
2164 006667 215    TCM R36
2165 006670 122 250 375    LDB R22,=375
2166 006673 316 342 015    JSB =OUT2
2167 006676 360 003    JMP DONUM
2168 006700 316 332 015 SGN+   JSB =CND00
2169 006703 231    DONUM   BCD
2170 006704 132 250 003    LDB R32,=3C
2171 006707 136 201    DONLUP ELM R36
2172 006711 122 222    CLB R22
2173 006713 200    ELB R22
2174 006714 367 003    JZR OUTN
2175 006716 173 250 001    LDB R73,=1
2176 006721 316 332 015 OUTN   JSB =CND00
2177 006724 231    CIFDN   BCD
2178 006725 132 212    DCB R32
2179 006727 366 356    JNZ DONLUP
2180 006731 236    EXPEX   RTN
2181 006732 230    CND00   BIN
2182 006733 173 220    TSB R73
2183 006735 362 003    JOD OUT2
2184 006737 102 210    ICB R2
2185 006741 236    RTN
2186 006742 230    OUT2   BIN
2187 006743 122 312 060    ADB R22,=60
2188 006746 272 315 377    STBI R22,=PTR2-
2189 006751 236    RTN
2190 006752

```

JIF NONE
E
PLUS - 60
JIF EXP SGN POS
MAKE EXP POS
MINUS - 60
GO OUTPUT IT
NEXT EXP DIG
JIF ZERO
NO MORE SUP
JIF MORE
JIF NOT SUPPRESS
FOR LATER DECR

>>>>> RESTORE STATEMENT 02-22-77 <<<<<<<

```

2192 006752  *
2192 006752  *
2193 006752  *   RESTORE STATEMENT
2194 006752  *
2195 006752  *   DATA STATEMENT SEARCH - DASRC/DASRCC
2195 006752  *
2195 006752  *
2196 006752  *
    
```

DATTK, EQU 206

 RSTL# EQU 334 RESTORE LN # TOKEN

**** RESTORE LN # ATTRIBUTES *****

 OCT 227

```

2201 006752 227  *
2202 006753  *
2203 006753 230  RSTO, BIN
2204 006754 123 250 334 LDB R23,=334 RESTORE L# token
2205 006757 316 377 377 RSTCOM JSB =FETADD get dest address
2206 006762 165 323 006 ADMD R65,=FWCURR make it absolut
2207 006765 200
2207 006766 263 314 377 STMD R65,=PTR2 set to start of line
2208 006771 316 142 016 JSB =DASRC1 search for data
2209 006774 366 064 JNZ RSTERR jif no DATA token found
2210 006776  * JMP RRTN
2211 006776 165 261 314 RRTN LDMD R65,=PTR2 save address of data
2211 007001 377
2212 007002 052 243 STM R65,R52 copy it
2213 007004 261 006 200 LDMD R65,=FWCURR
2214 007007 315 015 000 SBM R65,=P.DTLG get DATA LINE offset
2215 007012 000 OCT 0
2216 007013 263 314 377 STMD R65,=PTR2 set pointer
2217 007016 241 LDM R65,R52 address of data
2218 007017 152 273 315 STMI R52,=PTR2- P.DTAG & P.DTLG (line#)
2218 007022 377
2219 007023 235 CLE
2220 007024 234 ICE set 'data found' flag
2221 007025 236 RSTOK RTN
2222 007026
    
```

**** RESTORE LABEL ATTRIBUTES *****

 OCT 227

```

2225 007027  *
2226 007027 230  RESTL, BIN
2227 007030 123 250 335 LDB R23,=335 RESTORE LABEL token
2228 007033 316 357 015 JSB =RSTCOM do common stuff
2229 007036 316 377 377 JSB =SKPNM+ skip the label
2230 007041 236 RTN
2231 007042
    
```

**** RESTORE ATTRIBUTES TABLE *****

 OCT 241 76024

```

2233 007042 241  *
2234 007043  *
2235 007043 230  RESTO, BIN
2236 007044 175 223 CLM R75 line# 0
2237 007046 316 377 377 RESTO1 JSB =FNDL2 find line (use PTR2)
2238 007051 175 271 316 LDMI R75,=PTR2+ move up to line start
    
```

```

KARMA 03/14/81=====
ITEM   LDD   OBJECT CODE   SPC=ORNMRT OBJ=GE2NM3   9/11/1981 12:08 PM PG 54
=====
> >>>> RESTORE STATEMENT 02-22-77 <<<<<<<<
2. 3 007054 377
2239 007055 316 117 016 JSB =DASRC+ find data
2240 007060 370 343 JEN RSTOK jif not found
2241 007062
2242 007062 316 377 377 RSTERR JSB =ERROR+
2243 007065 042 OCT 340 missing data stmt
2244 007066
2245 007066 230 DASRC BIN
2246 007067 165 261 006 LDMD R65,=FWCURRE
2246 007072 200
2247 007073 315 015 000 SBM R65,=P.DTLG offset for data line
2248 007076 000 OCT 0
2249 007077 263 314 377 STMD R65,=PTR2 make it point there
2250 007102 271 315 377 LDMD R65,=PTR2- get addr of data line
2251 007105 165 263 314 DASRC- STMD R65,=PTR2 point there
2251 007110 377
2252 007111 271 315 377 LDMD R65,=PTR2- trash line#
2253 007114 316 377 377 JSB =SKP2NA PTR2 --> nextline+1
2254 007117 316 142 016 DASRC+ JSB =DASRC1 search one line
2255 007122 367 252 JZR RRTH jif found
2256 007124 165 055 241 LDM R65,R55 copy to restore ptr2
2257 007127 231 BCD
2258 007130 175 311 231 CMM R75,=231,231,12 last line?
2258 007133 231 012
2259 007135 230 BIN
2260 007136 372 345 JNC DASRC- nope
2261 007140 235 CLE yes, so not found
2262 007141 236 RTN
2263 007142
2264 007142 155 261 314 DASRC1 LDMD R55,=PTR2 save addr of data line
2264 007145 377
2265 007146 175 271 315 LDMD R75,=PTR2- trash line#
2265 007151 377
2266 007152 165 250 206 LDB R65,=DATTK. token to find
2267 007155 316 133 006 JSB =FINDTK
2268 007160 236 RTN
2269 007161
2270 007161 *RTN LDMD R65,=PTR2 save address of data
2271 007161 * STM R65,R52 copy it
2272 007161 * LDMD R65,=FWCURRE
2273 007161 * SBM R65,=P.DTLG get DATA LINE offset
2274 007161 * OCT 0
2275 007161 * STMD R65,=PTR2 set pointer
2276 007161 * LDM R65,R52 address of data
2277 007161 * STMI R52,=PTR2- P.DTAG & P.DTLG (line#)
2278 007161 * CLE
2279 007161 * ICE set 'data found' flag
2280 007161 *STOK RTN
2281 007161

```

```

KARMA 03/14/81=====
ITEM LOC OBJECT CODE SRC=ORNMRT OBJ=GENNM3 9/11/1981 12:08 PM PG 55
=====
>>>>>> READ STATEMENT 11-01-80 <<<<<<
2283 007161 *****
2283 007161 *
2284 007161 * READ STATEMENT *
2284 007161 *
2284 007161 *****
2285 007161
2286 007161 ***** READ ATTRIBUTES TABLE *****
2287 007161 241 OCT 241
2288 007162 *****
2289 007162 140 223 READ, CLM R40 SET UP NON-TAPE INTERCEPT
2290 007164 263 321 203 STMD R40,=SCTEMP
2291 007167 236 RTH
2292 007170 ***** READN ATTRIBUTES TABLE *****
2293 007170 044 OCT 44
2294 007171 *****
2295 007171 120 222 READN, CLB R20
2296 007173 360 004 JMP READDT do common stuff
2297 007175
2298 007175 ***** READS ATTRIBUTES TABLE *****
2299 007175 044 OCT 44
2300 007176 *****
2301 007176 120 222 READS, CLB R20
2302 007200 212 DCB STRING FLAG
2303 007201
2304 007201 230 READDT BIN
2305 007202 165 261 006 LDMD R65,=FWCURR
2305 007205 200
2306 007206 315 020 000 SRM R65,=P.DTAG offset for DATA
2307 007211 000 OCT 0
2308 007212 263 314 377 STMD R65,=PTR2
2309 007215 271 315 377 LDMI R65,=PTR2-
2310 007220 366 007 JNZ CKEOL jif P.DATA set
2311 007222 175 223 CLM R75 line# 0
2312 007224 316 046 016 JSB =REST01 go set it
2313 007227 371 146 JEZ DATEXP jif no data
2314 007231 316 357 016 CKEOL JSB =NXDATA get next data
2315 007234 155 261 310 LDMD R55,=PTR1
2316 007237 377
2317 007240 006 345 PUMD R55,+R6 save pointer 1
2318 007242 261 314 377 LDMD R55,=PTR2
2319 007245 263 310 377 STMD R55,=PTR1 set it to data
2320 007250 165 310 004 CMB R65,=4 real?
2321 007253 366 014 JNZ TSTINT jif not real
2322 007255 316 377 377 JSB =ICONST
2323 007260 120 220 STR$? TSB R20
2324 007262 367 032 JZR STONUM jif read numeric
2325 007264 316 377 377 JSB =VAL$. convert to a string
2326 007267 360 020 JMP STR$1-
2327 007271 TSTINT DRP 165
2328 007273 310 032 CMB R65,=32 integer?
2329 007275 366 005 JNZ STR$ jif no
2330 007277 316 377 377 JSB =INTCON
2330 007300 360 356 JMP STR$?

```

```

>>>>> READ STATEMENT 11-01-80 <<<<<<<
2 007302 120 220 STR# TSB R20 want string?
2332 007304 267 043 JZR TYPERR jif invalid type
2333 007306 316 377 377 STR#1 JSB =SCONST
2334 007311 316 377 377 STR#1- JSB =STOST
2335 007314 260 003 JMP RDRTH
2336 007316 316 377 377 STONUM JSB =STOSV
2337 007321 155 261 006 RDRTH LDMD R55,=FWCURR
2337 007324 200
2338 007325 315 020 000 SBM R55,=P.DTAG offset for data
2339 007330 000 OCT 0
2340 007331 263 314 377 STMD R55,=PTR2
2341 007334 261 310 377 LDMD R55,=PTR1 get data pointer
2342 007337 273 315 377 STMI R55,=PTR2- store it
2343 007342 155 006 343 RDRTH1 POMD R55,-R5
2344 007345 263 310 377 STMD R55,=PTR1 restore pointer 1
2345 007350 236 HXRTH RTN
2346 007351
2347 007351 316 377 377 TYPERR JSB =ERROR
2348 007354 041 OCT 33D DATA TYPE MISMATCH
2349 007355 260 363 JMP RDRTH1
2350 007357 165 263 314 HXDATA STMD R65,=PTR2 set pointer to data
2350 007362 377
2351 007363 270 315 377 LDBI R65,=PTR2- load byte
2352 007366 310 016 CMB R65,=16
2353 007370 266 356 JNZ HXRTH jif not EOL
2354 007372 316 066 016 JSB =DASRC get next DATA stmt
2 007375 370 360 JEN HXDATA found one
2356 007377 316 377 377 DATEXP JSB =ERROR+
2357 007402 042 OCT 34D DATA EXPIRED
2358 007403

```

```

>>>>>>> INITIALIZE PROGRAM SUBROUTINE
2360 007403      *INIPGM  SAD
2361 007403      *        BIN
2362 007403      *        LDMD R45,=FWCURR
2363 007403      *        SBM  R45,=P.LEN      advance to P.LEN
2364 007403      *        OCT  0
2365 007403      *        STND R45,=PTR1      set PTR1
2366 007403      *        LDMI R45,=PTR1-     get program length
2367 007403      *        JNZ  INPRTH        jif non zero
2368 007403      *        LDMD R45,=FWCURR
2369 007403      *        SBMD R45,=BOVAR     calc length
2370 007403      *        STMI R45,=PTR1+     update length
2371 007403      *INPRTH  PAD
2372 007403      *GNXTL   RTN
2373 007403

```

>>>>> STRING CONCATENATION <<<<<<<<

```

2 5 007403      **** STR CONCATINATION ATTRIBUTES TABLE ****
2376 007403 007 053      OCT 7,53
2377 007405      ****
2378 007405 230      CONCA.  BIN
2379 007406 155 223      CLM  R55
2380 007410 173 012 343      POND R73,-R12      R73 = len2, R75 = addr2
2381 007413 143 343      POND R43,-R12      R43 = len1, R45 = addr1
2382 007415 173 345      PUND R73,+R12      push back str2
2383 007417 136 043 241      LDM  R36,R43      get length 1
2384 007422 073 303      ADM  R36,R73      add length 2
2385 007424 372 007      JNC  NOTRUN      jif less 65536
2386 007426 251 377 377      LDM  R36,=377,377      set to 65536
2387 007431 316 377 377      JSB  =WARN      warning OVF
2388 007434 070      OCT  56D
2389 007435 136 055 243 NOTRUN STM  R36,R55      copy byte amount
2390 007440 316 377 377      JSB  =RESMEM      reserve room
2391 007443 173 012 343      POND R73,-R12
2392 007446 370 056      JEN  MSRTN      jif overflow
2393 007450 136 345      PUND R36,+R12      new length to R12
2394 007452 043 305      SBM  R36,R43      len2 = total - len1
2395 007454 165 012 345      PUND R65,+R12      address of result$
2396 007457 263 314 377      STMD R65,=PTR2      set pointer
2397 007462 261 310 377      LDMD R65,=PTR1
2398 007465 006 345      PUND R65,+R6      save pointer
2399 007467 145 263 310      STMD R45,=PTR1      set to str1
2399 007472 377
2 007473 102 043 241      LDM  R2,R43      make copy of length 1
2401 007476 367 003      JZR  MS2      jif str1 null
2402 007500 316 127 017      JSB  =MSLOP      put str1 in result
2403 007503 136 221      MS2    TSM  R36      jif len2=0
2404 007505 367 011      JZR  BADLEN
2405 007507 002 243      STM  R36,R2
2406 007511 175 263 310      STMD R75,=PTR1      set PTR1 to str2
2406 007514 377
2407 007515 316 127 017      JSB  =MSLOP      put str2 in result
2408 007520 165 006 343 BADLEN POND R65,-R6
2409 007523 263 310 377      STMD R65,=PTR1      restore pointer 1
2409 007526 236      MSRTN  RTN
2409 007527
2412 007527 167 270 311 MSLOP  LDBI R67,=PTR1-      copy string
2412 007532 377
2413 007533 272 315 377      STBI R67,=PTR2-
2414 007536 102 213      DCM  R2
2415 007540 266 365      JNZ  MSLOP      jif not done
2416 007542 236      CHNRT- RTN
2417 007543      *      HED RESULT
2418 007543      *RESUL. JSB  =NO.CRS      CLEAR CRS TO LEFT OF RESUL
2419 007543      *      LDMD R40,=RESULT
2420 007543      *      PUND R40,+R12
2421 007543      *      JSB  =ONER      must be real
2422 007543      *      CLM  R54
2423 007543      *      LDB  R54,=26      max
2 007543      *      LDM  R30,=XLPEH      use temporary

```

```

>>>> STRING CONCATENATION <<<<<<
2425 007543      *      JSB  =CVNUM      go convert
2426 007543      *      LDM  R36,R54     byte count
2427 007543      *      JSB  =HLFLIN     out to crt
2428 007543      *      CLE
2429 007543      *CHNRT-  RTN

```

```

KARNA 03/14/81=====
ITEM    LOC    OBJECT CODE  SRC=ORNAME?  USE=GENHMS  9/11/1981 12:08 PM  PG 60
=====
>>>>  CHAIN  <<<<<<<
2 1 007543  *** CHAIN ATTRIBUTE TABLE *****
2432 007543  * OCT 241
2433 007543  *****
2434 007543
2435 007543  CHAIN. JSB =SETTR1
2436 007543  CHAIN. BSS 0 AUGUST 7,1980
2437 007543  * HERE FOR TRANSLATION ONLY
2438 007543
2439 007543  * WE MAY HAVE TO REPORT CHANGE IN FWCURR
2440 007543  * GETNAM DOES BIN
2441 007543  * JSB =GETNAM
2442 007543  * LDMD R56,=FWPRGM
2443 007543  * STMD R56,=FWCURR
2444 007543  * JSB =CALTAP GO LOAD IT
2445 007543  * CHAIN NOW DONE BY MASS STORAGE ROM
2446 007543  * CHAIN+ WAS SHARED BY MASS STORAGE AND SYSTEM
2447 007543  * AND NOW NEEDED ONLY BY MASS STORAGE.
2448 007543  CHAIN+ BSS 0 Post-processing(cf MS rom)
2449 007543 117 310 300 CMB R17,=300 ERRORS?
2450 007546 373 372 JCY CHNRT- EXIT IF ERRORS
2451 007550 316 377 377 JSB =RSETG- RESET GO
2452 007553  *****
2452 007553  *
2453 007553  * R40-47 CLEARED BY RSETG-,DRP=R45
2453 007553  *
2453 007553  *****
2 007553 263 343 212 STMD R#,=RMEM RESET RES MEM COUNT
2455 007556 262 047 200 STBD R#,=CONTOK MAKE CONT LEGAL
2456 007561 262 104 202 STBD R#,=DALLED CLEAR DALL FLAG 5-7-81
2457 007564 145 261 041 LDMD R45,=LWAMEM
2457 007567 200
2458 007570 263 036 200 STMD R#,=NXTRTN RESET NEXT RETURN PTR
2459 007573 155 251 034 LDM R55,=34,1,0 NEEDED BY ASSIGNL
2459 007576 001 000
2460 007600 316 377 377 JSB =ASIGNL FIND ASSIGN BUFFER PTR
2461 007603 170 220 TSB R70 ARE THERE ANY ASSIGN BUFF
2462 007605 367 011 JZR ARO JIF NO
2463 007607 165 055 303 ADM R65,R55 PASS OVER ASSIGN BUFFERS
2464 007612 263 025 200 STMD R65,=LAVAIL TO LAVAIL
2465 007615 263 030 200 STMD R65,=CALVRB AND CALVRB
2466 007620 ARO BSS 0 SET OPTBAS
2467 007620 166 222 CLB R66 RESET OPTB FLAG
2468 007622 262 346 212 STBD R66,=OPTB
2469 007625 316 377 377 JSB =ALINI+ FOR ALLO ON THE FLY
2470 007630 116 250 002 LDB R16,=2 SET TO RUN MODE
2471 007633 145 250 001 LDB R45,=1
2472 007636 262 304 377 STBD R45,=RULITE SET RUN LIGHT
2473 007641 261 006 200 LDMD R45,=FWCURR GET FWCURR ADDR
2474 007644 055 243 STM R45,R55 COPY
2475 007646 315 040 000 SBM R45,=40,0,0 SUBTRACT P.GOG.
2475 007651 000
2476 007652 263 257 203 STMD R45,=OLDPCR FOR TRACE
2 007655 263 115 202 STMD R45,=PCR PCR

```

```

2 8 >>> CHAIN
2 8 007660 241 LDM R45,R55
2479 007661 315 006 000 EBM R45,=6,0,0 GET TYPE
2479 007664 000
2480 007665 263 314 377 STMD R45,=PTR2
2481 007670 270 315 377 LDBI R#,=PTR2-
2482 007673 362 015 JCD ILCHER JIF ERROR
2483 007675 MAIN1 BSS 0
2484 007675 *AIN1 JSB =INIVAR GO INIT VARIABLES
2485 007675
2486 007675 * CHANGED STRATEGY TO FLAG TRACE ON THE FLY
2487 007675 * LDBD R22,=TRCFL2 TRACE ALL FLAG
2488 007675 * JSB =TRVR6L GO SET TRACE FLAGS
2489 007675
2490 007675 136 250 007 LDB R36,=7
2491 007700 262 065 210 STBD R36,=ROMFL TELL ROMS WHATS HAPPENING
2492 007703 316 377 377 JSB =ROMINI
2493 007706 316 377 377 JSB =SET240
2494 007711 236 RTN
2495 007712 316 377 377 ILCHER JSB =ERROR+
2 8 007715 037 OCT 310 ILLEGAL CHAIN
2497 007716 FIN
  
```

```

KARMA 03/14/81=====
SYMBOL  VALUE  TYPE  COUNT  Symbol Table  9/11/1981 12:03 PM PG 62
=====
AD  3  005741  LCL  0  <--- NOT REFERENCED??
A  4  177777  EXT  2
A  1  000723  LCL  3
ADDR1A 000753  LCL  1
ADDR1B 000765  LCL  1
ADDRR  000667  LCL  1
ADDR  000677  LCL  1
ALINE# 104237  G DAD  1
ALINI+ 177777  EXT  1
ARDON  003724  LCL  3
ARDON+ 003720  LCL  1
ARITH  003600  ENT  3
ARO  007620  LCL  1
ASIGNL 177777  EXT  1
ATTOK  000100  EQU  0  <--- NOT REFERENCED??
BADLEN 007520  LCL  1
BLANK  000040  G EQU 11
BOVAR  100014  G DAD  1
BUFF  002653  LCL  2
CALC AD 002062  LCL  1
C  FOR 103372  G DAD  2
CALSCH 003371  LCL  2
CALVRB 100030  G DAD  1
CHAIN+ 007543  ENT  1
CHAIN. 007543  ENT  1
CHNRT- 007542  LCL  1
CHRAD  004245  LCL  3
CHRAD- 004241  LCL  0  <--- NOT REFERENCED??
CI  DX 004250  LCL  0  <--- NOT REFERENCED??
CHKRT  004016  LCL  2
CHRTX  004040  LCL  2
CIFDN  006724  LCL  0  <--- NOT REFERENCED??
CKCLO  004461  LCL  1
CKCLOR 004505  LCL  3
CKDON  000331  LCL  1
CKDONE 000771  LCL  1
CKEOL  007231  LCL  1
CKINC  005623  LCL  3
CKI COP 000623  ENT  2
Q  OT  005642  LCL  1
CKRTN  005626  LCL  1
CKSEP  005601  LCL  3
CLCFC  000254  LCL  1
CLEAN  002757  LCL  2
CLOSE  000051  G EQU  2
CLSFND 004511  LCL  1
CHDOU  006732  LCL  2
COMMA  000054  G EQU  4
COMMA$ 003165  ENT  1
COMMA. 003304  ENT  1
COMPR  005724  LCL  1
COMRTN 003266  LCL  2
CONCA. 007405  ENT  1
CONINT 177777  EXT  1

```

```

KARMA 03/14/81=====
SYMBOL  VALUE    TYPE    COUNT    Symbol Table    9/11/1981 12:08 PM  PG 63
=====
COOK      100047    G DAD      1
CL SUB    177777    EXT        1
CR        000015    G EQU      2
CREP1     004314    LCL        2
CS.C.     100212    G DAD      1
CTABL     005345    LCL        1
CVNUM     003420    ENT        1
CVNUMG    003450    ENT        2
DALLED    101104    G DAD      1
DASRC     007066    LCL        1
DASRC+    007117    LCL        1
DASRC-    007105    LCL        1
DASRC1    007142    LCL        2
DATEXP    007377    LCL        1
DATTK     000206    EQU        1
DCFILL    004227    LCL        3
DCR45     002241    LCL        1
DCSLOP    177777    EXT        1
DCSTR1    177777    EXT        1
DECT      004266    LCL        1
DECT      177777    EXT        1
DISBUF    100542    G DAD      1
DISP      002330    ENT        1
DISP.+    002334    ENT        1
DISPLN    101136    G DAD      1
DISPTR    100060    G DAD      1
DOCOM     002367    LCL        1
DONE?     004517    LCL        2
DONLUP    006707    LCL        1
DONUM     006703    LCL        1
DRV12     177777    EXT        1
DSFLAG    101121    G DAD      1
DUFMT     003225    LCL        1
DUFMT1    003255    LCL        1
EEE        004412    LCL        1
EFORM     006645    LCL        1
EMOVDN    177777    EXT        1
EMOVUP    177777    EXT        2
ENDSGM    000617    LCL        1
EXOK      000016    EQU        0    <--- NOT REFERENCED??
ERGOTO    100111    G DAD      2
ERR1      000221    LCL        1
ERR2      000225    LCL        1
ERROR     177777    EXT        3
ERROR+    177777    EXT        12
ERTOK     000360    EQU        0    <--- NOT REFERENCED??
EX        004732    LCL        1
EXNXT     001275    ENT        4
EXPEX     006731    LCL        1
FCHAR     005371    LCL        1
FCHAR+    005361    LCL        1
FCRT1     005471    LCL        2
FCRTN     005473    LCL        7
FCRTR     005551    LCL        1

```

KARMA 03/14/81=====										=====									
SYMBOL	VALUE	TYPE	COUNT	Symbol Table		02/11/1981 12:00 PM PG 64													
=====										=====									
FETADD	177777	EXT	2																
FETVA	177777	EXT	0	<--- NOT REFERENCED??															
F L	006267	LCL	3																
FILLIT	006111	LCL	1																
FINDTK	003133	LCL	2																
FLOOP	000933	LCL	1																
FENTER1	003277	LCL	3																
FENTER2	004264	LCL	4																
FENTER4	005120	LCL	3																
FMTERR	004455	LCL	11																
FND1	005526	LCL	1																
FNDL2	177777	EXT	1																
FNDVLP	001202	LCL	1																
FNUM	177777	EXT	2																
FOR,	000001	ENT	1																
FOR1	000132	LCL	1																
FORMDC	003426	LCL	1																
FORMG	003456	ENT	2																
FORMG+	003463	LCL	0	<--- NOT REFERENCED??															
FORMH	003479	ENT	2																
FOR1+	003440	ENT	1																
FORMNC	003504	LCL	1																
FORMOV	003503	LCL	1																
FRMN1	003534	LCL	1																
FTADR	177777	EXT	4																
FWBIN	100044	G DAD	1																
FWCURR	100006	G DAD	11																
GCHAR#	005552	LCL	6																
G R-	005565	ENT	4																
GKRN	005600	LCL	2																
GETFC	000231	LCL	5																
GETFEX	000266	LCL	2																
GETREP	005474	LCL	2																
GLBL1?	002226	LCL	1																
GLBL2?	002275	LCL	1																
GLOP	005476	LCL	1																
GLOPL	005514	LCL	1																
GLOPT	005521	LCL	1																
GOLBL	000262	EQU	0	<--- NOT REFERENCED??															
GOLBL	000279	EQU	0	<--- NOT REFERENCED??															
G TOK	000133	EQU	0	<--- NOT REFERENCED??															
GOTOK	000132	EQU	0	<--- NOT REFERENCED??															
GSNVLP	000505	LCL	2																
GTFOR	000606	LCL	2																
HLFLIN	177777	EXT	0	<--- NOT REFERENCED??															
ICONST	177777	EXT	1																
IDCHAR	005530	LCL	1																
IDCHR1	005533	LCL	1																
ILCHER	007712	LCL	1																
INCADR	101779	G DAD	3																
IMCLEN	101766	G DAD	4																
IMERR	103724	G DAD	4																
IMERR?	004444	LCL	2																
INSADR	101763	G DAD	3																

```

KARMA 03/14/81=====
SYMBOL  VALUE    TYPE    COUNT    Symbol Table    9/11/1981 12:08 PM PG 65
=====
INCL EN  101761  G DAD    2
I  OK    000216      EQU    1
INWADR  101773  G DAD    2
INC1     004343      LCL    1
INCS     003709      LCL    1
INC3?    003674      LCL    1
INDTOK   000367      EQU    1
INFR4    177777      EXT    1
INT?     003652      LCL    1
INTCON   177777      EXT    1
INTMUL   177777      EXT    1
INVAL    002323      LCL    3
KAA      004430      LCL    1
KCLR     002034      ENT    2
KCLR-    001653      ENT    2
KCLRRLP  002101      LCL    1
KEY2     001736      LCL    3
KEYERR   001600      LCL    4
KEYTAB   102016  G DAD    2
KFMT     003331      LCL    0    <--- NOT REFERENCED??
KFMT1    003343      LCL    1
KSETLP   001672      LCL    1
KTRIPL   001743      LCL    1
LAVAIL   100025  G DAD    9
LBLTK?   003146      LCL    1
LBLTOK   000257      EQU    1
LD1      006053      LCL    1
LD23     006401      LCL    1
LDEX1    006240      LCL    1
LDEXIT   006231      LCL    4
LDIGIT   006044      LCL    1
LEGCAL   101403  G DAD    1
LEGDU-   001706      LCL    1
LEGDUN   001712      ENT    2
LEGEND   101145  G DAD    2
LENKLB   000036      EQU    1
LINELN   101714  G DAD    2
LJTAB    004716      LCL    1
LTABL    004552      LCL    1
LTHUT    005115      LCL    1
LURAC    005054      LCL    1
LV-PT2   001341      LCL    2
LWAMEN   100041  G DAD    1
MAIN1    007675      LCL    0    <--- NOT REFERENCED??
MATCH?   177777      EXT    1
MATH2    001141      LCL    2
MINUS    000055  G EQU    1
MLTLOP   002064      LCL    1
MOVEDN   002542      LCL    1
MS2      007503      LCL    1
MSLOP    007527      LCL    3
MSRTN    007526      LCL    1
MVAGN    004002      LCL    1
MVCKD    002517      LCL    0    <--- NOT REFERENCED??

```

CARNA 03/14/81

SYMBOL	VALUE	TYPE	COUNT	Symbol Table	9/11/1981 12:08 PM PG 66
AVIUP	002504	LCL	1		
AVI	001155	ENT	1		
AVI	001040	LCL	3		
NEXTEX	001274	LCL	1		
NGEXP	003773	LCL	1		
NO.CRS	177777	EXT	0	←--- NOT REFERENCED??	
NODEC	003710	LCL	1		
NOINCS	003702	LCL	1		
NORM	006066	LCL	2		
NOTE	006101	LCL	1		
NOTOVR	002465	LCL	3		
NOTRUN	007435	LCL	1		
NR12	003647	LCL	1		
NR3	003756	LCL	2		
NRE	005707	LCL	1		
NSHFT	006063	LCL	2		
HTPRL	002427	LCL	2		
NTSTD	005267	LCL	1		
NTSTL	005307	LCL	1		
NXDATA	007357	LCL	2		
NX R	001335	LCL	4		
NX LOOP	002551	LCL	1		
NXLOP1	002563	LCL	1		
NXRTN	007350	LCL	1		
NXT	001402	LCL	2		
NXTLIN	177777	EXT	1		
NXTMEM	100022	G DAD	2		
NXTRTN	100036	G DAD	1		
NXT	001330	LCL	1		
NXITK.	000217	EQU	1		
NXZRTN	002656	LCL	2		
OFALL	001777	LCL	1		
OFALL+	002003	LCL	1		
OFFER.	002131	ENT	1		
OFKEY.	001753	ENT	1		
OFONE	002023	LCL	1		
OFONE+	002026	LCL	1		
OLDPCR	101657	G DAD	1		
ON.	002140	ENT	1		
ON	177777	EXT	2		
ON	177777	EXT	1		
ONER	177777	EXT	1		
ONERR.	002111	ENT	1		
ONFLAG	100065	G DAD	1		
ONK1	001624	LCL	1		
ONKEY.	001447	ENT	1		
ONLP1	002205	LCL	4		
ONLP2	002260	LCL	2		
OPEN	000950	G EQU	1		
OPTB	105346	G DAD	1		
OUT1	006333	LCL	3		
OUT1C	006325	LCL	2		
OUT1X	006371	LCL	2		
OUT1X-	006363	LCL	2		

```

=====
XARMA 03/14/81=====
SYMBOL  VALUE      TYPE      COUNT      Symbol Table      9/11/1981 12:08 PM PG 67
=====
OUT      006742      LCL      2
O/P      005753      LCL      1
OUTCEX   006043      LCL      1
OUTCLP   005772      LCL      1
OUTOIG   006250      LCL      4
OUTE     006175      LCL      1
OUTMO    006211      LCL      1
OUTN     006721      LCL      1
OUTSGN   006562      LCL      2
OVFLO    006241      LCL      2
P#RTN    003271      LCL      4
P.BUFF   101706      G DAD     7
P.DTAG   000020      G EQU     2
P.DTLG   000015      G EQU     2
P.FCNG   000026      G EQU     1
P.FLAG   101712      G DAD     5
P.PTR    101710      G DAD     8
PCR      101115      G DAD     5
PEOL     002616      ENT      2
PERIOD   000056      G EQU     3
P.C      000262      LCL      1
PLUS     000053      G EQU     1
FOPR40   005341      LCL      1
POS.5    005733      LCL      1
POS.5L   005735      LCL      1
PRFEX    002611      LCL      3
PRFLAG   101120      G DAD     1
PRIN.+   002355      ENT      1
PRINT.   002351      ENT      1
PRLIN1   002661      LCL      0      <--- NOT REFERENCED??
PRLIN2   002736      ENT      2
PRLIN3   002725      LCL      1
PRLINE   002660      ENT      1
PRLN2-   002730      LCL      1
PRNFMT   002407      ENT      1
PRNT##   177777      EXT      0      <--- NOT REFERENCED??
PRNT#N   177777      EXT      0      <--- NOT REFERENCED??
PRNTLN   101137      G DAD     1
PROCHR   004677      LCL      1
P.UF     107454      G DAD     1
P.PTR    100062      G DAD     1
PS.C.    100222      G DAD     1
PTR1     177710      G DAD    39
PTR1+    177712      G DAD     4
PTR1-    177711      G DAD    11
PTR1-+   177713      G DAD     2
PTR2     177714      G DAD    44
PTR2+    177716      G DAD     4
PTR2-    177715      G DAD    50
PTR2-+   177717      G DAD     8
PUFILL   006145      LCL      1
PUIT     006006      LCL      1
QLCOM    005001      LCL      1
QLIT     004772      LCL      1

```

```

KARMA 03/14/81=====
SYMBOL  VALUE      TYPE      COUNT      Symbol Table      9/11/1981 12:03 PM PG 63
=====
RNDM      005015      LCL        1
DTYPE     000042      G EQU      2
R_EX      006576      LCL        1
RADIX     006567      LCL        1
RADX      004306      LCL        1
RADXR     004303      LCL        1
RDEXIT    006644      LCL        2
RDIGIT    006577      LCL        1
RDLP1     006634      LCL        3
RDLP2     006626      LCL        1
RDRTH     007321      LCL        1
RDRTH1    007342      LCL        1
READ#     177777      EXT        0      <--- NOT REFERENCED??
READ#N    177777      EXT        0      <--- NOT REFERENCED??
READ      007162      ENT        1
READDT    007201      LCL        1
READN     007171      ENT        1
READS     007176      ENT        1
RELMEM    177777      EXT        3
RESEIL    177777      EXT        4
RESLI     177777      EXT        1
RMEM      177777      EXT        1
RESTL     007027      ENT        1
RESTO     007043      ENT        1
RESTO1    007046      LCL        1
RESTPC    000567      LCL        1
RM+SET    177777      EXT        4
RMEM      105343      G DAD      1
R_EX      005754      LCL        4
ROI       001153      LCL        4
ROMFL     104065      G DAD      1
ROMINI    177777      EXT        1
ROUND     005667      ENT        2
RRTN      006776      LCL        1
RSET      005156      LCL        1
RSETG-    177777      EXT        1
RSPTRS    177777      EXT        3
RSTCOM    006757      LCL        1
RSTERR    007062      LCL        1
RST      000334      EQU        0      <--- NOT REFERENCED??
R_O..     006753      ENT        1
RSTOK     007025      LCL        1
RTBRAC    005124      LCL        1
RTBREX    005147      LCL        1
RTBRX1    005152      LCL        1
RUKEYS    001610      LCL        1
RULITE    177704      G DAD      1
SAVER6    104066      G DAD      2
SC12?     005327      LCL        1
SC12RT    005344      LCL        2
SCANL     004557      LCL        1
SCANL1    004104      LCL        1
SCANLP    004101      LCL        1
SCANLR    004604      LCL        2

```

```

CARNA 03/14/81=====
SYMBOL  VALUE  TYPE  CCOUNT  Symbol Table  9/11/1981 12:03 PM PG 69
=====
SCOLS  004627  LCL  1
S ALX  004572  ENT  2
SCANH  004044  LCL  1
SCANUS 004062  LCL  1
SCLRT+ 004644  LCL  1
SCLRTX 004656  LCL  2
SCNLGO 004577  LCL  1
SCONST 177777  EXT  1
SCTEMP 101721  G DAD 2
SEMIC# 003174  ENT  1
SEMIC. 003313  ENT  1
SEND0C1 005235  LCL  1
SEND0C2 005215  LCL  1
SEND0C 005202  LCL  0 <--- NOT REFERENCED??
SEND0CL 005257  LCL  1
SEND0CR 005177  ENT  2
SEP 004356  LCL  1
SEP1 004374  LCL  2
SEP10 177777  EXT  1
SEP 004361  LCL  1
SEP 004347  LCL  1
SE1240 177777  EXT  2
SETCNT 003661  LCL  2
SETFIL 006150  LCL  2
SETFL1 006160  LCL  1
SETONF 002311  LCL  1
SETTR1 177777  EXT  2
SFILL 004233  LCL  2
SFILL1 004235  LCL  3
SGN+ 006700  LCL  1
SGNEX 006566  LCL  2
SHFAGN 004026  LCL  1
SIGN1 006521  LCL  2
SIGNF 004324  LCL  2
SIGNF1 006505  LCL  2
SIGNFF 006411  LCL  1
SIGNFX 006446  LCL  2
SIGNN 006507  LCL  1
SIGNX 006535  LCL  3
SKCNA 177777  EXT  4
SK F- 000411  LCL  1
SKPLB 001727  LCL  2
SKPLOP 000421  LCL  4
SKPNM+ 177777  EXT  3
SKPNM1 177777  EXT  1
SKYTXT 106610  G DAD 1
SLASH 000057  G EQU 1
SPECOT 004760  LCL  3
SPRTN 004411  LCL  0 <--- NOT REFERENCED??
SPSEP 005653  LCL  0 <--- NOT REFERENCED??
SRCHMO 001412  LCL  1
SRCHTK 001354  ENT  2
STAR 000052  G EQU 2
STAR1 004254  LCL  1

```

```

KARMA 03/14/81=====
SYMBOL  VALUE  TYPE  COUNT  Symbol Table  9/11/1981 12:03 PM PG 70
=====
STR      004251  LCL    1
SP      000314  ENT    1
SUM      007316  LCL    1
STOR0    006403  LCL    1
STOST    177777  EXT    1
STOSV    177777  EXT    2
STPRT+   000602  LCL    1
STPRTH   000605  LCL    0  <--- NOT REFERENCED??
STR      003177  LCL    1
STR$     007302  LCL    1
STR$1    007306  LCL    0  <--- NOT REFERENCED??
STR$1-   007311  LCL    1
STR$?    007260  LCL    1
STRNG    004541  LCL    1
SUBR     001006  LCL    1
SUBROI   177777  EXT    1
SUBRR1   000776  LCL    0  <--- NOT REFERENCED??
SVPTRS   177777  EXT    3
SZOK     001667  LCL    1
TMOTR2   104221  G DAD   1
T        000279  ENT    1
TOKS5?   003161  LCL    1
TOS      101744  G DAD   3
TRAFFIC  177777  EXT    3
TRYRBL   177777  EXT    0  <--- NOT REFERENCED??
TSEXP    003630  LCL    1
TSRTN    001140  LCL    2
TSTEND   177777  EXT    1
T      LCL    002642  LCL    4
TSINT    007271  LCL    1
TSTNXT   000473  LCL    1
TSTR1    001106  LCL    2
TSTR2    001125  LCL    2
TSTRP    005444  LCL    2
TSTRP1   005451  LCL    1
TSTUS    003316  LCL    1
TSTZ     006312  LCL    3
TYPERR   007351  LCL    1
TYTAB    004126  LCL    1
UP       003345  LCL    1
UPIN#    003065  ENT    1
UPCALF   001324  LCL    1
UPER     001327  LCL    1
UPFOR    001315  LCL    1
USCOM    003071  LCL    1
USEXIT   003032  LCL    1
USING.   003026  ENT    1
USING?   100064  G DAD  10
USINGL   000345  EQU    1
USINL.   003051  ENT    1
VAL$     177777  EXT    1
WARN     177777  EXT    1
WRTLIN   002767  LCL    2
WTFST    004657  LCL    3

```

KARMA 03/14/81=====

SYMBOL	VALUE	TYPE	CCOUNT	Symbol Table	9/11/1981 12:08 PM PG 71
REF	004162	LCL	1		
ST	002534	LCL	1		
ZONELN	000025	EQU	2		
ZRO	000000	G LAD	2		

TABLE OF CONTENTS - RUNTIME STUFF		1
ENTRY POINTS AND EXTERNALS		2
LINE "FOR" ROUTINE GEMINI 9-80		5
RUNTIME TO/STEP Gemini 9-80		8
RUNTIME "NEXT" 12-15-76		14
TOKEN SEARCH ROUTINE		16
ON KEY# OFF KEY# Gemini 9-80		17
ON GOTO/GOSUB Gemini 9-80		21
UNFORMATTED PRINT 02-17-77		23
PRINT LINE		24
USING		27
FORM#CVNUM		31
SCANN - SCAN FOR FORMAT ITEM		35
RESTORE STATEMENT 02-22-77		53
READ STATEMENT 11-01-80		55
INITIALIZE PROGRAM SUBROUTINE		57
STRING CONCATENATION		58
CHAIN		60

REPORT HAD 0 ERRORS 0 WARNINGS 490 LABELS LAST ERROR AT 0