

>>>>>> TABLE OF CONTENTS <<<<<<<<

000000	* 1	Table of Contents
3 000000	* 2	Function call runtime
4 000000	* 5	INPUT runtime routines
5 000000	* 6	INPUT in graphics
6 000000	* 7	Runtime INPUT string
7 000000	* 8	Runtime INPUT numeric
8 000000	* 9	INPUT error recovery

9 000000		
10 000000	*ENTRIES	
11 000000	ENT FNCAL.	
12 000000	ENT FNCAL\$	
13 000000	ENT FNLET.	
14 000000	ENT FNRTN.	
15 000000	ENT INPUH.	
16 000000	ENT INPUT.	
17 000000	ENT INPU\$.	
18 000000	ENT FNRET.	
19 000000		
20 000000	EXT GRAFIN	
21 000000	EXT FETSYA	
22 000000	EXT FTAPU	
23 000000	EXT FTSTL1	
24 000000	EXT FTSTL2	
25 000000	EXT ONER+	
26 000000	EXT SETALO	
27 000000	EXT SKFNM+	
28 000000	EXT SRCHTK	
29 000000	EXT STOSV	
30 000000	EXT STOST	
31 000000	EXT ROMJSB	
32 000000		

WHY FNRET. AND FNRTN.????

```
ITEM      LOC      OBJECT CODE  SRC=CCGFZZ OBJ=GE%GFZ   9/11/1981 10:47 AM  PG  2
```

◀◀◀◀◀◀◀◀

				INPTOK EQU 137		
				***	TOKEN 26 & 27 ATTRIBUTES TABLE	*****
				OCT 6		

35	000000			FNCAL\$	BSS 0	SAME AS STRING
36	000000	006		LDB	R23,=203	STR FUNC
37	000001			JMP	FNCAL0	
38	000001			OCT	6	ATTRIBUTE
39	000001	123	250 203	FNCAL.	BSS 0	
40	000004	360	004	LDB	R23,=201	NUM FUNC
41	000006			FNCAL0	BSS 0	
42	000006	006		BIN		
43	000007			LDMI	R75,=PTR1-	GET FUNC ADDR
44	000007					
45	000007	123	250 201	JNZ	FNCAL2	JIF ALLOCATED
46	000012			STBD	R#,=GINTDS	DISABLE INTERRUPTS
47	000012	230		JSB	=CRDF?	AND ALLOCATE IT
48	000013	175	271 311	STBD	R#,=GINTEN	ENABLE INTERRUPTS
48	000016	377		FNCAL-	BSS 0	
49	000017	366	023	CMB	R17,=300	ANY ERROR ?
50	000021	262	001 377	JNC	FNCAL1	JIF NO
51	000024	316	377 377	RTN		ELSE RETURN
51	000027	262	000 377	FNCAL1	BSS 0	
53	000032			LDMI	R75,=PTR1	GET REL ADDR
54	000032	117	310 300			
55	000035	372	001	FNCAL2	BSS 0	
56	000037	236		DRP	!75	
57	000040			ADMD	R75,=FWCURR	MAKE IT ABS
58	000040	175	271 310	SBM	R75,=4,0,0	PT TO FUNC DEF POINTER
59	000043	377				
60	000044			STMD	R75,=PTR2	
61	000044	323	006 200	LDMI	R75,=PTR2-	GET REL PTR TO DEF FN
62	000047	315	004 000	JNZ	FNCAL3	JIF FN DEFINED
62	000052	000		LDMD	R75,=PTR1	SAVE POINTER
63	000053	263	314 377	PUMD	R75,+R6	
64	000056	271	315 377	STMD	R75,=PTR2	
65	000061	366	072	JSB	=BOP	POS1 TO BEGIN OF PRGM
66	000063	261	310 377	CLB	R26	
67	000066	006	345	LDB	R27,=207	LOOKING FOR DEF FN
68	000070	263	314 377	JSB	=ROMJSB	SEARCH FOR DEF FN
69	000073	316	377 377	DEF	SRCHTK	
70	000076	126	222	OCT	0	
71	000100	127	250 207	JEZ	ERRFH	JIF NOT FOUND
72	000103	316	377 377	LDMD	R75,=PCR	SAVE PCR
73	000106	377	377			
74	000110	000		PUMD	R75,+R6	
75	000111	371	033	LDMD	R75,=PTR1	
76	000113	175	261 115	JSB	=FNDPC	LOCATE PCR OF DEF FN
76	000116	202		JSB	=DEFFN.	ALLOC FOR DEF FN
77	000117	006	345	PUMD	R75,-R6	RESTORE PCR
78	000121	261	310 377	STMD	R75,=PCR	
79	000124	316	225 001			
80	000127	316	377 377			
81	000132	175	006 343			
81	000135	263	115 202			

>>>> FUNCTION CALL RUNTIME				<<<<<<<<					
83	000140	343		POMD	R75,-R6		RESTORE POINTER		
84	000141	263	310 377	STMD	R75,=PTR1				
85	000144			*****	CMB	R17,=300	ALLOC ERR ?		
86	000144			*****	JNC	FNCAL1	JIF NOT		
87	000144			*****	RTN		ELSE RTN		
88	000144	260	264	JMP	FNCAL-		CHECK FOR ERROR		
89	000146								
90	000146			ERRFN	BSS	0			
91	000146	175	006 343	POMD	R75,-R6		THROW AWAY 3 BYTES		
92	000151			ERRFN1	BSS	0			
93	000151	316	377 377	JSB	=ERROR+				
94	000154	047		OCT	390		FN MISSING		
95	000155								
96	000155			FNCAL3	BSS	0			
97	000155	165	271 315	LDMI	R65,=PTR2-		GET RETURN ADDR		
97	000160	377							
98	000161	367	004	JZR	CALOK		JIF NOT RECURSIVE CALL		
99	000163	316	377 377	JSB	=ERROR+		RECUR CALL ERROR		
100	000166	052		OCT	420				
101	000167								
102	000167			CALOK	BSS	0			
103	000167	110	251 150	LDM	R10,=XLPEN		UPPER LIMIT OF STACK		
103	000172	215							
104	000173	012	305	SBM	R10,R12		AVILABLE STACK SPACE		
105	000175	311	000 001	CMM	R10,=0,1		REQUIRE 256 BYTES		
106	000200	265	004	JPS	STAKOK		JIF ENOUGH SPACE		
107	000202	316	377 377	JSB	=ERROR+		ELSE STACK OVF		
108	000205	032		OCT	260				
109	000206								
110	000206			STAKOK	BSS	0			
111	000206	165	261 314	LDMD	R65,=PTR2		SAVE POINTER		
111	000211	377							
112	000212	006	345	PUMD	R65,+R6				
113	000214	175	323 006	ADMD	R75,=FWCURR		ABS FUN DEF ADR		
113	000217	200							
114	000220	263	314 377	STMD	R75,=PTR2				
115	000223	316	377 377	JSB	=3KPNM+		SKP FUN NAM IN CALL		
116	000226	170	270 315	LDBI	R70,=PTR2-		# OF PAR IN DEF		
117	000231	377							
117	000232	206		LRB	R70				
118	000233	134	270 311	LDBI	R34,=PTR1-		# OF PAR IN CALL		
118	000236	377							
119	000237	070	300	CMB	R34,R70		EQUAL ?		
120	000241	266	111	JNZ	BADP		JIF NOT		
121	000243	316	117 001	JSB	=3KPNZ		SKIP PAR NAMES		
122	000246	175	261 314	LDMD	R75,=PTR2		SAVE BEGIN OF PAR STOR		
122	000251	377							
123	000252	263	150 215	STMD	R75,=XLPEN				
124	000255	110	012 241	LDM	R10,R12		COPY R12 STK PTR		
125	000260	134	262 160	STBD	R34,=YLPEN		ZERO PAR ?		
125	000263	215							
126	000264			POPPAR	BSS	0			
127	000264	367	075	JZR	LINKOK		JIF YES		

=====

>>>>				FUNCTION CALL RUNTIME		<<<<<<<<	
128	000266	133 270 315		LDBI R33,=PTR2-	PAR TYPE IN DEF		
129	000271	377					
129	000272	132 270 311		LDBI R32,=PTR1-	PAR TYPE IN CALL		
129	000275	377					
130	000276	317 200 200		AHM R32,=200,200	SAME TYPE ?		
131	000301	033 300		CMB R32,R33			
132	000303	266 047		JNZ BADP1	JIF NOT		
133	000305	175 271 315		LDHI R75,=PTR2-	SKIP NAME PTR		
133	000310	377					
134	000311	132 220		TSB R32	STRING PAR		
135	000313	264 012		JNG STRPR	JIF YES		
136	000315	170 271 315		LDHI R70,=PTR2-	SKIP PAR STORAGE		
136	000320	377					
137	000321	110 315 010		SBM R10,=10,0	SKIP PAR PASSED		
137	000324	000					
138	000325	260 021		JMP TSTMO			
139	000327		STRPR	BSS 0			
140	000327	175 271 315		LDHI R75,=PTR2-	GET TOTAL LEN		
140	000332	377					
141	000333	211		ICM R75	2-BYTE MAX		
142	000334	211		ICM R75			
143	000335	215		TCM R75	NEGATE FOR SKIP		
144	000336	323 314 377		ADMD R75,=PTR2	SKIP STORAGE		
145	000341	263 314 377		STMD R75,=PTR2			
146	000344	110 315 005		SBM R10,=5,0	SKIP PAR PASSED		
146	000347	000					
147	000350		TSTMO	BSS 0			
148	000350	134 212		DCB R34			
149	000352	260 310		JMP POPPAR			
150	000354						
151	000354		BADP1	BSS 0			
152	000354		BADP	BSS 0			
153	000354	175 006 343		POMD R75,-R6	THROW AWAY PTR TO FUNC		
154	000357	316 377 377	ER40	JSB =ERROR+			
155	000362	050		OCT 400	PARAMETER MISMATCH		
156	000363						
157	000363		LINKOK	BSS 0			
158	000363	165 261 314		LDMD R65,=PTR2	SAVE PTR2 FOR LATER		
158	000366	377					
159	000367	263 062 211		STMD R65,=TMP1			
160	000372	175 261 150		LDMD R75,=XLPEN	RESTORE PAR STOR ADDR		
160	000375	215					
161	000376	263 314 377		STMD R75,=PTR2			
162	000401	110 263 150		STMD R10,=XLPEN	SAVE R12 STK ADDR		
162	000404	215					
163	000405	134 260 160		LDBD R34,=YLPEN	GET PAR COUNT		
163	000410	215					
164	000411		NOVLP	BSS 0	LOOP TO MOVE PAR		
165	000411	367 120		JZR MVDONE	JIF NO MORE PAR		
166	000413	134 006 344		PUBD R34,+R6	SAVE PAR COUNT		
167	000416	133 270 315		LDBI R33,=PTR2-	GET PAR TYPE		
167	000421	377					
3	000422	264 025		JNG STRPR.	JIF STRING		


```

=====
> >>>> FUNCTION CALL RUNTIME <<<<<<<
1 000424 175 271 315 LDMI R75,=PTR2- SKIP NAME PTR
169 000427 377
170 000430 160 010 341 POMB R60,+R10 GET PAR
171 000433 316 377 377 JSB =ONER+ CONVERT TO REAL
172 000436 230 BIN
173 000437 160 273 315 STMI R60,=PTR2- STORE PAR
173 000442 377
174 000443 TSTM02 BSS 0
175 000443 134 006 342 POBD R34,-R6 RESTORE PAR COUNT
176 000446 212 DCB R34
177 000447 360 340 JMP MOVLP
178 000451
179 000451 STRPR. BSS 0
180 000451 163 010 341 POMB R63,+R10 SAVE SOURCE STRING
181 000454 175 271 315 LDMI R75,=PTR2- GET NAME POINTER
181 000457 377
182 000460 012 345 PUMB R75,+R12 PUSH NAME POINTER
183 000462 133 344 PUBD R33,+R12 PUSH HEADER
1 000464 ***** LDMI R71,=PTR2- GET TOT,MAX,ACT
1 000464 ***** LDMD R75,=PTR2 GET SINK ADDR
186 000464 ***** PUMB R73,+R12 PUSH MAX LEN, ADDR
187 000464 ***** PUMB R73,+R12 PUSH ONE MORE
188 000464 316 377 377 JSB =FTSTL2 PUSH DEST ADDR
189 000467 163 012 345 PUMB R63,+R12 PUSH SOURCE BACK
190 000472 165 073 241 LDM R65,R73 GET MAX LEN
191 000475 167 222 CLB R67 CLEAR HIGH BYTE
192 000477 175 065 305 SBM R75,R65 PT TO NEXT PAR STORAGE
193 000502 006 345 PUMB R75,+R6 SAVE POINTER
194 000504 316 377 377 JSB =STOST STORE THE STRING PAR
195 000507 175 006 343 POMB R75,-R6
196 000512 263 314 377 STND R75,=PTR2
197 000515 360 324 JMP TSTM02
198 000517
199 000517 SKPAN2 BSS 0
200 000517 170 220 TSB R70 ZERO COUNT ?
201 000521 367 007 JZR SKPANX JIF YES
202 000523 SKPA22 BSS 0
203 000523 316 377 377 JSB =SKP2NA SKIP ONE NAME
204 000526 170 212 DCB R70 DEC COUNT
205 000530 366 371 JNZ SKPA22 LOOP IF MORE
206 000532 236 SKPANX RTN
207 000533
208 000533 MYDONE BSS 0
209 000533 112 261 150 LDMD R12,=XLPEN POP PAR FROM R12 STK
209 000536 215
210 000537 175 006 343 POMB R75,-R6 RESTORE POINTER
211 000542 263 314 377 STMD R75,=PTR2
212 000545 261 310 377 LDMD R75,=PTR1 SAVE RETURN ADDR
213 000550 273 314 377 STMI R75,=PTR2
214 000553 261 115 202 LDMD R75,=PCR SAVE PCR
215 000556 273 315 377 STMI R75,=PTR2-
216 000561 176 261 344 LDMD R76,=TOS SAVE TOS
216 000564 203

```

```

>>>> FUNCTION CALL RUNTIME <<<<<<<<
000565 112 263 344 STMD R12,=TOS SETUP NEW TOS
217 000570 203
218 000571 176 273 315 STMI R76,=PTR2-
218 000574 377
219 000575 175 261 343 LDMD R75,=RMEM GET RES MEM COUNT
219 000600 212
220 000601 273 315 377 STMI R75,=PTR2- SAVE RES MEM
221 000604 223 CLM R75 RESET RES MEM
222 000605 263 343 212 STMD R75,=RMEM RESET RES MEM COUNT
223 000610 116 272 315 STBI R16,=PTR2- SAVE CSTAT
223 000613 377
224 000614 250 002 LDB R16,=2 SET RUN-MODE
225 000616 175 261 062 LDMD R75,=TMP1 XFER TO FUNC
225 000621 211
226 000622 263 310 377 STMD R75,=PTR1
227 000625 FNDPC BSS 0
228 000625 316 377 377 JSB =80P PT TO BEGIN OF PGM
229 000630 FNDPCR BSS 0 LOOKING FOR PCR
229 000630 ORP 165
230 000630 263 115 202 STMD R65,=PCR ASSUME PCR
232 000633 263 314 377 STMD R65,=PTR2
233 000636 271 315 377 LDNI R65,=PTR2- SKIP LINE #
234 000641 223 CLM R65 RESET
235 000642 270 315 377 LDNI R65,=PTR2- GET STSIZE
236 000645 215 TCM R65 NEGATE
237 000646 323 314 377 ADMD R65,=PTR2 ADVANCE ONE LINE
238 000651 075 301 CMM R65,R75 PASSED FN DEF ?
239 000653 373 353 JCY FNDPCR JIF NOT
240 000655 236 RTN
241 000656
242 000656
243 000656
244 000656 016
245 000657
246 000657 230
247 000660 112 321 344 FNRET. BIN
247 000663 203 CMMD R12,=TOS FN RESULT STRING ?
248 000664 367 071 JZR FNRTN. JIF NOTHING ON STACK
249 000666 362 003 JOD FNRETS JIF YES
250 000670 160 ORP 60
251 000671 360 001 JMP FNRET1
252 000673 FNRETS BSS 0
253 000673 163 ORP 63
254 000674 FNRET1 BSS 0
255 000674 012 343 POMD R#, -R12 POP OFF VALUE
256 000676 160 263 062 STMD R60,=TMP1 SAVE IT
256 000701 211
257 000702 165 271 313 LDNI R65,=PTR1-+ GET ADDR
257 000705 377
258 000706 367 042 JZR FNER39 JIF NOT DEFINED
259 000710 316 125 002 JSB =FNLET1 GET FN ADDR
260 000713 160 261 062 LDMD R60,=TMP1 RESTORE FUN VALUE
261 000716 211
  
```

*** FN RETURN ATTRIBUTES TABLE *****

OCT 16

FNRET. BIN

CMMD R12,=TOS FN RESULT STRING ?

JZR FNRTN. JIF NOTHING ON STACK

JOD FNRETS JIF YES

ORP 60

JMP FNRET1

FNRETS BSS 0

ORP 63

FNRET1 BSS 0

POMD R#, -R12 POP OFF VALUE

STMD R60,=TMP1 SAVE IT

LDNI R65,=PTR1-+ GET ADDR

JZR FNER39 JIF NOT DEFINED

JSB =FNLET1 GET FN ADDR

LDMD R60,=TMP1 RESTORE FUN VALUE

```

>>>> FUNCTION CALL RUNTIME <<<<<<<
261 000717 146 220 TSB R46 IS IT STRING FUNC ?
262 000721 364 010 JNG FNRET2 JIF YES
263 000723 160 012 345 PUMD R60,+R12
264 000726 316 377 377 JSB =STOSV ELSE STORE NUMBER
265 000731 360 024 JMP FNRTN.
266 000733 FNRET2 BSS 0
267 000733 163 012 345 PUMD R63,+R12 PUSH STRING
268 000736 316 377 377 JSB =STOST STORE STRING
269 000741 360 014 JMP FNRTN. DO FUNC
270 000743
271 000743 BADFN BSS 0
272 000743 106 213 DCM R6 TRASH RETURN
273 000745 213 DCM R6
274 000746 316 377 377 JSB =ERROR+
275 000751 050 OCT 400 BAD FUNC CALL
276 000752
277 000752 FNER39 BSS 0
278 000752 316 377 377 JSB =ERROR+
279 000755 047 OCT 390 FN MISSING
280 000756
281 000756 **** FN END ATTRIBUTES TABLE *****
282 000756 313 OCT 313
283 000757 *****
284 000757 FNRTN. BSS 0
285 000757 230 FNGET BIN
286 000760 316 377 377 JSB =RELMEM RELEASE TEMP MEM
287 000763 175 271 311 LDNI R75,=PTR1- GET ADDR
288 000766 377
289 000767 367 361 JZR FNER39 JIF FN NOT DEFINED
290 000771 323 006 200 ADMD R75,=FWCURR MAKE IT ABSOLUTE
291 000774 263 314 377 STMD R75,=PTR2
292 000777 164 271 315 LDNI R64,=PTR2- GET HEADER/NAME PTR
293 001002 377
294 001003 316 377 377 JSB =SETTR1 SAVE FOR RETRACE
295 001006 DRP 145
296 001006 271 315 377 LDNI R45,=PTR2- SKP FUNC DEF ADDR
297 001011 271 315 377 LDNI R45,=PTR2- RTN ADDR = 0 ?
298 001014 367 325 JZR BADFN JIF YES-BAD CALL
299 001016 175 223 CLM R75
300 001020 273 314 377 STMI R75,=PTR2 CLEAR RTN ADDR
301 001023 271 315 377 LDNI R75,=PTR2- GET PCR
302 001026 263 115 202 STMD R75,=PCR RESTORE PCR
303 001031 176 271 315 LDNI R76,=PTR2- GET TOS
304 001034 377
305 001035 263 344 203 STMD R76,=TOS RESTORE TOS
306 001040 ***** LDMD R75,=FWCURR POINT TO P.RMEM
307 001040 ***** SBM R75,=P.RMEG
308 001040 ***** OCT 0
309 001040 ***** STMD R75,=PTR1
310 001040 175 271 315 LDNI R75,=PTR2- GET P.RMEM
311 001043 377
312 001044 ***** STMI R75,=PTR1- RESTORE P.RMEM
313 001044 263 343 212 STMD R75,=RMEM RESTORE RES MEM COUNT

```

```

KARMA 03/14/81=====
ITEM   LOC   OBJECT CODE SRC=CCGFZ2 OBJ=GEZGFZ  9/11/1981 10:47 AM PG  8
=====
>>>>> FUNCTION CALL RUNTIME                                     <<<<<<<<
001047 145 263 310          STMD R45,=PTR1          PC = RTN ADDR
310 001052 377
311 001053 116 270 315      LDBI R16,=PTR2-          RESTORE CSTAT
311 001056 377
312 001057 763 003          JEV  FNGETO             JIF NOT CALC
313 001061 117 250 002      LDB R17,=2             FLAG INPUT COMPLETE
314 001064                      FNGETO  BSS  0
315 001064 167 220          TSB R67                 ISOLATE STR FLAG
316 001066 764 013          JNG STRV                JIF STRING
317 001070 160 271 315      LDMI R60,=PTR2-          NUMERIC VALUE
317 001073 377
318 001074 316 377 377      JSB  =FETIND            DO UNDEFINED TEST
319 001077 160 012 345      PUMD R60,+R12           PUSH VALUE
320 001102 236              RTN
321 001103
322 001103                      STRV   BSS  0
323 001103 316 377 377      JSB  =STRIND            FETCH STRING
324 001106 236              RTN
325 001107
326 001107                      **** FN LET ATTRIBUTES TABLE ****
327 001107 217              OCT 217
328 001110                      ****
329 001110                      FNLET.  BSS  0
330 001110 230              BIN
331 001111 165 270 311      LDBI R65,=PTR1-          STEP OVER VAR TOKEN
331 001114 377
332 001115 271 311 377      LDMI R65,=PTR1-          GET ADDR
333 001120 367 230          JZR  FNER39             JIF FN MISSING
334 001122 316 377 377      JSB  =SKPNM+           SKIP THE FN NAME
335 001125                      FNLET1 BSS  0
336 001125 316 377 377      JSB  =FETSYA           GET DEST ADDR
337 001130 175 261 314      LDMO R75,=PTR2          SKIP FUNC ADR, RTN ADR
337 001133 377
338 001134 315 017 000      SBM  R75,=17,0,0
338 001137 000
339 001140 263 314 377      STMD R75,=PTR2
340 001143 146 220          TSB  R46                 STRING FUNC ?
341 001145 765 004          JPS  HSTR                JIF NOT
342 001147 316 377 377      JSB  =FTSTL1
343 001152 236              RTN
344 001153
345 001153                      HSTR   BSS  0
346 001153 316 377 377      JSB  =FTAPU
347 001156 236              RTN
348 001157

```



```

PARMA 03/14/81=====
ITEM   LOC   OBJECT CODE SRC=CCGFZ2 OBJ=GENGFZ  9/11/1981 10:47 AM PG 9
=====
/ >>>> INPUT RUNTIME ROUTINES <<<<<<<<
350 001157          GMARK EQU 77          QUESTION MARK
351 001157
352 001157          **** TOKEN 137 ATTRIBUTES TABLE ****
353 001157 241          OCT 241
354 001160          *****
355 001160 230          INPUT. BIN          FOR COMPARES
356 001161 116 310 002          CMB R16,=2          RUN?
357 001164 366 105          JNZ INPER          INPUT ILLEGAL IN RUN
358 001166 175 261 310          LDMD R75,=PTR1          SAVE BASIC PC
358 001171 377
359 001172 263 151 210          STMD R75,=SAYPC
360 001175 112 263 204          STMD 12,=INPTOS          SAVE STACK FOR ERRORS
360 001200 200
361 001201 103 260 302          LDBD 3,=CRTSTS          SET UP FLAG
361 001204 377
362 001205 365 007          JPS INALFA
363 001207 316 377 377          JSB =ROMJSB          BFR INPUT FOR GRAF
364 001212 377 377          DEF GRAFIN
365 001214 001          OCT 1
366 001215 236          RTN
367 001216
368 001216 316 377 377 INALFA JSB =COPSUB          CIF SCTEMP=1 OR 2
369 001221 367 005          JZR INALF1          JIF NO
370 001223 270 310 203          LDBI R#, =P, PTR          BUFFER PTR
371 001226 366 003          JNZ INALF2          JIF BUFFER NOT EMPTY
372 001230 316 377 377 INALF1 JSB =DISP.          SET UP THINGS
373 001233 154 223          INALF2 CLM R54
374 001235 211          ICM R54          PUSH 1 FOR STRING LENGTH
375 001236 316 377 377          JSB =RESFIL          RESERVE & BLANK ONE BYTE
376 001241 126 250 077          LDB R26,=GMARK          LOAD A "?"
377 001244 272 317 377          STBI R26,=PTR2-+          PUSH IT
378 001247 316 377 377          JSB =TRAFIC          APPEND TO BUFFER
379 001252 316 377 377          JSB =PRLINE          PRINT IT
380 001255 122 251 236 SETR22 LDM R22,=INPBUF          USE INPBUF FOR INPUT
380 001260 200
381 001261 263 143 202          STMD R22,=INPTR          BUFFER
382 001264          * SET CALVRB SO WE CAN TRACH CALC
383 001264          * MODE VARIABLES LATER.
384 001264          * LDMD R22,=LAVAIL
385 001264          * STMD R22,=CALVRB
386 001264 116 250 004          LDB R16,=4          SET IDLE IN INPUT
387 001267 316 377 377          JSB =SET240
388 001272 236          RTN
389 001273 316 377 377 INPER JSB =ERROR+          BAD STMT
390 001276 130          OCT 380
391 001277          *****

```

```

=====
ARMA 03/14/81=====
ITEM    LOG    OBJECT CODE  SRC=CCGFZ2 OBJ=GEMGFZ  9/11/1981 10:47 AM  PG 10
=====
>>>>  RUNTIME INPUT STRING ROUTINE                                     <<<<<<<<
001277 033                                OCT 33                ATTRIBUTES FOR INPU$.
394 001300 230                INPU$.  BIN
395 001301 175 261 022        LDMD R75,=HXTMEM                SET FOR PARSE
395 001304 200
396 001305 263 314 377        STMD R75,=PTR2
397 001310 263 341 203        STMD R75,=STSIZE                FOR PARSE
398 001313 261 310 377        LDMD R75,=PTR1                SAVE BASIC PC
399 001316 263 151 210        STMD R75,=SAVPC
400 001321 112 263 344        STMD R12,=TOS
400 001324 203
401 001325 110 251 205        LDM R10,=INCOM$                COMPLETION ADDR
401 001330 003
402 001331 263 167 200        STMD R10,=INPCOM
403 001334 261 143 202        LDMD R10,=INPTR                SET UP FOR PARSE
404 001337 316 377 377        JSB =GCHAR                    FIRST NON-BLANK
405 001342
405 001342
406 001342
406 001342
407 001342
409 001342
410 001342
411 001342
411 001342
411 001342
412 001342
413 001342 120 310 015        CMB R20,=CR                    CR HERE, SO NO DCM LATER?
414 001345 367 047            JZR INPHUL                      JIF YES, STORE INPTR LATER
415 001347 110 263 143        STMD R10,=INPTR                STORE R10+1
415 001352 202
416 001353
416 001353
417 001353
418 001353
419 001353
420 001353
420 001353
420 001353
421 001353
422 001353 120 310 042        CMB R20,=QUOTE                IS IT A QUOTED STRING?
423 001356 367 010            JZR YESSTR                      JIF YES
424 001360 021 242            STB R20,R21                    SAVE FOR UNSHIFTED
425 001362 316 377 377        JSB =ALFA
426 001365 371 017            JEZ NOTSTR                      JIF NOT ALFA
427 001367 240                LDB R#,R#                      RESTORE R20
428 001370 316 377 377 YESSTR JSB =STREX+                    STRING EXPR
429 001373 371 011            JEZ NOTSTR                      JIF NOT STRING EXPR
430 001375 114 310 015        CMB R14,=CR                    CR?
431 001400 367 127            JZR INPC0                      JIF VALID STR EXPR
432 001402 310 054            CMB R14,=COMMA                  COMMA?
433 001404 367 123            JZR INPC0                      JIF VALID FTR EXPR
434 001406 316 377 377 NOTSTR JSB =REP.1B                    CLEAR ERRORS
434 001411 110 261 143        LDMD R10,=INPTR                AND INPTR

```

>>>>> RUNTIME INPUT STRING ROUTINE <<<<<<<<

```
435 001414 202
436 001415 213          DCM R10          CR WAS NOT POSSIBLE!
437 001416 112 261 344 INPNUL LDMD R12,=TOS AND R12
437 001421 203
438 001422 114 250 042          LDB R14,=QUOTE LOOK LIKE QUOTED STR
439 001425          *
439 001425          *
440 001425          * For inputs like A$CDEF          *
440 001425          *
440 001425          *
441 001425
442 001425 175 261 341          LDMD R75,=STSIZE RESTORE PARSE POINTER
442 001430 203
443 001431 263 314 377          STMD R75,=PTR2
444 001434 316 377 377          JSB =UNQUOT          UNQUOTE STR TO , OR CR
445 001437 134 250 016          LDB R34,=16          APPEND AN EOL
446 001442 272 315 377          STBI R34,=PTR2-
447 001445 120 310 015          CMB R20,=CR
448 001450 367 122          JZR INPN1.
449 001452 360 063          JMP INPC0+          NOW EXIT
450 001454
```

RUNTIME INPUT NUMERIC ROUTINE

<<<<<<<<

001454						* ROUTINE INPUN. GETS ONE NUMERIC VALUE	
453 001454						* AND PUTS ON STACK (DOES NOT STORE!)	
454 001454						*** TOKEN 221 ATTRIBUTES TABLE	*****
455 001454	033					OCT 33	
456 001455						*****	
457 001455	230					INPUN. BIN	GCHAR NEEDS BINARY
458 001456	175 261 022					LDMD R75,=NXTMEM	FOR PARSE
459 001461	200						
459 001462	263 314 377					STMD R75,=PTR2	
460 001465	263 341 203					STMD R75,=STSIZE	
461 001470	261 310 377					LDMD R75,=PTR1	SAVE BASIC PC
462 001473	263 151 210					STMD R75,=SAVPC	
463 001476	110 251 226					LDM R10,=INCOM.	COMPLETION ADDR
463 001501	003						
464 001502	263 167 200					STMD R10,=INPCOM	SAVE COMPLETION
465 001505	261 143 202					LDMD R10,=INPTR	SET UP FOR PARSE
466 001510	316 377 377					JSB =GCHAR	
467 001513	316 377 377					JSB =NUMVA+	PARSE NUM VALUE
468 001516	371 147					JEZ ER43	JIF NUMVAL NOT FOUND
469 001520	114 310 015					CMB R14,=CR	ONLY CR AND COMMA ARE
470 001523	367 004					JZR INPCO	VALID HERE, MUST FLAG
471 001525	310 054					CMB R14,=COMMA	STRING LIKE "YES" AS
472 001527	366 136					JNZ ER43	NON-NUMERIC (Y IS VRBL)
473 001531	134 250 016 INPCO					LDB R34,=16	
474 001534	272 315 377					STBI R34,=PTR2-	PUSH EOL
475 001537	175 261 151 INPCO+					LDMD R75,=SAVPC	RESTORE BASIC PC
476 001542	210						
476 001543	263 314 377					STMD R75,=PTR2	
477 001546	136 271 317					LDMI R36,=PTR2++	LOOK AHEAD 2 TOKEN
477 001551	377						
478 001552	316 377 377					JSB =TSTEND	EXPECTING MORE?
479 001555	114					ORP 14	
480 001556	370 006					JEN CKCOMA	JIF YES
481 001560	310 015					CMB R14,=CR	<CR>?
482 001562	366 117					JNZ ER45	TOO MANY INPUTS
483 001564	360 006					JMP INPN1.	
484 001566		CKCOMA				ORP 114	
485 001566	310 054					CMB R14,=COMMA	COMMA?
486 001570	366 103					JNZ ER44	TOO FEW INPUTS
487 001572	110 213					DCM R10	DCM FOR NEXT PASS
488 001574	110 263 143 INPN1.					STMD R10,=INPTR	FOR NEXT TIME
489 001577	202						
489 001600	116 250 005					LDB R16,=5	SET EXECUTING IN INPUT
490 001603	360 050					JMP INPRTH	FORCE EXIT
491 001605		INCOM#				BSS 0	
492 001605	230					BIN	
493 001606	130 012 241					LDM R30,R12	COPY STACK PTR
494 001611	315 012 000					SBM R30,=100,0	BACKUP PASS DEST LEN
495 001614	171 030 245					LDMD R71,R30	LOAD DEST + SRCE LEN
496 001617	176 071 301					CMM R76,R71	CMP SRC LEN TO DEST
497 001622	367 002					JZR INCOM.	JIF EQ
498 001624	373 033					JCY ER56	JIF OVFL
499 001626		INCOM.				BSS 0	

ARMA 03/14/81=====

ITEM	LOC	OBJECT CODE	SRC=CCGFZZ	OBJ=GENCFZ	9/11/1981	10:47 AM	PG 13
------	-----	-------------	------------	------------	-----------	----------	-------

=====

>>>>> RUNTIME INPUT NUMERIC ROUTINE <<<<<<<<

5	001626			INCM	BSS	0	
501	001626	230			BIN		
502	001627	175	261	151	LDMD	R75,=SAVPC	RESTORE BASIC PC
502	001632	210					
503	001633	263	310	377		STMD	R75,=PTR1
504	001636				* NOW	CLEAN UP IN CASE IN FOR/NEXT	
505	001636	316	377	377		JSB	=RELMEM
506	001641	175	261	030	LDMD	R75,=CALVRS	TRASH CALC VARB
506	001644	200					
507	001645	263	025	200		STMD	R75,=LAVAIL
508	001650	250	002		LDB	R75,=2	INPUT COMPLETE
509	001652	117	075	224	GRB	R17,R75	
510	001655	316	377	377	INPRTH	JSB	=SET240
511	001660	236				RTN	RETURN TO EXEC

>>>> INPUT ERROR RECOVERY				<<<<<<<<	
514	001661	316 377 377 ER56	JSB =ERROR		
	001664	070	OCT 56D	STRING OVERFLOW	
515	001665	360 031	JMP BADA..		
516	001667	316 377 377 ER43	JSB =ERROR	NUMERIC INPUT	
517	001672	053	OCT 43D		
518	001673	360 012	JMP BADA++		
519	001675	316 377 377 ER44	JSB =ERROR	TOO FEW INPUTS	
520	001700	054	OCT 44D		
521	001701	360 004	JMP BADA++		
522	001703	316 377 377 ER45	JSB =ERROR	TOO MANY INPUTS	
523	001706	055	OCT 45D		
524	001707	175 261 111 BADA++	LDMD R75,=ERGOTO	ON ERROR?	
524	001712	200			
525	001713	367 003	JZR BADA..	JIF NO	
526	001715	106 213	DCM R6	TRASH 1 RETURN	
527	001717	213	DCM R6		
528	001720	175 261 115 BADA..	LDMD 75,=PCR	RESET PC	
528	001723	202			
529	001724	230	BIN	ADD NEEDS BINARY	
530	001725	315 004 000	SBM 75,=4,0,0	SKIP OVER L.N., LENGTH	
530	001730	000			
531	001731	263 310 377	STMD R75,=PTR1		
532	001734	316 377 377	JSB =ROMRTH	RESELECT ROM 0	
533	001737		*NOW SEE IF THIS IS AN INPUT OR ENTER STMT		
534	001737	112 261 204	LDMD 12,=INPTOS	RESET TOP OF STACK	
534	001742	200			
535	001743	316 377 377	JSB =RELMEM	RELEASE TEMP MEMORY	
536	001746	316 377 377	JSB =REPORT	REPORT IT	
537	001751	236	RTN	DONE	

>>>> EXTERNALS <<<<<<<<

539 001752	EXT ALFA
540 001752	EXT SETTR1
541 001752	EXT CRDF?
542 001752	EXT BOP
543 001752	EXT DEFFN.
544 001752	EXT SKP2NA
545 001752	EXT STRIND
546 001752	EXT RELMEM
547 001752	EXT RESFIL
548 001752	EXT COPSUB
549 001752	EXT TSTEND
550 001752	EXT REPORT
551 001752	EXT REP.1B
552 001752	EXT UNQUOT
553 001752	EXT STREX+
554 001752	EXT DISP.
555 001752	EXT GCHAR
556 001752	EXT NUMVA+
557 001752	EXT SET240
558 001752	EXT FETIND
559 001752	EXT TRAFIC
560 001752	EXT PRLINE
561 001752	EXT ERROR
562 001752	EXT ERROR+
563 001752	EXT ROMRTH

565 001752	*	HED SEARCH FOR A TOKEN
566 001752	*	
567 001752	*	
568 001752	*	EXT GETNXT
569 001752	*	EXT SKFRS
570 001752	*	EXT BININT
571 001752	*	EXT SKPCON
572 001752	*	EXT SKPNAM
573 001752	*	EXT RELJMP
574 001752	*	EXT MATCH?
575 001752	*	EXT LABTK?
576 001752	*	EXT SKPANM

578 001752	*	HED SEARCH FOR DEF FN TOKEN
------------	---	-----------------------------

579 001752	*	
580 001752	*	SEARCH FOR A TOKEN

581 001752	*	
582 001752	*	IN:
583 001752	*	R27=DEF FN TOKEN
584 001752	*	PTR2=POINTS TO NAME OR VALUE OR LINE #
585 001752	*	PTR1=STARTING ADDR TO SEARCH FROM

586 001752	*	
587 001752	*	OUT:
588 001752	*	IF NOT FOUND THEN E=0 ;
589 001752	*	
590 001752	*	IF FOUND THEN E#0 &
591 001752	*	PTR1=POINTS TO MATCHING TOKEN

>>>>> EXTERNALS <<<<<<<<

001752	*			
003 001752	*			
594 001752	*EOL	BSS 0		
595 001752	*	LDMI R74,=PTR1-	POP LINE# & STSIZE	
596 001752	*	CMM R75,=231,231,12	LAST LINE ?	
597 001752	*	JNZ EOL1	JIF NOT	
598 001752	*	CLE	SIGNIFY NOT FOUND	
599 001752	*	POMD R36,-R6	TRASH ONE RETURN	
600 001752	*EOL1	RTH		
601 001752	*			
602 001752	*SRCHTK	BSS 0		
603 001752	*	JSB =GETNXT	GET NEXT TOKEN	
604 001752	*	CMB R23,R27	SAME TOKEN ?	
605 001752	*	JZR SRCHMO	JIF YES	
606 001752	*SRCHNX	BSS 0		
607 001752	*	CMB R36,=30	CLASS >= 30	
608 001752	*	JCY SRCHTK	JIF YES	
609 001752	*	LLM R36	*2 FOR JSB	
610 001752	*	LDMD R36,X36,SKTAB	FETCH ENTRY	
611 001752	*	JSB X36,ZRO	SKIP THIS TOKEN	
612 001752	*	JMP SRCHTK	KEEP ON SEARCHING	
613 001752	*			
614 001752	*SRCHMO	BSS 0		
615 001752	*	LDMD R65,=PTR1	SAVE POINTER	
616 001752	*	PUMD R65,+R6		
617 001752	*	LDMI R65,=PTR1-	SKIP ADDR	
618 001752	*	JSB =MATCH?	SAME NAME ?	
619 001752	*	DRP 165		
620 001752	*	POMD R65,-R6	RESTORE POINTER	
621 001752	*	STMD R65,=PTR1		
622 001752	*	JEN FRRET	JIF FOUND	
623 001752	*	JMP SRCHNX	ELSE SRCH NXT TOKEN	
624 001752	*			
625 001752	*SKTAB	BSS 0		
626 001752	*	DEF EOL	0 END OF LINE PROCESSING	
627 001752	*	DEF SKPYAR	1 FETCH VAR	
628 001752	*	DEF BININT	2 INT CONST	
629 001752	*	DEF SKPVAX	3 STORE VAR	
630 001752	*	DEF SKPCON	4 REAL CONST	
631 001752	*	DEF SKPHAM	5 STRING CONST	
632 001752	*	DEF SKPFNC	6 FN CALL	
633 001752	*	DEF SKPLIN	7 JMP LINE#	
634 001752	*	DEF SKPLIN	10 GOTO GOSUB	
635 001752	*	DEF RELJMP	11 JMP REL	
636 001752	*	DEF SKPFN	12 FN DEF	
637 001752	*	DEF SKPADR	13 FN END	
638 001752	*	DEF SKFROM	14 EXT ROM	
639 001752	*	DEF FRRET	15 OPTION BASE	
640 001752	*	DEF SKPADR	16 FN RET	
641 001752	*	DEF FRRET	17 FN LET	
642 001752	*	DEF FRRET	20 DATA	
643 001752	*	DEF FRRET	21 DIM	
644 001752	*	DEF FRRET	22 SHORT	


```

>>>>> EXTERNALS <<<<<<<
001752 * DEF FRRET 23 INTEGER
646 001752 * DEF FRRET 24 COM
647 001752 * DEF SKPLIN 25 ELSE JMP#
648 001752 * DEF RELJMP 26 ELSE JMP REL
649 001752 * DEF SKPLIN 27 USING LINE #
650 001752 *
651 001752 +SKPVAX BSS 0
652 001752 * CMB R23,=237 LABEL TOKEN ?
653 001752 * JZR SKPVA2 JIF YES
654 001752 +SKPVAR BSS 0
655 001752 * LDNI R75,=PTR1- SKIP ADDR
656 001752 +SKPVA2 BSS 0
657 001752 * JSB =SKPNAM SKIP VAR NAME
658 001752 +FRRET RTH
659 001752 *
660 001752 +SKPLIN BSS 0
661 001752 * JSB =LASTK? LABEL TOKEN ?
662 001752 * JZR SKPVAR JIF YES
663 001752 +SKPADR BSS 0
664 001752 * LDNI R75,=PTR1- SKIP LN# OR ADDR
665 001752 * RTH
666 001752 *
667 001752 +SKPRON BSS 0
668 001752 * LDNI R76,=PTR1- SKIP ROM# & TOKEN
669 001752 * RTH
670 001752 *
671 001752 +SKPFNC BSS 0
672 001752 * JSB =SKPVAR SKIP FN ADR & NAME
673 001752 * JSB =SKPNAM SKIP PAR TYPES
674 001752 * RTH
675 001752 *
676 001752 +SKPFN BSS 0
677 001752 * JSB =SKPVAR SKIP FUNC ADR & NAME
678 001752 * LDNI R75,=PTR1- SKIP JUMP ADDR
679 001752 * LDBI R70,=PTR1- GET PAR COUNT & STR FLG
680 001752 * LR9 R70 DIV BY 2 TO GET COUNT
681 001752 * JSB =SKPANM SKIP PAR NAMES
682 001752 * JSB =SKPRS SKIP PARAMENTERS
683 001752 * RTH
684 001752
685 001752 FIN

```

SYMBOL	VALUE	TYPE	COUNT	Symbol Table
ALFA	177777	EXT	1	
BA++	001707	LCL	2	
BAL	001720	LCL	2	
BADFN	000743	LCL	1	
BADP	000354	LCL	1	
BADP1	000354	LCL	1	
BOP	177777	EXT	2	
BALOK	000167	LCL	1	
BALVRB	100030	G DAD	1	
BKCOMA	001566	LCL	1	
COHNA	000054	G EQU	3	
COPSUB	177777	EXT	1	
CR	000015	G EQU	5	
CRDF?	177777	EXT	1	
CRSTTS	177702	G DAD	1	
DEFFN.	177777	EXT	1	
DISP.	177777	EXT	1	
ER40	000357	LCL	0	<--- NOT REFERENCED??
ER43	001667	LCL	2	
ER44	001675	LCL	1	
ER45	001703	LCL	1	
ER	001661	LCL	1	
ERGOTO	100111	G DAD	1	
ERRFN	000146	LCL	1	
ERRFN1	000151	LCL	0	<--- NOT REFERENCED??
ERROR	177777	EXT	4	
ERROR+	177777	EXT	7	
FETIND	177777	EXT	1	
FETSV4	177777	EXT	1	
FNCAL\$	000001	ENT	1	
FNCAL-	000032	LCL	1	
FNCAL.	000007	ENT	1	
FNCAL0	000012	LCL	1	
FNCAL1	000040	LCL	1	
FNCAL2	000044	LCL	1	
FNCAL3	000153	LCL	1	
FNDPC	000625	LCL	1	
FNDPCR	000630	LCL	1	
FNER39	000752	LCL	3	
FNGET	000757	LCL	0	<--- NOT REFERENCED??
FNT	001064	LCL	1	
FHLET.	001110	ENT	1	
FHLET1	001125	LCL	1	
FHRET.	000657	ENT	1	
FHRET1	000674	LCL	1	
FHRET2	000733	LCL	1	
FHRETS	000673	LCL	1	
FNRTH.	000757	ENT	4	
FTAPU	177777	EXT	1	
FTSTL1	177777	EXT	1	
FTSTL2	177777	EXT	1	
FWCURR	100006	G DAD	3	
GCHAR	177777	EXT	2	
GINTDS	177401	G DAD	1	

```

=====
KARMA 03/14/81=====
SYMBOL  VALUE      TYPE      COUNT      Symbol Table      9/11/1981 10:47 AM PG 19
=====
GINTEN  177400      G DAD      1
GFIN    177777      EXT      1
IF1     001230      LCL      1
INHALF2 001233      LCL      1
INALFA  001216      LCL      1
INCML   001626      LCL      0      <--- NOT REFERENCED??
INCOM$  001605      LCL      1
INCOM,  001626      LCL      2
INPBUF  100236      G DAD      1
INPCO   001531      LCL      3
INPCO+  001537      LCL      1
INPCOM  100167      G DAD      2
INPER   001273      LCL      1
INPN1,  001574      LCL      2
INPHUL  001416      LCL      1
INPRTN  001655      LCL      1
INPTOK  000137      EQU      0      <--- NOT REFERENCED??
INPTOS  100204      G DAD      2
INPTR   101143      G DAD      6
INPU$.  001300      ENT      1
INRUN,  001435      ENT      1
INUT,   001160      ENT      1
LAVAIL  100025      G DAD      1
LINKOK  000363      LCL      1
MOVLP   000411      LCL      1
MYDONE  000533      LCL      1
NOTSTR  001406      LCL      2
NSTR    001153      LCL      1
NMYA+   177777      EXT      1
NMEM    100022      G DAD      2
ONER+   177777      EXT      1
P.PTR   101710      G DAD      1
PCR     101115      G DAD      6
POPPAR  000264      LCL      1
PRLINE  177777      EXT      1
PTR1    177710      G DAD     12
PTR1-   177711      G DAD      6
PTR1-+  177713      G DAD      1
PTR2    177714      G DAD     22
PTR2-   177715      G DAD     27
P-+     177717      G DAD      2
QMARK   000077      EQU      1
QUOTE   000042      G EQU      2
RELMEM  177777      EXT      3
REP.18  177777      EXT      1
REPORT  177777      EXT      1
RESFIL  177777      EXT      1
RMEM    105343      G DAD      3
ROMJ8B  177777      EXT      2
ROMRTN  177777      EXT      1
SAVPC   104151      G DAD      5
SET240  177777      EXT      2
SETALO  177777      EXT      0      <--- NOT REFERENCED??
SETR22  001255      LCL      0      <--- NOT REFERENCED??

```

SETTR1	177777	EXT	1
SENA	177777	EXT	1
SKF22	000523	LCL	1
SKPAN2	000517	LCL	1
SKPANX	000532	LCL	1
SKPNM+	177777	EXT	2
SRCHTK	177777	EXT	1
STAKOK	000206	LCL	1
STOST	177777	EXT	2
STOSV	177777	EXT	1
STREX+	177777	EXT	1
STRIND	177777	EXT	1
STRPR	000327	LCL	1
STRPR.	000451	LCL	1
STRV	001103	LCL	1
STSIZE	101741	G DAD	3
TMP1	104462	G DAD	4
TOS	101744	G DAD	6
TRAFIC	177777	EXT	1
STEND	177777	EXT	1
TS10	000350	LCL	1
TS102	000443	LCL	1
UNQUOT	177777	EXT	1
ALPEN	106550	G DAD	5
ESSTR	001370	LCL	1
ALPEN	106560	G DAD	2

TABLE OF CONTENTS	1
FUNCTION CALL RUNTIME	2
INIT RUNTIME ROUTINES	3
RUNTIME INPUT STRING ROUTINE	10
RUNTIME INPUT NUMERIC ROUTINE	12
INPUT ERROR RECOVERY	14
EXTERNALS	15

DCGFZZ HAD 0 ERRORS 0 WARNINGS 134 LABELS LAST ERROR AT 0