

ARMA 03/14/81=====

ITEM	LOC	OBJECT CODE	SRC=CCHRX	OBJ=GENHRX	9/11/1981	11:40 AM	PG 1
------	-----	-------------	-----------	------------	-----------	----------	------

=====

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

000000 * UHL

TABLE OF CONTENTS

>>>>>	000000	*	2	ENTRY POINTS	<<<<<<<
5	000000	*	4	MAIN LN(X) SUBROUTINE	
6	000000	*	7	DIVIDE	
7	000000	*	7	SHIFT	
8	000000	*	9	LOG BASE 10	
9	000000	*	9	MOD/REM	
10	000000	*	11	REAL/INTEGER CH3	
11	000000	*	11	REAL/INTEGER SUB	
12	000000	*	11	REAL/INTEGER ADD	
13	000000	*	13	EXP ENTRY	
14	000000	*	14	SQUARE ROOT	
15	000000	*	15	MULTIPLY	
16	000000	*	17	INTEGER MULTIPLY	
17	000000	*	18	RANDOM NUMBER GENERATOR	
18	000000	*	19	Y*X	
19	000000	*	20	INF	
20	000000	*	21	SGN	
21	000000	*	21	FTR ENTRY	
22	000000	*	21	DRCOM	
23	000000	*	21	LOGCOM	
24	000000	*	23	PI	
25	000000	*	23	CEILING	
26	000000	*	23	OVF/UF	
27	000000	*	23	DEG-RAD	
28	000000	*	24	ABS	
29	000000	*	24	INT(=FLOOR)	
30	000000	*	24	A\B	
31	000000	*	25	FP	
32	000000	*	25	NFR	
33	000000	*	25	EPS	
34	000000	*	25	RAD-DEG	
35	000000	*	25	IP	
36	000000	*	27	N FACTORIAL	
37	000000	*	29	MAIN FORWARD TRIG SUBROUTINE	
38	000000	*	33	RANDOMIZE	
39	000000	*	33	ROUND	
40	000000	*	33	SEPARATE	
41	000000	*	35	UTILITIES	
42	000000	*	35	PQO	
43	000000	*	35	MIN/MAX	
44	000000	*	36	MAIN EXP(X) SUBROUTINE	
45	000000	*	38	OCT-DEC-OCT	
46	000000	*	40	TABLES AND CONSTANTS	
47	000000	*	41	REAL/INTEGER DATA FETCH	

TEMP ENTRY POINT FOR HRTEST

ENTRY POINTS, EXTERNALS

CCCCCCCC

100	000000	ENT MOD10
103	000000	ENT MPY10
104	000000	ENT MPY28
105	000000	ENT MPY30
106	000000	ENT MPYRO1
107	000000	ENT MPYR
108	000000	ENT NFR
109	000000	ENT NFR5
110	000000	ENT QVF10
111	000000	ENT PI10
112	000000	ENT RAD10
113	000000	ENT REM10
114	000000	ENT RND10
115	000000	ENT RNDINI
116	000000	ENT RNDIZ.
117	000000	ENT RNDSET
118	000000	ENT RDNF
119	000000	ENT ROU10
120	000000	ENT SSROI+
121	000000	ENT SEC10
122	000000	ENT SEP10
123	000000	ENT SEP15
124	000000	ENT SEP20
125	000000	ENT SGN5
126	000000	ENT SHF10
127	000000	ENT SHRONF
128	000000	ENT SIN10
129	000000	ENT SQR30
130	000000	ENT SQR5
131	000000	ENT SUB10
132	000000	ENT SUBIL
133	000000	ENT SUBROI
134	000000	ENT TAN10
135	000000	* ENT TBL2
136	000000	* ENT TBL3
137	000000	ENT TBL3B
138	000000	ENT TRC01
139	000000	ENT TRC7
140	000000	ENT TRCS1
141	000000	ENT UT210
142	000000	ENT YTX5

TEMP ENTRY PT FOR HRTEST

143	000000	
144	000000	* EXTERNALS
145	000000	EXT CONINT
146	000000	EXT ERROR
147	000000	EXT ERROR+
148	000000	EXT RTOIN
149	000000	EXT BRTS35
150	000000	EXT BRTS40
151	000000	EXT TIME.
152	000000	EXT ON?R
153	000000	EXT ONEB
154	000000	

```

>>>>>      MAIN LN(X) SUBROUTINE      <<<<<<<<
000000 316 045 012 LN20  JSB= ONER      GET REAL X.
137 000003 316 102 010  JSB= SEP10     SEPARATE X.
158 000006 140 221      LN22  TSM R40     M(X)=0?
159 000010 366 004      JNZ LN24      JIF M(X)#0.
160 000012 316 377 377  JSB= ERROR+  ELSE REPORT LN(0) OR LOG(0).
161 000015 014      OCT 120      ERROR NUMBER.
162 000016 132 220      LN24  TSB R32     X>0?
163 000020 376 054      JRZ LN30      JMP & FIND LN(X) IF X>0.
164 000022 316 377 377  JSB= ERROR+  ELSE REPORT LOG OF NEG NO.
165 000025 015      OCT 130      CORRESPONDING ERROR NO.
166 000026 140 050 243 LN36  STM R40,R50 COPY X.
167 000031 147 314 020  SBB R47=10C X-1.
168 000034 316 142 001  JSB= SHF10  ACCUM -J; NORM X-1 PRIOR TO /.
169 000037 136 034 243  STM R36,R34  MOVE EXP(X-1) TO R34.
170 000042 223      CLM R36     EXP(X)=0.
171 000043 140 221      TSM R40     X=1?
172 000045 367 123      JZR LN34      JIF YES; DONE, LOG(1)=0 IN R40.
173 000047 316 141 010  JSB= UT210  SWAP X & X-1.
174 000052 316 137 001 LN38  JSB= DIV20  (X-1)/X OR X/(X+1).
      000055 136 211      ICM R36     MOVE DEC ALL WAY TO LEFT.
176 000057 136 034 243 LN40  STM R36,R34  MOVE EXP.
177 000062 170 223      CLM R70     CLEAR PQ AREA.
178 000064 134 215      TCM R34     J.
179 000066 311 010 000  CHM R34=8C,0 J>=8?
180 000071 374 164      JLZ LN80      JIF YES; NO PQ NEEDED.
181 000073 235      CLE      READY FOR PSEUDO-DIVIDE.
182 000074 360 100      JMP LN50      JMP TO PSEUDO-DIVIDE LOOP.
      000076 136 221      LN30  TSM R36     1<=X<10?
184 000100 367 324      JZR LN36      JIF YES.
185 000102 140 215      TCM R40     10-10XM.
186 000104 136 211      LN32  ICM R36     MOVE DEC ALL WAY TO LEFT.
187 000106 026 243      STM R36,R26  SAVE EX.
188 000110 223      CLM R36     EXP(1-XM).
189 000111 316 142 001  JSB= SHF10  SHIFT OUT LEADING 0'S.
190 000114 316 057 000  JSB= LN40   -B0=1-XM; GO FIND LN.
191 000117 140 012 345  PUMD R40+R12 SAVE M(LN(XM)).
192 000122 223      CLM R40     READY FOR EX.
193 000123 133 222      CL3 R33     S(LN10).
      000125 146 026 241  LDM R46,R26  MOVE EX.
195 000130 365 003      JPS LN33      JIF POS.
196 000132 215      TCM R46     ABS(EX).
197 000133 132 216      NCB R32     GIVES -EX.
198 000135 136 014 243 LN33  STM R36,R14  SAVE EXP(LN(XM)).
199 000140 251 003 000  LDM R36=3C,0 MAX EXP(EX).
200 000143 316 142 001  JSB= SHF10  ELIM LEAD 0'S IN EX.
201 000146 134 223      CLM R34     EXP(LN10).
202 000150 150 261 241  LDM R50=TBL2 M(LN10).
202 000153 011
203 000154 316 035 003  JSB= MPY30  (EX)<(LN10).
204 000157 134 014 241  LDM R34,R14  GET EXP(LN(XM)).
205 000162 150 012 343  POMD R50-R12 GET M(LN(XM)).
206 000165 133 216      NCB R33     LN(XM)<0.
207 000167 316 106 002  JSB= ADD20  LNX=LN(XM)+(EX)<(LN10).
  
```

```

>>>>>      MAIN LN(X) SUBROUTINE
2 000172 236      LN34  RTH      END.
209 000173 233      LN62  DCE      -BI(10 TO J)-BI-1.
210 000174 170 210      ICB R70      INCREMENT PQ DIGIT.
211 000176 140 060 243 LN50  STM R40,R60 COPY BI(10 TO J).
212 000201 136 221      TSM R36      J=0?
213 000203 367 022      JZR LN58      JIF YES; DECADE 0, NO SHIFTS.
214 000205 167 203      ERM R67      1ST RT SHIFT TO ALIGN BI FOR +.
215 000207 167 000 242 LN52  STB R67,R0  SAVE CONTENTS OF E.
216 000212 131 036 240      LD3 R31,R36  SHIFT COUNTER.
217 000215 360 002      JMP LN56      CONTINUE.
218 000217 167 207      LRM R67      SHIFT BI.
219 000221 131 210      LN56  ICB R31      INCREMENT COUNTER.
220 000223 366 372      JNZ LN54      LOOP TILL CTR=0; BI ALIGNED.
221 000225 100 204      LL3 R0      RESTORE E.
222 000227 140 060 303 LN52  ADM R40,R60 <-BI>(10 TO J)+<-BI>.
223 000232 372 001      JNC LN60      JIF NO CARRY.
224 000234 234      ICE      ELSE UP DIGIT IN E.
225 000235 370 334      LN60  JEN LN62      JIF E#0; MORE TO CURRENT DECADE.
226 000237 140 060 305      SBM R40,R60  RESTORE TO VAL BEFORE OVERDRAFT.
227 000242 243      STM R40,R60  COPY <-BI>(10 TO J).
228 000243 170 205      LLM R70      SHIFT PQ.
229 000245 136 310 223      CMB R36=93C  ALL 3 PQ DIGITS DONE?
230 000250 367 005      JZR LN80      EXIT PSEUDO-DIV LOOP IF=0.
231 000252 213      DCM R36      ELSE DECREMENT PQ DIGIT CTR.
232 000253 140 205      LLM R40      MIN 1 RT SHIFT; THIS IS 1ST.
233 000255 360 330      JMP LN52      CONTINUE LOOP.
234 000257 136 213      LN80  DCM R36      DEC PT TO NORMAL SPOT.
235 000261 316 333 000      JSB= HEWR  GO CALCULATE R/(1-.5R).
236 000264 132 222      CL3 R32      S(LNXM)=0.
237 000266 134 251 141      LDM R34=TBL1 POINT TO 1ST LN CONSTANT.
238 000271 011
239 000272 140 205      LLM R40      TEMP SHIFT R.
240 000274 136 213      DCM R36      ADJUST EXP.
241 000276 360 012      JMP LN96      JMP TO PSEUDO-MULTIPLY LOOP.
242 000300 212      LN92  DCB R#      DECREMENT PSEUDO-QUOTIENT DIG.
243 000301 140 303      ADM R40,R#      ACCUMULATE PARTIAL SUM PS.
244 000303 372 001      JNC LN94      JIF NO CARRY.
245 000305 234      ICE      ELSE ADJUST DIGIT IN E.
246 000306 170 220      LN94  TSB R70      DONE WITH CURRENT PQ DIGIT?
247 000310 377 366      JRN LN92      LOOP TILL PQ DIGIT = 0.
248 000312 170 221      LN96  TSM R70      PQ=0?
249 000314 367 064      JZR LN98      EXIT LOOP WHEN PQ=0.
250 000316 147 203      ERM R47      ELSE SHIFT PS.
251 000320 174 207      LRM R74      SHIFT PQ.
252 000322 160 034 341      PCMD R60+R34  GET NEXT LN CONSTANT.
253 000325 136 211      ICM R36      ADJUST EXP(LN(XM)).
254 000327 235      CLE      LEAD DIGIT = 0 INITIALLY.
255 000330 060      ARP 60      SET ARP FOR TIGHT LOOP.
256 000331 360 353      JMP LN94      JMP TO TIGHT LOOP.
257 000333 140 221      HEWR  TSM R40      R=0?
258 000335 367 036      JZR HEWR2      JIF R=0; R/(1-.5R)=0.
259 000337 012 345      PUMD R40+R12  ELSE TEMP SAVE M(R).
259 000341 136 034 243      STM R36,R34  TEMP SAVE EXP(R).

```

```

>>>>      MAIN LN(X) SUBROUTINE      <<<<<<<<
000344 147 207      LRM R47      PROTECT FOR CARRY.
261 000346 140 050 243      STM R40,R50      READY TO FORM SR.
262 000351 040 303      ADN R40,R40      2R.
263 000353 303      ADN R40,R40      4R.
264 000354 050 303      ADN R40,R50      5R.
265 000356 316 377 377      JSR= BRTS35      SHIFT 5R RIGHT.
266 000361 150 012 343      PCMD R50-R12      GET M(R).
267 000364 140 215      TCM R40      M(1-.5R).
268 000366 766 006      JNZ LN84      JIF 5R NOT=0 AFTER SHIFTING.
269 000370 050 241      LDM R40,R50      ELSE ANSWER=R.
270 000372 136 034 241      LDM R36,R34      RESTORE EXP(R).
271 000375 236      HEWR2      RTN      RTN WITH ANSWER=R.
272 000376 136 213      LN84      DCM R36      EXP(1-.5R)=-1.
273 000400 360 017      JMF DIV17      NEW R=R/(1-.5R) AND EXIT.
274 000402 371 004      LN98      JEZ LN99      JIF MSD=0 IN E.
275 000404 147 203      ERM R47      ELSE SHIFT IT INTO R40.
276 000406 136 211      ICM R36      ADJUST EXP.
277 000410 360 130      LN99      JMP SHF10      SHIFT OUT LEAD 0'S & EXIT.
078 000412
< 000412
280 000412 020 055      ***** LN E ATTRIBUTES TABLE *****
281 000414      OCT 20,55
*****
282 000414 316 000 000 LN5      JSR =LN20      GET LN(X)
283 000417 760 074      JMF DIV11      DO RDNF

```

```

>>>>          DIVIDE AND SHIFT ROUTINES          <<<<<<<
285 000421 136 215      DIV17  TCM R36      EXP(ANS)=EXP(A)+(EXP(B)),UTI.
286 000423 235          CLE                LEFT DIGIT OF A=0.
287 000424 150 040 301  CMM R50,R40      IS A>=B?
288 000427 373 003      JCY DIV77        JIF A>=B.
289 000431 205          LLM R50          ELSE SHIFT LEFT DIG OF A TO E.
290 000432 134 213      DCM R34          ADJUST EXP DUE TO SHIFT.
291 000434 130 250 025 DIV77  LDB R30=150  COUNTER (16 DIGIT MANTISSAS).
292 000437 131 222      CL3 R31          READY FOR LEFT DIGIT OF A.
293 000441 200          EL3 R31          LEFT DIGIT OF A TO R31, 0 TO E.
294 000442 316 125 010      JSB=UT110    CALC SGN,EXP,INIT R40 AND R60.
295 000445 260 004          JMP DIV79    GO DO DIVIDE.
296 000447 150 205      DIV30  LLM R50      SHIFT R; LEFT DIGIT TO E.
297 000451 131 200      EL3 R31          LEFT DIG OF R TO R31, 0 TO E.
298 000453 150          DIV79  DRP 50      DRP READY FOR TIGHT LOOP.
299 000454 260 004          JMP DIV50    JMP TO TIGHT LOOP.
300 000456 131 212      DIV39  DCB R31      BORROW 1 FROM LEFT DIGIT OF R.
301 000460 150          DRP 50      DRP READY FOR TIGHT LOOP.
302 000461 234          DIV40  ICE          INCREMENT FOR EACH SUBTRACT.
303 000462 305          DIV50  SSM R#,R#    CAST OUT MULTIPLES OF DENOM.
304 000463 373 374      JCY DIV40        CONTINUE TILL DONE.
305 000465 131 220      TSB R31          IS LEFT DIGIT OF R=0?
306 000467 266 265      JNZ DIV39        LOOP TILL = 0.
307 000471 150 303      ADM R50,R#      ELSE OVERDRAFT; RESTORE R.
308 000473 140 201      ELM R40          SAVE NEW DIGIT OF ANSWER.
309 000475 130 212      DCB R30          DECREMENT DECADE COUNTER.
310 000477 265 346      JPS DIV30        CONTINUE IF R33>=0.
311 000501 236          DIV99  RTH          END.
312 000502          ****  DIVIDE ATTRIBUTES TABLE  ****
313 000502 012 051          OCT 12,51
314 000504          ****
315 000504 316 066 012 DIV2  JSB= TUOR      GET A AND B (REAL) FOR A/B.
316 000507 316 076 010 DIV10 JSB= SEP15     SEPARATE A AND B.
317 000512 316 121 001 DIV14 JSB= DIV88     GET UNPACKED QUO IN 32,36,40.
318 000515 104 251 340 DIV11 GTD R0NF      ROUND, NFR, AND EXIT.
319 000521 140 221          DIV88  TSM R40      TEST FOR 0 DENOMINATOR.
320 000523 266 012          JNZ DIV20      GO DO DIVIDE IF NO.
321 000525 316 377 377      JSB= ERROR    GO FLAG DIVIDE BY 0 OCCURRED.
322 000530 010          OCT 80          CORRESPONDING ERROR NO.
323 000531 133 206          DIV89  LRB R33      SGN TO E.
324 000533 104 251 241      GTD FTR88     DIRECT RETURN WITH OVF VALUE.
324 000536 007
325 000537 316 021 001 DIV20 JSB= DIV17     GO DO A/B.
326 000542 140 221          SHF10  TSM R40      TEST MANTISSA.
327 000544 266 010          JNZ SHF30     IF M#0 GO SHIFT OFF LEAD 0'S.
328 000546 104 251 316      GTD LCOM1     SET SGN=EXP=0 AND EXIT ROUTINE.
328 000551 004
329 000552 136 213          SHF20  DCM R36      ADJUST EXPONENT.
330 000554 140 205          LLM R40      SHIFT OUT LEAD ZERO.
331 000556 374 372          SHF30  JLZ SHF20   KEEP LOOPING TILL NO LEAD 0'S.
332 000560 236          RTH          END ROUTINE.

```

```

CARMA 03/14/81=====
ITEM   LOC   OBJECT CODE SRC=CCHRXN OBJ=GENHRX  9/11/1981 11:40 AM PG 9
=====
>>>> LOG BASE 10, MOD/REM
000561          *** LOG 10 ATTRIBUTES TABLE ***
335 000561 020 055          OCT 20,55
336 000563          *****
337 000563 316 000 000 LOGT5 JSR= LN20      FIND LN(X).
338 000566 316 306 004          JSR= LOGCOM   GO GET READY FOR DIVIDE.
339 000571 140 261 241          LDND R40=TBL2 GET MANT(LN(10)).
339 000574 011
340 000575 260 313          JMP DIV14      LOG(X), ROUND, NFR & EXIT.
341 000577          * MODULO/REMAINDER ROUTINES
342 000577          *** REM ATTRIBUTES TABLE ***
343 000577 040 055          OCT 40,55
344 000601          *****
345 000601 102 222          REH10 CL3 R2      FLAG FOR IP(X) (INTEGER PART).
346 000603 260 005          JMP MOD20      GO CALC RMD(A,B)=A-B(IP(A/B)).
347 000605          *** MOD ATTRIBUTES TABLE ***
348 000605 012 051          OCT 12,51
349 000607          *****
350 000607 102 222          MOD10 CL3 R2      CLEAR FLAG AREA.
351 000611 210          ICB R2      FLAG=0 FOR INT(X) (FLOOR(X)).
352 000612 316 066 012 MOD20 JSR= TUOR      REAL B TO R40, REAL A TO R50.
353 000615 316 141 010          JSR= UT210    B TO R50, A TO R40.
354 000620 150 221          TSM R50      B=0?
355 000622 266 061          JNZ MOD35      JIF B#0.
356 000624 140 012 345 MOD30 PUMD R40+R12  PUSH ANS=A ONTO STACK.
357 000627 236          RTH          EXIT WITH ANS ON STACK.
358 000630 104 251 343 MOD40 GTO NFR      PACK ANS=A AND PUT ON STACK.
359 000633 005
360 000634 147 207          MOD50 LRM R47      GENERATE CARRY AREA.
361 000636 157 207          LRM R57      ALIGN M(B).
362 000640 140 050 305 MOD60 SBM R40,R50  CAST OUT MULTIPLES OF B.
363 000643 373 373          JCY MOD60      LOOP TILL OVERDRAFT.
364 000645 203          ADM R40,R50      RECOVER FROM OVERDRAFT.
365 000646 205          LLM R40      FOR NEXT LOWER DECADE.
366 000647 130 213          DCM R30      DECREMENT COUNTER.
367 000651 365 365          JPS MOD60      LOOP TILL ALL DECADES DONE.
368 000653 136 034 241          LDM R36,R34  EXP(ANS)=EXP(B).
369 000656 102 220          TSB R2      DOING REMAINDER FUNCTION?
370 000660 367 017          JZR MOD70      JIF YES, ALREADY HAVE RMD ANS.
371 000662 133 032 300          CMB R33,R32  SGN(A)=SGN(B)?
372 000665 376 012          JRZ MOD70      JIF YES, ALREADY HAVE MOD ANS.
373 000667 242          STB R33,R32  SGN(ANS)=SGN(B).
374 000670 140 221          TSM R40      A-K*B=0 EXACTLY?
375 000672 367 005          JZR MOD70      ANS=0 IF R40=0.
376 000674 150 205          LLM R50      ELSE SHIFT B TO ALIGN WITH R40.
377 000676 040 305          SBM R50,R40      GET ANSWER.
378 000700 243          STM R50,R40      MOVE M(ANS).
379 000701 104 251 335 MOD70 GTO SHRONF  SHIFT OFF LEAD 0'S OF ANS, DONE.
380 000704 005
381 000705 140 221          MOD35 TSM R40      A=0?
382 000707 367 313          JZR MOD30      JIF A=0 (THEN ANS=A=0).
383 000711 316 076 010          JSB= SEP15    SEPARATE A AND B.
384 000714 130 036 241          LDM R30,R36  DUPLICATE EXP(A).
385 000717 034 305          SBM R30,R34  EXP(A)-EXP(B).

```

>>>>>		LOG BASE 10, MOD/REM		<<<<<<<	
384	000721 365 311	JPS MOD50	JIF EXP(A)>=EXP(B).		
385	000723 102 220	TSB R2	DOING REMAINDER FUNCTION?		
386	000725 367 301	JZR MOD40	JIF YES (THEN RMD(A,B)=A).		
387	000727 132 033 300	CMB R32,R33	SGN(A)=SGN(B)?		
388	000732 376 274	JRZ MOD40	JIF YES (THEN A MOD B = A).		
389	000734 360 143	JMF ADD15	ELSE A MOD B = A+B.		

```

ARMA 03/14/81=====
ITEM   LOC   OBJECT CODE SRC=CCHRRX OBJ=GENHRX  9/11/1981 11:40 AM PG 11
=====
>>>>> REAL/INTEGER CHS, SUB, ADD                                     <<<<<<<
000736      **** MONADIC - ATTRIBUTES TABLE *****
392 000736 007 050      OCT 7,50
393 000740
394 000740 140 012 343 CHSROI PCMD R40,-R12 GET VALUE
395 000743 144 310 377      CMS R44,=377 IS IT REAL?
396 000746 373 014      JCY CHS10X JIF NO
397 000750 140 221      CHS10 TSM R40 IS X=0?
398 000752 367 004      JZR CHS11 DO NOTHING IF NUMBER IS ZERO.
399 000754 141 204      LL3 R41 SAVE XS IN E.
400 000756 216      NCB CHANGE SIGN (0 TO 9 OR 9 TO 0).
401 000757 202      ER3 RESTORE XS.
402 000760 140 012 345 CHS11 PUMD 40,+12 PUSH ANSWER ON STACK & EXIT.
403 000763 236      RTN
404 000764 145 215      CHS10X TCM R45 TEN'S COMP
405 000766 360 370      JMP CHS11 PUSH IT OUT & EXIT
406 000770      **** SUB ATTRIBUTES TABLE *****
407 000770 007 051      OCT 7,51
408 000772      *****
409 000772 316 071 010 SUBROI JSB =TUOSEP GET 2 REALS OR INTS
410 000775 316 025 002 SPROI+ JSB= SUBIL TO SUB INNER LOOP.
411 001000 360 017      JMP ADSUEX EXIT TO PUSH RESULT.
412 001002 316 076 010 SUB10 JSB =SEP15 SEPARATE THE MOTHER
413 001005 132 216      SUB11 NCB R32 CHANGE SIGN OF B
414 001007 360 070      JMP ADD15 NOW ADD
415 001011
416 001011      **** ADD ATTRIBUTES TABLE *****
417 001011 007 051      OCT 7,51
418 001013      *****
419 001013 316 071 010 ADDROI JSB =TUOSEP GET 2 REALS OR INTS
420 001016 316 033 002 ADROI+ JSB= ADSU TO INNER LOOP.
421 001021 104 251 172 ADSUEX GTD MPYEX EXIT TO PUSH RESULT.
422 001024 003
423 001025 132 216      SUBIL NCB R32 CHS OF B IN CASE REAL.
424 001027 371 055      JEZ ADD20 JIF BOTH REALS.
425 001031 145 215      TCM R45 ELSE COMPLEMENT B.
426 001033 371 051      ADSU JEZ ADD20 JIF BOTH REALS.
427 001035 145 055 303 ADRI10 ADM R45,R55 A+B
428 001040 374 030      JLZ ADD8 JIF SUM OK.
429 001042 147 310 220      CMS R47= 90C GOOD SUM IF LEFT DIG=9.
430 001045 374 023      JLZ ADD8 JIF SUM OKAY NEG INTEGER.
431 001047 132 222      CL3 R32 SIGN POS.
432 001051 147 310 120      CMS R47=50C TEST SIGN.
433 001054 372 004      JNC POSGN JIF SIGN IS POS.
434 001056 145 215      TCM R45 MAKE MANTISSA POS.
435 001060 132 216      NCB R32 MAKE IT'S SIGN NEG.
436 001062 153 223      POSGN CLM R53 NEED TO CLEAR R40 TAIL.
437 001064 040 243      STM R53,R40 CLEAR R40 TAIL.
438 001066 136 251 005      LDM R36=5C,0 EXP ANS=5.
439 001071 000
440 001072 236      ADD3 RTN END.
441 001073 316 066 012 ADD9 JSB= TUOR DEMAND TWO REALS.
442 001076      +ADD10 IS ONLY USED BY INTERNAL "ODDBALL" CALLS
443 001076 316 076 010 ADD10 JSB =SEP15 SEPARATE OUT FOR ODDBALL CALLS

```

```

>>>>> REAL/INTEGER CHS, SUB, ADD
400 001101 316 106 002 ADD15 JSB =ADD20 DO THE ADD
443 001104 360 142 JMF EXP11 DO THE ROUNF THING
444 001106 150 221 ADD20 TSM 50 IS A = 0?
445 001110 367 014 JZR OPND=0 JIF YES
446 001112 * IF THE NUMBER IN R50 IS ZERO, DO NOTHING!
447 001112 140 221 TSM R40 IS B = 0?
448 001114 366 011 JNZ A&B#0 JIF NO
449 001116 * IF R40 IS ZERO, R50 IS YOUR ANSWER!
450 001116 050 241 LDM R40,R50 MANTISSA TO OUTPUT
451 001120 136 034 241 LDM R36,R34 MOVE EXPONENT
452 001123 132 033 240 LDB R32,R33 MOVE SIGN
453 001126 236 OPND=0 RTH DONE IF ONE OPND IS ZERO
454 001127 134 036 305 A&B#0 SSM R34,R36 T=EXP(A)-EXP(B).
455 001132 367 005 JZR EXP= JIF EQUAL EXPONENTS
456 001134 365 010 JPS SWAP+- JIF IN WRONG ORDER
457 001136 215 TCM R34 CHANGE COUNTER TO >0
458 001137 360 025 JMF ADDNOW EVERYTHING COOL NOW
459 001141 * EXPONENTS SAME, SEE IF LARGER MANT IS IN R40
460 001141 140 050 301 EXP= CMM R40,R50 SEE WHICH MANTISSA IS BIGGER
461 001144 373 020 JCY ADDNOW JIF ALREADY CORRECT
462 001146 160 040 241 SWAP+- LDM R60,R40 SWAP MANTISSAE
463 001151 150 243 STH R50,R40 THIS COSTS 3 BYTES EXTRA
464 001153 060 241 LDM R50,R60 (JSB =UT210 DOES IT ALSO)
465 001155 132 242 STB R32,R60 SWAP SIGNS THRU R60
466 001157 033 240 LDB R32,R33 R33 TO R32
467 001161 160 242 STB R60,R33 R60 TO R33
468 001163 136 034 303 ADM R36,R34 LARGER EXP, EXP(A), TO R36.
469 001166 133 032 226 ADDNOW XRS R33,R32 =0 IF SGN(A)=SGN(B), ELSE=9.
470 001171 204 LL3 R33 +/- INDICATOR & LEAD 0/9 INCR.
471 001172 367 002 JZR ADD40 JMP IF SIGNS THE SAME.
472 001174 150 215 TCM R50 OPP SGNS; NEGATE MANT IN R50.
473 001176 134 311 026 ADD40 CMM R34=160,0 EXP DIFF>=16?
473 001201 000
474 001202 033 ARP 33 SET ARP FOR MANT ALIGN LOOP.
475 001203 372 007 JNC ADD50 JMP IF <16 SHIFTS OF SMALL M.
476 001205 104 251 141 ADD95 GTO SHF10 GO ELIM LEAD ZEROS AND END.
476 001210 001
477 001211
478 001211 157 207 ADD52 LRM R57 SHIFT RT SMALL MANT TO ALIGN.
479 001213 302 AD3 R57,R# GIVES LEAD 9'S OR LEAD 0'S.
480 001214 134 212 ADD50 DCB R34 BEGIN LOOP TO SHIFT SMALL M.
481 001216 373 371 JCY ADD52 CONTINUE LOOP TILL SHIFTS DONE.
482 001220 140 050 303 ADM R40,R50 M(LG)+-M(SM).OMPL OF OTHER.
483 001223 *FOR SUBTRACT, CARRY IS ALWAYS 1
484 001223 372 360 JNC ADD95 IF CARRY
485 001225 133 220 TSB R33 IS THIS A SUBTRACT?
486 001227 366 354 JNZ ADD95 JIF YES, DONE
487 001231 235 CLE CREATE A 1 IN E
488 001232 234 ICE AND
489 001233 147 203 ERM R47 PUT AS 1ST DIGIT OF ANS
490 001235 136 211 ICM R36 FIX EXP
491 001237 236 RTH DONE
492 001240 * EXP(X) ENTRY

```

```

ARMA 03/14/81=====
ITEM   LOC   OBJECT CODE SRC=COHRXX OBJ=GENHRX  9/11/1981 11:40 AM PG 13
=====
>>>>  REAL/INTEGER CHS, SUB, ADD                                <<<<<<<<
, 001240          *** EXP ATTRIBUTES TABLE ***
494 001240 020 055          OCT 20,55
495 001242          ****
496 001242 316 045 012 EXP3  JSB= ONER   GET REAL X.
497 001245 316 246 010      JSB= EXP18   GO CALCULATE EXP(X).
498 001250 360 054      EXP11  JMF SQR11   ROUND, NFR, AND EXIT.

```

```

ARMA 03/14/81=====
ITEM LOC OBJECT CODE SRC=CCRRXX OBJ=GEZHRX 9/11/1981 11:40 AM PG 14
=====
>>>> SQUARE ROOT ROUTINE <<<<<<<
500 001252 140 070 303 SQR65 ADM R40,R70 INCREMENT ANSWER DIGIT.
501 001255 150 040 305 SQR45 SBM R50,R40 GENERATE NEW REMAINDER.
502 001260 373 370 JCY SQR65 CONTINUE UNTIL OVERDRAFT.
503 001262 303 ADM R50,R40 RECOVER FROM OVERDRAFT.
504 001263 205 LLM R50 FOR NEXT DECADE.
505 001264 140 060 305 SBM R40,R60 ANNIHILATE OLD 5 CONSTANT.
506 001267 177 207 LRM R77 NEW CONST 0...1...0 INCREMENT.
507 001271 167 207 LRM R67 NEW CONST 0...5...0 INCREMENT.
508 001273 130 212 DCB R30 DECREMENT DECADE COUNTER.
509 001275 373 121 JCY SQR40 QUIT AFTER 15 DIGITS GENERATED.
510 001277 140 205 LLM R40 KILL LEADING ZERO.
511 001301 760 302 SQR99 JMP ADD95 SHIFT OUT LEAD 0'S & EXIT.
512 001303 *** SQR ATTRIBUTES TABLE *****
513 001303 020 055 OCT 20,55
514 001305 *****
515 001305 316 045 012 SQR5 JSB= ONER GET REAL X.
516 001310 141 220 TSB R41 NEGATIVE INPUT?
517 001312 376 004 JRZ SQR15 IF NOT, JUMP AND CONTINUE.
518 001314 316 377 377 JSB= ERROR+ ERROR; SQR OF NEG NO.
519 001317 012 OCT 100 CORRESPONDING ERROR NO.
520 001320 316 102 010 SQR15 JSB= SEP10 UNPACK X.
521 001323 316 332 002 JSB= SQR30 GO DO SQUARE ROOT.
522 001326 104 251 340 SQR11 GTD R0NF ROUND, NFR, AND EXIT.
522 001331 005
523 001332 147 207 SQR30 LRM R47 5M NEEDS CARRY AREA.
524 001334 140 050 243 STM R40,R50 M.
525 001337 150 303 ADM R50,R50 2M.
526 001341 303 ADM R50,R50 4M.
527 001342 040 303 ADM R50,R40 5M IN R50.
528 001344 134 036 241 LDM R34,R36 EXP.
529 001347 362 002 JOD SQR35 JIF EXP IS ODD.
530 001351 157 207 LRM R57 5M DECIMAL ALIGNMENT.
531 001353 150 221 SQR35 TSM R50 IS 5M=0?
532 001355 766 003 JNE SQR36 JIF NOT 0.
533 001357 140 223 CLM R40 ELSE ANS=0.
534 001361 236 RTH END.
535 001362 136 036 303 SQR36 ADM R36,R36 2 EXP.
536 001365 303 ADM R36,R36 4 EXP.
537 001366 034 303 ADM R36,R34 5 EXP IN R34.
538 001370 137 207 LRM R37 EXP(ANS)=INT(5EXP/10).
539 001372 135 310 120 CMB R35=500 LEAD DIGIT FOR EXP=0 IF EXP>0.
540 001375 372 003 JNC SQR37 JMP IF EXP>=0.
541 001377 137 312 220 ADB R37=900 GENERATE LEAD DIGIT=9.
542 001402 130 250 024 SQR37 LDB R30=140 SET UP DECADE COUNTER.
543 001405 170 223 CLM R70 READY FOR INCREMENTING CONST 1.
544 001407 177 210 ICB R77 GENERATE THE CONSTANT 1.
545 001411 160 223 CLM R60 CLEAR CONSTANT AREA.
546 001413 166 250 120 LDB R66=500 LOAD CONSTANT 5.
547 001416 140 223 CLM R40 INITIALIZE ANSWER ACCUMULATOR.
548 001420 140 060 303 SQR40 ADM R40,R60 ADD CONSTANT.
549 001423 360 230 JMP SQR45 GO GENERATE RT. DIGIT OF BYTE.

```

```

>>>>      MULTIPLY. INTEGER OR REAL      <<<<<<<<
001425 316 032 003 MPY10  JSB =MPY28      GO TO MULTIPLY INNER LOOP
552 001430 360 274      MPY11  JMP SQR11      DO ROUNF
553 001432 316 076 010 MPY28  JSB= SEP15      GO UNPACK A AND B.
554 001435 316 131 010 MPY30  JSB= UT110F      CALC SQN, EXP.
555 001440 100 250 057      LD3 R0,=57      INITIALIZE POINTER.
556 001443 160 040 241      LDM R60,R40
557 001446 050 225      ORM R60,R50      TRAILING ZERO INDICATOR.
558 001450 367 076      JZR MPY36      0*0=0.
559 001452 230      BIN      MODE FOR POINTER INCREMENT.
560 001453 100 210      MPY31  ICB R0      INCREMENT POINTER.
561 001455 101 220      TSB R*      FOUND#0?
562 001457 367 372      JZR MPY31      JIF NO.
563 001461 150 060 243      STM R50,R60      ORIGINAL MULTIPLIER.
564 001464 100 314 020      SBB R0,=20      POINTER=4X.
565 001467 101 220      TSB R*      CHECK FOR NON-ZERO.
566 001471 367 004      JZR MPY32      JIF R40=MPYR ( > TRAIL 0'S).
567 001473 140 243      STM R40,R60      ELSE R50 IS THE MULTIPLIER.
568 001475 050 241      LDM R40,R50      SET UP TO ELIM TRAIL 0'S.
569 001477 150 223      MPY32  CLM R50      CLEAR UPPER DIGS OF MPYR.
001501 101 050 243      STM R*,R50      ELIM MPYR TRAIL 0'S.
571 001504 140 223      CLM R40      INITIALIZE ACCUMULATE AREA.
572 001506 060      ARP 60      SET ARP FOR TIGHT LOOP.
573 001507 231      BCD      RESET MODE FOR REST OF MPY.
574 001510 150 221      TSM R50      TRAP OUT ZERO.
575 001512 367 034      JZR MPY36      JIF ANS=0.
576 001514 157 207      MPY77  LRM R57      SHIFT OUT TRAILING 0'S OF MPY'R.
577 001516 371 374      JEZ MPY77      LOOP TILL NO TRAILING 0'S.
001520 100 222      CL3 R0      WHERE LEFTMOST DIGIT ACCUMULATES.
579 001522 140      DRP 40      READY FOR TIGHT LOOP.
580 001523 303      MPY50  ADM R#,R#      ACCUMULATE ANSWER.
581 001524 372 003      JNC MPY52      JIF NO CARRY TO LEFTMOST DIGIT.
582 001526 100 210      ICB R0      TALLY ONE IN LEFTMOST DIGIT.
583 001530 140      DRP 40      RESET DRP.
584 001531 233      MPY52  DCE      DECREMENT COUNTER.
585 001532 370 367      MPY60  JEH MPY50      KEEP ACCUMULATING TILL E=0.
586 001534 100 206      LRB R0      LEFT DIGIT TO E, 0 TO R0.
587 001536 150 221      TSM R50      SEE IF=0 YET.
588 001540 366 012      JNZ MPY70      CONTINUE IF#0.
001542 371 004      JEZ MPY36      DONE IF LEFT DIGIT=0.
590 001544 147 203      ERM R47      GET LEFT DIGIT FROM E.
591 001546 136 211      ICM R36      ADJUST EXP DUE TO ABOVE SHIFT.
592 001550 104 251 141 MPY36  GTD SHF10      SHIFT OUT LEAD 0'S AND EXIT.
592 001553 001
593 001554 147 203      MPY70  ERM R47      SHIFT, READY FOR NEXT DECADE.
594 001556 157 207      LRM R57      NEXT MULTIPLIER DIGIT TO E.
595 001560 140      DRP 40      RESET DRP.
596 001561 360 347      JMP MPY60      GO TO TIGHT LOOP.
597 001563      **** MPY ATTRIBUTES TABLE *****
598 001563 012 051      OCT 12,51
599 001565      *****
600 001565 316 071 010 MPYROI JSB= TUOSEP 2 INTS OR 2 REALS.
601 001570 316 204 003      JSB= MPYR      GO MULTIPLY THEM.
600001573 144 310 377 MPYEX  CMB R44=377  INTEGER RESULT?

```

```

CARMA 03/14/81=====
ITEM   LOC   OBJECT CODE SRC=COHRXX OBJ=GENHRX  9/11/1981 11:40 AM PG 16
=====
>>>>  MULTIPLY, INTEGER OR REAL                                     <<<<<<<
600 001576 372 230 JNC MPY11 JIF REAL.
604 001600 140 012 345 PUMD R40+R12 ELSE PUSH INT ON STK.
605 001603 236 RTN EXIT.
606 001604 371 227 MPYR JEZ MPY30 JIF REALS.
607 001606 100 222 CL3 R0 CLEAR SIGN FLAG.
608 001610 157 220 TSB R57 SEE IF NEG.
609 001612 265 004 JPS SGN2 JIF POS.
610 001614 155 215 TCM R55 MAKE POS.
611 001616 100 210 ICB R0 FLAG 1 SIGN.
612 001620 147 220 SGN2 TSB R47 SEE IF OTHER NO. NEG.
613 001622 265 004 JPS SGN3 JIF POS.
614 001624 145 215 TCM R45 MAKE POS.
615 001626 100 210 ICB R0 FLAG 1 SIGN.
616 001630 145 055 301 SGN3 CMM R45,R55 WHICH IS LARGEST?
617 001633 373 003 JCY NOFLIP JIF 40>=50.
618 001635 316 141 010 JSB= UT210 SMALLEST TO R50.
619 001640 170 223 NOFLIP CLM R70 READY FOR MULTIPLICAND.
620 001642 145 070 243 STM R45,R70 MULTIPLICAND.
621 001645 140 223 CLM R40 CLEAR ACCUMULATOR.
622 001647 155 050 243 STM R55,R50 RT JUSTIFY MULTIPLIER IN R50.
623 001652 267 045 JZR POS TRAP OUT ZERO.
624 001654 153 223 CLM R53 CLEAR LEFT END R50 GARBAGE.
625 001656 152 207 MPYLOP LRM R52 NEXT MPYR DIGIT TO E.
626 001660 140 DRP 40 FOR TIGHT LOOP.
627 001661 070 ARP 70 FOR TIGHT LOOP.
628 001662 260 002 JMP TSTEZ GO MPY BY CURRENT DIGIT IN E.
629 001664 303 ADLOOP ADL R#,R# ACCUMULATE ANSWER.
630 001665 233 DCE MULTIPLIER DIGIT DECREMENT.
631 001666 370 374 TSTEZ JEN ADLOOP LOOP TILL E=0.
632 001670 170 205 LLM R70 FOR NEXT DECADE ALIGNMENT.
633 001672 150 221 TSM R50 ANY MORE#0 MULTIPLIER DIGITS?
634 001674 266 360 JNZ MPYLOP LOOP TILL MULTIPLIER DEPLETED.
635 001676 170 223 CLM R70 READY FOR TEST CONSTANT.
636 001700 172 250 020 LDB R72=10C MAX INTEGER TEST CONSTANT.
637 001703 142 072 301 CMM R42,R72 ANS TOO BIG IN ABS VALUE?
638 001706 373 015 JCY OOPS JIF ABS(ANS)>>99999.
639 001710 145 040 241 LDM R45,R40 ANSWER TO UPPER PART OF R40.
640 001713 100 220 TSB R0 TEST SGN FLAG FOR POS/NEG ANS.
641 001715 263 002 JEV POS JIF ANS POS.
642 001717 145 215 TCM R45 ELSE MAKE ANS NEGATIVE.
643 001721 144 250 377 POS LDB R44=377 INTEGER TAG.
644 001724 236 RTN END.
645 001725 132 222 OOPS CL3 R32 SIGN POS.
646 001727 100 220 TSB R0 TEST SIGN.
647 001731 263 002 JEV OOPS1 JIF POS.
648 001733 132 216 NCB R32 SIGN NEG.
649 001735 104 251 311 OOPS1 GT0 INFR9 EXP=15, SHF10.
649 001740 005

```

```

>>>>>  INTEGER MULTIPLY INTM AND INTMUL                                     <<<<<<<
001741  *****
652 001741  * ENTRY: R66 is multiplier
653 001741  *       R76 is multiplicand
654 001741  * EXIT:  R54 is the result - 4 bytes long
655 001741  *       Note that the answer is a full 32 bit number
656 001741  *       and that the sign bit may be set!!
657 001741  *****
658 001741 232  INTMUL SAG          SAVE BIN BCD
659 001742 230  BIN          BUT I NEED BINARY
660 001743 160 006 345  PUMD R60+R6  SAVE FOR HOMER,
661 001746 174 345  PUMD R74+R6  SAVE FOR HOMER,
662 001750 154 223  CLM R54      CLEAR ACCUMULATOR,
663 001752 176 074 243  STM R76,R74  MOVE B DOWN TO R74,
664 001755  **          CMM R76,R66  NEED TO SWAP?
665 001755  **          JCY NOSWPM   JIF NO R76>R66
666 001755  **WE WANT R66 SMALL  SINCE IT IS SHIFTED UNTIL 0
667 001755  **          STM R66,R74  NEW VALUE IN 74
668 001755  **          LDM R66,R76  OLD 76 TO 66
669 001755  +NOSWPM CLM R76        NOW ZERO MOST SIG BYTES,
001755 223  CLM R76          NOW ZERO MOST SIG BYTES.
671 001756 166 060 243  STM R66,R60  MAKE OPERAND 2 BYTES
672 001761 367 013  MLOOP  JZR INTM2  JIF MULTIPLIER NOW ZERO
673 001763 363 003  JEV DONTAD  JIF NO ADD THIS TIME,
674 001765 154 074 303  ADM R54,R74  ADD B TO ACCUM,
675 001770 174 205  DONTAD  LLM R74   DOUBLE R74,
676 001772 161 207  LRM R61        MOVE R60 ONE BIT,
677 001774 360 363  JMF MLOOP      CONTINUE THE LOOP
678 001776 174 006 343 INTM2  POMD R74,-R6  RESTORE R74
679 002001 160 343  POMD R60,-R6  RESTORE R60
680 002003 237  PAD          RESTORE BCD/BIN
681 002004 236  RTH          DONE

```

RANDOM NUMBER GENERATOR ROUTINE

<<<<<<<

002005			**** RND ATTRIBUTES TABLE *****
002005	000	055	OCT 0,55
002007			*****
002007	147	222	RND10 CL3 R47 IN CASE 0000X
002011	150	223	CLM R50 ZERO OUT ACCUMULATOR.
002013	170	261 002	LCMD R70=X(K-1) GET LAST RANDOM NUMBER.
002016	204		
002017	177	207	LRM R77 WANT LAST RN RIGHT JUSTIFIED.
002021	160	251 147	LDM R60=67C,84C,92C,30C,11C,85C,02C,00C
002024	204	222 060	
002027	021	205 002	
002032	000		
002033			* LOAD MULTIPLIER A.
002033	136	251 025	LDM R36=15C,0C MULTIPLY LOOP COUNTER.
002036	000		
002037	970		ARP 70 SET ARP.
002040	167	207	RND40 LRM R67 GET MULTIPLIER DIGIT.
002042	150		DRP 50 SET DRP.
002043	260	002	JMP RND31 JUMP TO TIGHT LOOP.
002045	233		RND30 DCE DECREMENT MULTIPLIER DIGIT.
002046	303		ADM R#,R# ADD R70 TO R50.
002047	370	374	RND31 JEH RND30 LOOP TILL E=0.
002051	157	207	LRM R57 SHIFT NEW RT DIG OF PROD TO E.
002053	147	203	ERM R47 SAVE NEW DIGIT OF PROD IN R40.
002055	136	212	DCB R36 DECREMENT COUNTER.
002057	266	357	JNE RND40 CONTINUE MULTIPLY LOOP.
002061	140	263 002	STMD R40=X(K-1) SAVE NEW RANDOM NUMBER.
002064	204		
002065	136	213	DCM R36 EXP(RN)=-1.
002067	316	142 001	JSE= SHF10 ELIMINATE LEAD ZEROS.
002072	137	204	LL3 R37 ADJUST EXP FOR PACKING.
002074	136	040 243	STM R36,R40 PACK IN EXP.
002077	140	012 345	PUMD R40,+12 PUSH ANSWER ON STACK & EXIT.
002102	236		RTM DONE

Y-X ROUTINE

002103				****	Y-X ATTRIBUTES TABLE	*****
002103	014	051			OCT 14,51	
002105				*****		*****
002105	316	066	012	YTX5	JSB= TUOR	GET X AND Y (REAL).
002110	316	141	010		JSB= UT210	SWAP INPUTS AROUND.
002113	150	263	331		STND R50=TEMP2	SAVE X.
002116	203					
002117	316	076	010		JSB= SEP15	SEPARATE SGN,MANT,EXP(Y).
002122	142	221			TSM R42	TEST FOR 0 TO X POWER.
002124	367	065			JZR YTX12	JMP IF Y=0.
002126	132	220			TSB R32	TEST FOR NEG Y.
002130	376	027			JRZ YTX14	JMP IF Y POS.
002132	135	220			TSB R35	TEST FOR NEG EXP(X).
002134	375	104			JLN YTX16	JMP IF NEG EXP(X).
002136	360	005			JMP YTX18	JMP INTO X INTEGER TEST LOOP.
002140	134	221		YTX20	TSM R34	EXP(X) REDUCED TO ZERO?
002142	367	076			JZR YTX16	JMP IF YES.
002144	213				DCM R34	OTHERWISE DECREMENT EXP(X).
002145	152	205		YTX18	LL1 R52	SHIFT MANT(X).
002147	366	367			JNZ YTX20	CONTINUE LOOP IF RESIDUAL#0.
002151	134	221			TSM R34	TEST X FOR EVEN OR ODD.
002153	366	004			JNZ YTX14	JMP IF X EVEN.
002155	157	200			EL3 R57	GET LAST DIGIT OF M(X) FROM E.
002157	362	002			JOD YTX22	JMP IF MANT(X) ODD.
002161	132	222		YTX14	CL3 R32	ZERO FOR POSITIVE ANSWER.
002163	132	022	242	YTX22	STB R32,R22	SAVE SGN(ANS).
002166	316	076	000		JSB= LN30	CALCULATE LN(Y).
002171	150	261	331		LOMD R50=TEMP2	GET X.
002174	203					
002175	316	047	010		JSB= SEP20	SEPARATE IT.
002200	316	035	003		JSB= MPY30	X LN(Y).
002203	316	251	010		JSB= EXP20	Y TO X = E TO XLN(Y).
002206	132	022	240		LD3 R32,R22	SGN(Y TO X) TO R32.
002211	360	162		YTX11	JMF CSEC11	ROUND, NFR, AND EXIT.
002213	152	221		YTX12	TSM R52	MANT(X)=0?
002215	366	011			JNZ YTX26	JMP IF MANT(X)#0.
002217	316	377	377		JSB= ERROR	0 TO 0 POWER ERROR.
002222	006				OCT 60	
002223	147	250	020		LD3 R47=10C	DEFAULT ANSWER=1.
002226	360	036			JMP SGN6	PUSH ANSWER ON STACK & EXIT.
002230	133	220		YTX26	TSB R33	TEST FOR X NEG.
002232	376	032			JRZ SGN6	JMP IF POS.
002234	316	377	377		JSB= ERROR	REPORT OVF IN YTX ROUTINE.
002237	005				OCT 50	CORRESPONDING ERROR NO.
002240	360	125			JMP INF10	
002242	316	377	377	YTX16	JSB= ERROR+	ERROR: NEG Y TO NON-INT PWR.
002245	011				OCT 90	CORRESPONDING ERROR CODE.

```

=====
ARMA 03/14/81=====
ITEM    LOC    OBJECT CODE  SRC=CCRRXX DB=CCRRXX  9/11/1981 11:40 AM  PG 20
=====
>>>>>      SGN, FTR ENTRY, DRCOM, & LOGCOM      <<<<<<<<
002246      ***      SGN ATTRIBUTES TABLE      *****
760 002246 020 055      OCT 20,55
761 002250      *****
762 002250 316 045 012 SGN5      JSB= ONER      GET REAL X IN R40.
763 002253 141 206      LRS R41      SAVE SIGN.
764 002255 140 221      TSM R40      TEST INPUT.
765 002257 367 005      JZR SGN6      GIVE ANS=0 IF X=0.
766 002261 223      CLM R40      ZERO OUT, READY FOR A 1.
767 002262 147 210      ICR R47      ABS(ANS)=1.
768 002264 141 201      ELM R41      POSITION THE 1 AND ADD SIGN.
769 002266 104 251 346 SGN6      GTO NFR2      PUSH ANSWER ON STACK & EXIT.
769 002271 005
770 002272      *      DRCOM SUBROUTINE
771 002272 316 045 012 DRCOM      JSB= ONER      GET REAL T.
772 002275 140 050 243      STM R40,R50      COPY IT TO R50.
773 002300 223      CLM R40      CLEAR CONSTANT AREA.
774 002301 316 047 010      JSB= SEP20      SEP T IN R50.
775 002304 360 011      JMP LCOM1      SET SGN=EXP=0 AND EXIT.
776 002306      *      LOGCOM SUBROUTINE
002306 136 034 243 LOGCOM      STM R36,R34      EXP(X) READY FOR DIVIDE.
778 002311 140 050 243      STM R40,R50      MANT(LN(X)) READY FOR DIVIDE.
779 002314 132 033 242      STB R32,R33      SGN(LN(X)) READY FOR DIVIDE.
780 002317 132 222      LCOM1      CL3 R32      SET SGN=0.
781 002321 136 223      CLM R36      SET EXP=0.
782 002323 236      LCOM2      RTH      RETURN TO MAIN ROUTINE.
783 002324      *      FORWARD TRIG ENTRY
784 002324      ***      SEC ATTRIBUTES TABLE      *****
785 002324 020 055      OCT 20,55
786 002326      *****
787 002326 126 251 000 SEC10      LDM R26=0,900 STATUS FOR COS.
787 002331 220
788 002332 316 075 006      JSB= FTR12      CALCULATE COS.
789 002335 140 221      TSM R40      COS=0?
790 002337 366 031      JNZ CSEC30      JIF#0.
791 002341 316 233 007      JSB= FTR78      ELSE EXIT TO ANS=+INFINITY.
792 002344      ***      CSEC ATTRIBUTES TABLE      *****
793 002344 020 055      OCT 20,55
794 002346      *****
002346 126 251 011 CSEC10      LDM R26=9C,900 STATUS FOR SIN.
795 002351 220
796 002352 316 075 006      JSB= FTR12      CALCULATE SIN.
797 002355 140 221      TSM R40      SIN=0?
798 002357 366 011      JNZ CSEC30      JMP IF NOT ZERO.
799 002361 316 377 377      JSB= ERROR      ELSE REPORT ANS=+INFINITY.
800 002364 003      OCT 3D      CORRESPONDING WARNING ERROR NO.
801 002365      *****
802 002365      *      Note: CSEC10 errors fall through the two bytes of
803 002365      *      INF10 attributes and then do INF10. The two
804 002365      *      bytes are both harmless ARP's
805 002365      *****
806 002365 000 055      OCT 0,55      INF ATTRIBUTES
807 002367 316 237 007 INF10      JSB =FTR77      DIRECT RETURN WITH + INF
808 002372

```

```
=====
>>>>      SGH, FTR ENTRY, DRCON, & LOGCON      <<<<<<<<
002372 316 043 007 CSEC30 JSB=FTR53      1/(SIN OR COS),
810 002375 260 037 CSEC11 JMP FTR15      ROUND, PACK, & RTN WITH ANS.
811 002377      **** COT ATTRIBUTES TABLE ****
812 002377 020 055      OCT 20,55
813 002401      ****
814 002401 126 251 011 COT10 LDM R26=9C,9C STATUS FOR COT.
814 002404 011
815 002405 260 024      JMP FTR11      CONTINUE.
816 002407      **** SIN ATTRIBUTES TABLE ****
817 002407 020 055      OCT 20,55
818 002411      ****
819 002411 126 251 011 SIN10 LDM R26=9C,90C SIN ENTRY.
819 002414 220
820 002415 260 014      JMP FTR11      CONTINUE.
821 002417      **** COS ATTRIBUTES TABLE ****
822 002417 020 055      OCT 20,55
823 002421      ****
824 002421 126 251 000 COS10 LDM R26=0,90C COS ENTRY.
824 002424 220
825 002425 260 004      JMP FTR11      CONTINUE.
826 002427      **** TAN ATTRIBUTES TABLE ****
827 002427 020 055      OCT 20,55
828 002431      ****
829 002431 126 223      TAN10 CLM R26      TAN ENTRY.
830 002433 316 075 006 FTR11 JSB= FTR12      GO CALCULATE FUNCTION VALUE.
831 002436 260 127      FTR15 JMP RAD11      ROUND, NFR, AND EXIT.
)
=====
```

```

>>>>>      PI, CEIL, OVF/UF, DEG-RAD      <<<<<<<
002440      *      PI ROUTINE
002440      ***      PI ATTRIBUTES TABLE      *****
002440      000 055      OCT 0,55
002442      *****
002442 140 251 000 PI10      LDM R40=0,0,59C,53C,26C,59C,41C,31C
002445 000 131 123
002450 046 131 101
002453 061
002454 360 123      PI11      JMP ABS6      PUSH ANSWER ON STACK & EXIT.
002456      *      CEILING ROUTINE
002456      ***      CEIL ATTRIBUTES TABLE      *****
002456 020 055      OCT 20,55
002460      *****
002460 316 340 001 CEIL10 JSB= CHSR01      GIVES -X.
002463 316 240 005      JSB= INT5      FLOOR(-X).
002466 316 340 001      JSB= CHSR01      CEIL(X) = -FLOOR(-X).
002471 236      RTN      END.
002472      *      OVERFLOW/UNDERFLOW ROUTINE
002472 316 377 377 OVF19 JSB= ERROR      GO SAY UNDERFLOW OCCURRED.
002475 001      OCT 1D      ERROR NUMBER FOR UNDERFLOW.
002476 140 221      OVF10 TSM R40      TEST FOR MANTISSA=0.
002500 367 215      JZR LCON1      IF=0, SET EXP=SGN=0 & EXIT.
002502 137 310 005      CMB R37=5C      TEST EXP SIZE.
002505 372 126      JNC INFR10      JMP(RTN) IF OK POS EXP.
002507 136 311 001      CHM R36=01C,95C      TEST EXP SIZE.
002512 225
002513 373 120      JCY INFR10      JMP(RTN) IF OK NEG EXP.
002515 140 223      CLM R40      OVF OR UF. CLEAR MANT(X).
002517 137 310 120      CMB R37=50C      TEST EXP SIZE.
002522 373 346      JCY OVF19      JMP IF UNDERFLOW.
002524 136 251 231      LDM R36=99C,4C      EXP(X)=499.
002527 004
002530 142 217      HCM R42      MANT(X)=99...90000.
002532 316 377 377      JSB= ERROR+      GO SAY OVERFLOW OCCURRED.
002535 002      OCT 2D      ERROR NUMBER FOR OVERFLOW.
002536      ***      DEGRAD ATTRIBUTES TABLE      *****
002536 020 055      OCT 20,55
002540      *****
002540      *      DEGREE - RADIAN ROUTINE
002540 316 272 004 RAD10 JSB= DRCON      COMMON SETUP WITH RAD TO DEG.
002543 136 250 002      LDB R36=2C      EXP(45)+1.
002546 147 250 105      LDB R47=45C      45 TO R40.
002551 316 137 001      JSB= DIV20      T/450.
002554 150 261 131      LDM R50=DCON1      GET PI/4.
002557 011
002560 133 222      CLB R33      SGN(PI/4)=0.
002562 134 223      CLM R34      EXP(PI/4)+1.
002564 316 035 003      JSB= MPY30      T(RAD)=T(DEG) X PI/180.
002567 360 150      RAD11 JMP R0NF      ROUND, NFR, AND EXIT.

```

AB3, FLOOR(=INT), & ANB

<<<<<<<<

```

002571      *      ABSOLUTE VALUE ROUTINE
378 002571      ***  ABS ATTRIBUTES TABLE  *****
379 002571 020 055      OCT 20,55
380 002573      *****
381 002573 316 045 012 ABS3  JSB= ONER  GET REAL X.
382 002576 141 206      LRB R41  SHIFT OUT OLD SIGN.
383 002600 204      LL3  SHIFT IN ZERO.
384 002601 260 144      ABS6  JMP HFR2  PUSH ANSWER ON STACK & EXIT.
385 002603 316 045 012 INTFRA JSB= ONER  GET REAL X.
386 002606 316 102 010 INFR3 JSB= SEP10 SEPARATE IT.
387 002611 136 034 243 INFR4 STM R36,R34 DUPLICATE EXP(X).
388 002614 150 223      CLM R50  INITIAL INT(X).
389 002616 135 220      TSB R35  ABS(X)>=1?
390 002620 375 013      JLN INFR10 JIF X<1 (INT(X)=0, FRA(X)=X).
391 002622 140 221      INFR5 TSM R40  ANYTHING LEFT TO SHIFT?
392 002624 367 007      JZR INFR10 JIF NO NON-ZERO DIGITS LEFT.
393 002626 205      LLM R40  ELSE SHIFT LEFT DIGIT TO E.
394 002627 150 201      ELM R50  THEN TO R50 FOR INT(X).
395 002631 136 213      DCM R36  DECR INT(X) DIGIT COUNT.
      002633 373 365      JCY INFR5  LOOP IF MORE INT(X) DIGITS.
397 002635 236      INFR10 RTM  RTN.
398 002636      ***  INT ATTRIBUTES TABLE  *****
399 002636 020 055      OCT 20,55
900 002640      *****
901 002640 102 222      INT5  CLB R2  CLEAR FLAG AREA.
902 002642 210      ICB R2  FLAG=1 FOR FLOOR(X) (=INT(X)).
903 002643 360 173      JMP IP4  GO FIND ANSWER.
904 002645      ***  INTDIV ATTRIBUTES TABLE  *****
905 002645 012 051      OCT 12,51
906 002647      *****
907 002647 316 066 012 INTDIV JSB= TUOR  DEMAND REAL A AND B.
908 002652 316 076 010      JSB= SEP15 SEPARATE A AND B.
909 002655 316 121 001      JSB= DIV88 16 DIGIT ANSWER FOR A/B.
910 002660 102 222      CLB R2  FLAG=0 FOR IP(X).
911 002662 360 164      JMP IP3  FIND INT(16 DIGIT SEPARATED X).
912 002664 102 222      INFR1 CLB R2  FLAG=0 SAYS FIND IP(X).
913 002666 316 211 005 INFR2 JSB= INFR4 SPLIT APART IP(X), FP(X).
914 002671 316 141 010      JSB= UT210 IP(X) TO R40, FP(X) TO R50.
      002674 150 221      TSM R50  FP(X)=0?
915 002676 367 022      JZR TRUNC JIF X=INT (REAL) ALREADY.
917 002700 102 220      TSB R2  IP(X) OR INT(X)?
918 002702 367 016      JZR TRUNC JIF IP(X).
919 002704 132 220      TSB R32  X<0?
920 002706 376 012      JRZ TRUNC JIF X>=0 (NO ANS ADJUST NEC).
921 002710 140 211      ICM R40  ELSE ANS=NEXT SMALLEST INTEGER.
922 002712 136 251 025 INFR9 LDM R36=150,0 INITIALIZE EXP(ANS).
922 002715 000
923 002716 104 251 141      GTD SHF10 ELIM LEAD 0'S, ADJ EXP & EXIT.
923 002721 001
924 002722 316 142 001 TRUNC JSB= SHF10 SHIFT OUT LEAD 0'S.
925 002725 136 034 241      LDM R36,R34 RESTORE EXP(X).
926 002730 236      RTN  IP(X) OR INT(X) IN 32,36,40.

```

```

>>>>>      FP, NFR, EPS, RAD-DEG, & IP
002731      *      FP AND NORMAL FUNCTION RETURN
002731      ***      FP ATTRIBUTES TABLE      *****
020 055      OCT 20,55
002733      *****
002733      FP5      JSB= INTFR      GEN FP(X) IN R32, R36, R40.
002736      SHRONF      JSB= SHF10      SHIFT OUT LEAD ZEROS.
002741      RDNF      JSB= RDN10      ROUND RESULT.
002744      NFR      JSB= NFR5      JSB TO NFR INNER LOOP.
002747      NFR2      DRP 40      POINT AT RESULT TO BE PUSHED.
002750      NFR3      PUMD R#R12      PUSH RESULT ON STACK.
002752      RTN      RETURN.
002753      NFR5      JSB= OVF10      CHECK FOR OVERFLOW.
002756      STM R36,R40      PACK IN EXPONENT.
002761      LRS R32      PUT SIGN IN E.
002763      EL3 R41      PACK UP SIGN.
002765      RTN      RETURN.
002766      *      EPS ROUTINE
002766      ***      EPS ATTRIBUTES TABLE      *****
000 055      OCT 0,55
002770      *****
002770      EPS10      JSB= TRC01      GET 1.
002773      LD3 R51=500      GIVES 1E-500.
002776      ICR R50      GIVES 1E-499.
003000      JMP NFR3      GO PUSH ANS ON STACK AND EXIT.
003002      ***      RADDEG ATTRIBUTES TABLE      *****
003002      OCT 20,55
003004      *****
003004      *      RADIANT - DEGREE
003004      DEG10      JSB= DRCON      COMMON SETUP WITH DEG TO RAD.
003007      ICM R34      10T.
003011      ICM R34      100T.
003012      JSB= TRC7      GET PI/2.
003015      LDM R40,R70      MOVE IT TO R40.
003020      JSB= DIY20      100T/(PI/2).
003023      STM R40,R50      100T/(PI/2).
003026      LRM R57      .1(100T/(PI/2)).
003030      SSM R40,R50      .9(100T/(PI/2))=T IN DEG.
003032      JMP SHRONF      SHIFT, ROUND, NFR, AND EXIT.
003034      ***      IP ATTRIBUTES TABLE      *****
003034      OCT 20,55
003036      *****
003036      IP5      CLB R2      FLAG=0 FOR IP(X).
003040      IP4      JSB= ONERO1      GET X.
003043      JEN NFR2      JIF BCD INTEGER ALREADY.
003045      JSB= SEP10      SEPARATE X.
003050      IP3      JSB= INFR2      ELSE FIND IP(X) OR INT(X).
003053      JSB= RDNF      PACK UP ANS AND PUT ON STACK.
003056      PCMD R60-R12      GET ANSWER.
003061      JSB= RT0IN      TRY TO CONVERT IT TO BCD INT.
003064      LD3 R64=377      INT TAG IN CASE CONVERTED OK.
003067      DRP 60      ANS PTR IN CASE CONVERTED OK.
003070      JEZ NFR3      JIF ANSWER CONVERTED TO BIN OK.
003072      DRP 70      ELSE POINT TO FLOATING ANSWER.
  
```

ARMA 03/14/81=====

ITEM	LOC	OBJECT CODE	SRC=	CHRXX	OBJ=	GEZHRX	9/11/1981	11:40 AM	PG 25
=====									
>>>>		FP, NFR, EPS, RAD-DEG, & IP							<<<<<<<<
003073	260	253		JMF	NFR3		GO	PUSH ANSWER ONTO STACK.	

```

=====
>>>>>      MAIN FORWARD TRIG SUBROUTINE      <<<<<<<
984 003075 316 045 012 FTR12 JSB= ONER      GET REAL T=THETA FROM STACK.
984 003100 316 102 010      JSB= SEP10      SEPARATE SGN,MANT,EXP(T).
985 003103 122 222      CL3 R22      INITIALIZE S4=0.
986 003105 132 220      TSB R32      T<0?
987 003107 376 011      JRZ FTR14      JMP IF THETA>=0.
988 003111 222      CL3 R32      SET T>0<REFLECT ABOUT X-AXIS>.
989 003112 126 221      TSM R26      S6=0?
990 003114 374 002      JLZ FTR19      JIF S6=0 <TAN OR COT>.
991 003116 376 002      JRZ FTR14      JIF S6=9 & S10=0 <COS>.
992 003120 122 216      FTR19      NCB R22      SIN(-T)=-SIN(T) <CSEC,TAN,COT>.
993 003122 125 260 160 FTR14 LDBD R25=ORG GET DEG,RAD,GRD INDICATOR.
993 003125 200
994 003126 365 054      JPS FTR16      JMP IF RAD.
995 003130 363 007      JEV FTR18      JMP IF DEG.
996 003132 140 060 243      STM R40,R60      T IN GRDS.
997 003135 167 207      LRM R67      .1T.
998 003137 140 305      SSM R40,R60      .9T<GIVES T IN DEGREES>.
999 003141 170 223      FTR18      CLM R70      GET READY TO LOAD CONST.
000 003143 177 250 105      LD3 R77=450      LOAD ANGLE CONSTANT 45.
000 003146 136 213      DCM R36      XX.XXX MANT(T) DEC LOC ADJUST.
002 003150 137 220      TSB R37      T<10 DEG?
003 003152 375 007      JLH FTR20      JMP IF T<10 DEG.
004 003154 136 213      DCM R36      XXX.XX MANT(T) DEC LOC ADJUST.
005 003156 373 003      JCY FTR20      JMP IF 100 DEG <= T.
006 003160 211      ICM R36      XX.XXX MANT(T) DEC LOC ADJUST.
007 003161 147 207      LRM R47      GIVES 0XX.XX MANT(T).
008 003163 170 050 243 FTR20 STM R70,R50      PI/4 OR 45 STORE.
009 003166 137 220      TSB R37      T IN 1ST QUADRANT?
010 003170 374 020      JLZ FTR24      JMP IF T NOT IN 1ST QUADRANT.
011 003172 125 220      TSB R25      DEG OR RAD MODE?
012 003174 364 117      JNG FTR32      JMP IF DEG.
013 003176 147 207      LRM R47      ADJUST DEC LOC AND EXP:
014 003200 136 211      ICM R36      RAD FORMAT=0.XX, DEG=XX.X.
015 003202 360 111      JMP FTR32      T FORMAT=X.XX RAD, XX.X DEG.
016 003204 170 261 131 FTR16 LDMD R70=DCON1 PI/4 TO R70.
016 003207 011
017 003210 360 351      JMP FTR20      CONTINUE.
018 003212 157 207      FTR24      LRM R57      CREATE ROOM FOR LEFT CARRY.
018 003214 150 050 303      ADX R50,R50      PI/2 OR 90.
020 003217 303      ADX R50,R50      PI OR 180.
021 003220 303      ADX R50,R50      2 PI OR 360 IN R50.
022 003221 235      CLE      CREATE LEAD 0.
023 003222 234      ICE      DUMMY OFFSET DUE TO NEXT INSTR.
024 003223 233      FTR17      DCE      BORROW 1 FROM LEFT DIGIT.
025 003224 140 305      FTR25      SSM R40,R#      CAST OUT 2PI/360 AT HIGH DECADE.
026 003226 373 374      JCY FTR25      LOOP TILL OVERDRAFT.
027 003230 370 371      JEH FTR17      JIF CAN BORROW 1 FROM E.
028 003232 303      ADM R40,R#      ELSE RECOVER FROM OVERDRAFT.
029 003233 205      LLM R40      PREPARE FOR NEXT LOWER DECADE.
030 003234 136 213      DCM R36      DECREMENT DECADE COUNTER.
031 003236 373 364      JCY FTR25      LOOP TILL ALL DECADES DONE.
032 003240 223      CLM R36      EXP(T)=0. X.XX RAD, XXX.X DEG.
033 003241 147 203      ERM R47      RESTORE R40.
=====

```

```

>>>>      MAIN FORWARD TRIG SUBROUTINE      <<<<<<<<
003243 150 070 241      LDM R50,R70  PI/4 OR 45 TO R50.
003246 157 207      LRM R57      RAD: CARRY SLOT. DEG: DEC ALGN.
003250 150 050 303      ADX R50,R50  PI/2 (1.57...) OR 090.00.
003253 140 050 305 FTR26  SBM R40,R50  T-(PI/2 OR 90).
003256 372 021      JNC FTR28      JMP WHEN OVERDRAFT.
003260 126 216      HCB R26      CHANGES TRIG FN TO ITS CO-FN.
003262 377 004      JRN FTR29      WAS COS OR TAN: CHS OF ANS.
003264 127 220      TSB R27      COT OR SIN?
003266 375 363      JLN FTR26      JIF NOT COT.
003270 122 216      FTR29  HCB R22      CHS OF VALUE OF ANS OF TRIG FN.
003272 360 357      JMP FTR26      CONTINUE LOOPING.
003274 140 050 303 FTR38  ADX R40,R50  RECOVER FROM OVERDRAFT.
003277 360 037      JMP FTR34      CONTINUE.
003301 140 050 303 FTR28  ADX R40,R50  RECOVER FROM OVERDRAFT.
003304 125 220      TSB R25      DEG OR RAD TEST.
003306 365 002      JPS FTR30      JMP IF RAD.
003310 140 205      LLM R40      ADJUST DEC LOCATION TO XX.X.
003312 150 070 241 FTR30  LDM R50,R70  GET CONSTANT PI/4 OR 45.
003315 137 220      FTR32  TSB R37      DOES T HAVE NEG EXP?
003317 375 017      JLN FTR34      JMP IF EXP(T)<0.
003321 125 220      TSB R25      DEG OR RAD TEST.
003323 364 002      JNG FTR36      JMP IF DEG.
003325 157 207      LRM R57      ALIGN DEC(0.7...): RAD MODE.
003327 140 050 305 FTR36  SBM R40,R50  T-(PI/4 OR 45).
003332 372 340      JNC FTR38      JMP IF T<45 ALREADY.
003334 305      SBM R40,R50  T-(PI/2 OR 90).
003335 215      TCM R40      (PI/2 OR 90)-T.
003336 126 216      HCB R26      CHANGE TO CO-FUNCTION.
003340 316 142 001 FTR34  JSB= SHF10  ELIM LEAD ZEROS.
003343 125 220      FTR40  TSB R25      DEG OR RAD TEST.
003345 365 026      JPS FTR42      JMP IF RAD.
003347 316 141 010      JSB= UT210  MANT(T) TO R50, 45 TO R40.
003352 136 034 243      STM R36,R34  MOVE EXP FOR DIVIDE.
003355 223      CLM R36      EXP(45/10).
003356 211      ICM R36      EXP(45).
003357 316 137 001      JSB= DIV20  T/45.
003362 170 261 131      LDMD R70=DCON1 PI/4 TO R70.
003365 011
003366 050 243      STM R70,R50  PI/4 TO R50.
003370 134 223      CLM R34      SET EXPONENT FOR PI/4.
003372 316 035 003      JSB= MPY30  (T/4.5) TIMES PI/4.
003375 120 251 007 FTR42  LDM R20=7C,0  INITIALIZE PQ COUNTER.
003400 000
003401 130 251 251      LDM R30=TBL3  POINTER TO TRIG CONSTANT TABLE.
003404 011
003405 134 036 241      LDM R34,R36  DUPLICATE EXPONENT.
003410 211      ICM R34      INCREMENT EXP(T).
003411 374 037      JLZ FTR62      JMP IF .1<=T<1.
003413 147 207      LRM R47      ADJUST FORMAT OF T TO 0.XX.
003415 134 211      FTR46  ICM R34      INCREMENT EXP(T).
003417 374 031      JLZ FTR62      JMP WHEN EXP(T)=0.
003421 160 030 341      POMD R60+R30  DUMMY POP TO INCR R30 BY 10.
003424 120 212      DCB R20      DECREMENT PQ COUNTER.

```

```

ARMA 03/14/81=====
ITEM   LOC   OBJECT CODE   SRC=OCHRXX OBJ=QEXHRX   9/11/1981 11:40 AM PG 28
=====
>>>>      MAIN FORWARD TRIG SUBROUTINE      <<<<<<<
1084 003426 366 365      JNZ FTR46      LOOP IF PQ CTR # 0.
1085 003430 122 032 242  STB R22,R32    SMALL ANGLE PATH; SETUP SGN(T).
1086 003433 140 205      LLM R40      ELIM LEAD ZERO CREATED ABOVE.
1087 003435 126 221      TSM R26      WHICH FN IS BEING CALCULATED?
1088 003437 376 117      JRZ FTR60      JIF TAN OR COS.
1089 003441 375 006      JLN FTR54      JIF SIN, ANS=T.
1090 003443 316 340 007 FTR53 JSB= TRC01    GET CONST 1.
1091 003446 316 137 001 JSB= DIY20    COMPUTE ANS = 1/T.
1092 003451 236      FTR54 RTH      RETURN WITH ANSWER.
1093 003452 150 223      FTR62 CLM R50      CLEAR PQ AREA.
1094 003454 140 205      LLM R40      LEFT DIG TO E FOR PQO ROUTINE.
1095 003456 170 030 341 FTR64 PCMD R70+R30  GET NEXT TRIG CONSTANT.
1096 003461 316 154 010 JSB= PQ010    COMPUTE PQ DIGIT.
1097 003464 120 212      DCB R20      DECREMENT PQ COUNTER.
1098 003466 266 366      JNZ FTR64      JMP(LOOP) IF NOT ZERO.
1099 003470 147 203      ERM R47      RECOVER FROM LAST SHIFT IN PQO.
1100 003472 207      LRM R47      READY FOR PSEUDO-MULTIPLY.
1101 003473 157 207      LRM R57      PROPERLY POSITION PSEUDO-QUO.
1102 003475 160 223      CLM R60      CLEAR X ACCUMULATOR.
1103 003477 167 210      ICB R67      INITIALIZE FIRST
1104 003501 204      LLB R67      X TO 1.
1105 003502 136 251 007 LDM R36=7C,0 J=7, AND CLEAR EXP(X) AREA.
1106 003505 000
1107 003506 260 017      JMP FTR66      BEGIN PSEUDO-MULTIPLY LOOP.
1108 003510 131 212      FTR68 DCB R31      J=J-1.
1109 003512 177 207      LRM R77      Z<10 TO (-J)) LAST TIME THRU.
1110 003514 207      LRM R77      Z<10 TO (-2J)) LAST TIME THRU.
1111 003515 131 220      FTR70 TSB R31      J REDUCED TO 0 YET?
1112 003517 366 367      JNZ FTR63      JMP (LOOP) IF NOT = 0 YET.
1113 003521 140 060 303 ADN R40,R60    Z=Z+X.
1114 003524 160 070 305 SBM R60,R70    X=X-Z<10 TO (-2J)).
1115 003527 140 070 243 FTR66 STM R40,R70    SAVE Z.
1116 003532 136 031 242 STB R36,R31    SAVE J.
1117 003535 150 220      TSB R50      CURRENT PQ DIGIT = 0?
1118 003537 376 003      JRZ FTR72      JMP IF = 0.
1119 003541 212      DCB R50      DECREMENT CURRENT PQ DIGIT.
1120 003542 360 351      JMP FTR70      GO TO SHIFT LOOP.
1121 003544 157 207      FTR72 LRM R57      GET NEXT PQ DIGIT.
1122 003546 150 221      TSM R50      ANY MORE NON-ZERO PQ DIGITS?
1123 003550 367 021      JZR FTR74      JMP IN NO MORE.
1124 003552 136 212      DCB R36      REDUCE J FOR NEXT DECADE.
1125 003554 147 207      LRM R47      SHIFT THE SUM.
1126 003556 360 347      JMP FTR66      CONTINUE PSEUDO-MULTIPLY LOOP.
1127 003560 374 267      FTR60 JLZ FTR54      JIF TAN, ANS=T.
1128 003562 140 223      CLM R40      COS(T)=
1129 003564 147 250 020 LDB R47=10C    +1 OR -1.
1130 003567 136 223      CLM R36      EXP(ANS)=0.
1131 003571 260 127      JMP FTR84      GO GET SGN AND RTH WITH ANS.
1132 003573 150 060 241 FTR74 LDM R50,R60    MOVE X TO R50.
1133 003576 134 223      CLM R34      EXP(Y)=0.
1134 003600 132 223      CLM R32      SIGNS THE SAME FOR DIVIDE.
1135 003602 126 220      TSB R26      WHICH FUNCTION?
1136 003604 377 005      JRN FTR76      JIF SIN OR COT.

```

```

>>>>      MAIN FORWARD TRIG SUBROUTINE      <<<<<<<<
003606 316 141 010      JSB= UT210      SWAP X AND Y(10 TO J).
137 003611 136 215      TCM R36      REPLACE J BY -J.
138 003613 140 221      FTR76 TSM R40      DENOMINATOR=0?
139 003615 266 042      JNZ FTR79      JMP IF NOT =0.
140 003617 127 220      TSB R27      TAN?
141 003621 375 226      JLN FTR54      JIF NOT TAN OR COT.
142 003623 376 006      JRZ FTR78      JIF TAN.
143 003625 316 377 377      JSB= ERROR      REPORT COT FOR T ON X AXIS.
144 003630 003      OCT 3D      CORRESPONDING ERROR NO.
145 003631 260 004      JMP FTR77      GO GIVE ANS =+INF.
146 003633 316 377 377 FTR78 JSB= ERROR      TRIG OVERFLOW.
147 003636 004      OCT 4D      CORRESPONDING ERROR NO.
148 003637 235      FTR77 CLE      SGN +INF=0.
149 003640 140 223      FTR99 CLM R40      READY FOR ALL 9'S.
150 003642 140 217      FTR88 NCM R40      TAN90=9.99...99E499.
151 003644 141 250 004      LD3 R41=4C      SGN(MANT)=0. EXP=499.
152 003647 200      EL3 R41      PACK IN SIGN OF OVF ANS.
153 003650 140 012 345 FTR73 PUMD R40+R12 PUSH ANSWER ONTO STACK.
154 003653 100 006 343      PCMD R0-R6 POP RETURN STACK AND RETURN.
155 003656 140      DRP 40
156 003657 012      ARF 12
157 003660 236      RTH      DONE.
158 003661 136 215      FTR79 TCM R36      MAKES EXP(NUM) INCR IN SHF10.
159 003663 316 142 001      JSB= SHF10      PROTECT FOR 10/0N... DIV BUG.
160 003666 316 137 001      JSB= DIV20      GIVES TAN OR COT.
161 003671 127 220      TSB R27      WHICH FUNCTION?
162 003673 374 025      JLZ FTR84      JMP IF TAN OR COT.
163 003675 136 034 243      STM R36,R34      DUPLICATE EXP(TAN).
164 003700 140 050 243      STM R40,R50      DUPLICATE MANTISSA.
165 003703 316 035 003      JSB= MPY30      SQUARE TAN(T) (OR COT(T)).
166 003706 316 340 007      JSB= TRC01      GET CONSTANT 1.
167 003711 316 106 002      JSB= ADD20      ADD 1.
168 003714 316 332 002      JSB= SQR30      SQR(1+TANTAN) OR SQR(1+COTCOT).
169 003717 316 043 007      JSB= FTR53      ANS = SIN OR COS.
170 003722 122 032 242 FTR84 STB R22,R32      SGN(ANS).
171 003725 236      RTH      RETURN WITH ANSWER.
172 003726 170 261 131 TRC7 LDMD R70=DCON1 GET PI/4.
173 003731 011
174 003732 177 207      LRM R77      LEFT CARRY DIGIT AREA.
175 003734 170 070 303      ADM R70,R76      PI/2.
176 003737 236      RTH      END.
177 003740 133 222      TRC01 CL3 R33      SGN(CONST)=0.
178 003742 150 223      TRC01 CLM R50      ZERO OUT CONSTANT AREA.
179 003744 157 250 020      LD3 R57=10C      MANT(CONST)=1.
180 003747 134 223      CLM R34      EXP(CONST)=0.
181 003751 236      RTH      END.

```

```

PARMA 03/14/81=====
ITEM    LOC    OBJECT CODE  SRC=COHRXX OBJ=GENHRX   9/11/1981 11:40 AM  PG 30
=====
>>>>>    RANDOMIZE, ROUND, & SEPARATE    <<<<<<<
1182 003752 140 261 201 RNDINI LDMD R40=RND3 DEFAULT SEED TO R40.
1182 003755 011
1183 003756 260 043          JMF RSET20    GO STORE SEED AND EXIT.
1184 003760          ****  RANDOMIZE ATTRIBUTE TABLE  ****
1185 003760 241          OCT 241
1186 003761 316 377 377 RNDIZ. JSB =ON?R    GET OPTIONAL PARAMETER
1187 003764 370 010          JEN CFP        JIF USER SUPPLIED SEED
1188 003766 316 377 377 RNTIME JSB =TIME.    GET SYSTEM TIME
1189 003771 140 012 343      PCMD 40,-12    POP TIME IN MILLISECONDS
1190 003774 367 370          JZR RNTIME     TRY AGN IF ZERO
1191 003776 316 102 010 CFP JSB =SEP10
1192 004001 140 221          RNDSET TSM R40    RANDOMIZE 0?
1193 004003 367 016          JZR RSET20     JIF YES
1194 004005 136 211          RSET9 ICM R36    FIX UP ZERO OTHERWISE
1195 004007 365 004          JPS RSET10     QUIT WHEN EXP GOES POSITIVE.
1196 004011 147 207          LRM R47        SHIFT MANTISSA.
1197 004013 366 370          JNE RSET9      LOOP IF MANT#0.
1198 004015 136 041 242 RSET10 STB R36,R41  CONCATENATE 2 DIGITS OF EXP.
1199 004020 140 250 020      LD3 R40=10C    CONCATENATE A 1.
1200 004023 263 002 204 RSET20 STHD R#X(K-1) STORE THE NEW SEED.
1201 004026 236            RTN              RTN CONTROL TO CALLING ROUTINE.
1202 004027
1203 004027          *    ROUND ROUTINE
1204 004027 140 050 243 ROU10 STM R40,R50  COPY MANTISSA.
1205 004032 152 223          CLM R52        CLEAR ALL BUT LAST 4 DIGITS.
1206 004034 140 303          ADM R40,R50    ADD TO ORIGINAL MANTISSA.
1207 004036 372 060          JNC SEP41      CARRIES IF M>=99999999999995.
1208 004040 147 250 020      LD3 R47=10C    MANT=100000000000.
1209 004043 136 211          ICM R36        ADJUST EXPONENT.
1210 004045 360 051          JMP SEP41      GO CLEAR R40 AND R41 AND EXIT.
1211 004047 134 050 241 SEP20 LDM R34,R50  UNPACK B: EXP(B) TO R34.
1212 004052 135 206          LRB R35        SGH(MANT) TO E.
1213 004054 310 005          CMS R35=5C     EXP SIGN NEG?
1214 004056 372 002          JNC SEP30     JMP IF EXP SIGN POS.
1215 004060 312 220          ADB R35=90C    EXP IN R34M IN 10 COMPL FORM.
1216 004062 133 200          SEP30 ELB R33   SGH(B) TO R33.
1217 004064 150 222          CLB R50        CLEAR NON-
1218 004066 151 222          CLB R51        MANTISSA DIGITS.
1219 004070 236          SEP35 RTN          END.
1220 004071 316 116 012 TUOSEP JSB =TUOR01  GET THE DATA
1221 004074 370 372          JEN SEP35      DONE IF INTEGER
1222 004076 231          SEP15 BCD
1223 004077          *    LDM R34,R50      EXP TO R34
1224 004077          *    CLB R50          CLEAR EXP
1225 004077          *    CLB R51
1226 004077          *    LRB R35          SIGN TO E
1227 004077          *    CMS R35,=5C     SIGN OF EXP?
1228 004077          *    JNC SAP30       JIF EXP SIGN POS
1229 004077          *    ADB R35,=90C    EXP IN R34M IN 10 COMP FORM
1230 004077          +SAP30 ELB R33        SIGN(B) TO R33
1231 004077 316 047 010      JSB =SEP20
1232 004102          * NOW DOING OLD SEP10
1233 004102 136 040 241 SEP10 LDM R36,R40  UNPACK A:EXP TO R36

```

>>>>	RANDOMIZE, ROUND, & SEPARATE		LLLLLLLL
004105	137 206	LR9 R37	SGN(MANT) TO E
235 004107	310 005	CM9 R37,=5C	EXP SIGN NEG?
236 004111	372 002	JNC SAP40	JIF EXP SIGN POS .
237 004113	312 220	AD9 R37,=90C	EXP IN R36M IN 10 COMP FORM
238 004115	132 200	SAP40 EL3 R32	SGN(A) TO R32
239 004117	235	CLE	
240 004120	140 222	SEP41 CL3 R40	CLEAR EXP
241 004122	141 222	CL3 R41	
242 004124	236	RTH	DONE

```

ARMA 03/14/81=====
ITEM    LOC    OBJECT CODE  SRC=LCRRXX OBJ=GEHRX  9/11/1981 11:40 AM PG 32
=====
>>>>  UTILITIES, PQ0, AND MIN/MAX  <<<<<<
1244 004125
1245 004125      *      UTILITY ROUTINES
1246 004125 140 060 243 UT110  STN R40,R60  MANT TO R60.
1247 004130 223      CLM R40      INITIALIZE.
1248 004131 136 034 303 UT110F ADN R36,R34  ADD EXPONENTS.
1249 004134 132 033 226      XRS R32,R33  CALCULATE SIGN OF ANSWER.
1250 004137 060      ARP 60      SET ARP.
1251 004140 236      RTH        END.
1252 004141 150 060 243 UT210  STN R50,R60  R50 TO TEMP.
1253 004144 040 241      LDM R50,R40  R40 TO R50.
1254 004146 160 243      STN R60,R40  TEMP TO R40.
1255 004150 236      RTH        END.
1256 004151
1257 004151      *      EXP/TRIG PSEUDO-QUOTIENT ROUTINE
1258 004151 233      PQ030  DCE      BORROW FROM E.
1259 004152 150 210      PQ020  ICB R50  TALLY 1 FOR SUCCESSFUL SUBTRACT.
1260 004154 140 070 305 PQ010  SBN R40,R70  SUBTRACT LN(1+(10 TO (-J))).
1261 004157 373 371      JCY PQ020  LOOP TILL OVERFLOW.
1262 004161 370 366      JEN PQ030  CONTINUE LOOP IF E#0.
1263 004163 303      ADN R40,R70  RECOVER FROM OVERDRAFT.
1264 004164 150 205      LLM R50  READY PQ AREA FOR NEXT DIGIT.
1265 004166 140 205      LLM R40  R40 & E READY FOR NEXT DECADE.
1266 004170 236      RTH        END.
1267 004171      *      MIN/MAX ROUTINE
1268 004171      ****  MIN ATTRIBUTES TABLE  ****
1269 004171 040 055      OCT 40,55
1270 004173      ****
1271 004173 316 066 012 MIN10  JSB= TUOR  GET REAL A AND B.
1272 004176 316 141 010      JSB= UT210  PUT B IN R50, A IN R40.
1273 004201 140 012 345      PUMD R40+R12  REAL COPY OF A ON STACK.
1274 004204 150 345      PUMD R50+R12  REAL COPY OF B ON STACK.
1275 004206 360 012      JMP MAX11  CONTINUE.
1276 004210      ****  MAX ATTRIBUTES TABLE  ****
1277 004210 040 055      OCT 40,55
1278 004212      ****
1279 004212 316 066 012 MAX10  JSB= TUOR  GET REAL A AND B.
1280 004215 150 012 345      PUMD R50+R12  REAL COPY OF A TO STACK.
1281 004220 140 345      PUMD R40+R12  REAL COPY OF B TO STACK.
1282 004222 316 076 010 MAX11  JSB= SEP15  SEPARATE A AND B.
1283 004225 132 216      NCB R32  NEGATE B FOR A-B (B-A IF MIN).
1284 004227 316 106 002      JSB= ADD20  A+(-B) OR B+(-A).
1285 004232 150 012 343      PCMD R50-R12  POP B; A STILL ON STACK.
1286 004235 132 220      TSB R32  A-B POS? (B-A POS?).
1287 004237 376 004      JRZ MAX30  IF POS, ANS=A (BOTH CASES).
1288 004241 140 343      PCMD R40-R12  RID STACK OF A.
1289 004243 150 345      PUMD R50+R12  PUT ANS=B ON STACK.
1290 004245 236      MAX30  RTH        END.
1291 004246
1292 004246

```

```

>>>>      MAIN EXP(X) SUBROUTINE      <<<<<<<<
100 004246 316 102 010 EXP18  JSB= SEP10  SEPARATE X.
295 004251 316 276 010 EXP20  JSB= EXP17  CALCULATE EXP(X) - 1.
296 004254 316 342 007        JSB= TRCS1  GET 1 BUT LEAVE R33 ALONE.
297 004257 316 106 002        JSB= ADD20  (EXP(X)-1)+1.
298 004262 136 014 303        ADX R36,R14  K=EXP(ANS).
299 004265 132 220          TSB R32      X<0?
300 004267 376 004          JRZ EXP22    JIF X POS.
301 004271 222            CLB R32      SGN=0.
302 004272 316 043 007        JSB= FTR53  EXP(X) = 1/EXP(ABS(X)).
303 004275 236            RTH          END.
304 004276 132 033 242 EXP17  STB R32,R33  SAVE SGN(X) (NEW KILLS R32).
305 004301 124 251 241        LDH R24=TBL2 PT TO LN10; BOTTOM OF LN TABLE.
305 004304 011
306 004305 150 223          CLM R50      CLEAR PQ AREA.
307 004307 126 223          CLM R26      J=0.
308 004311 114 223          CLM R14      K ACCUMULATOR FOR X-LN(10).
309 004313 235            CLE          FOR PRESCALE & ENTERING PQO.
310 004314 137 310 120        CMS R37=50C  EXP(X)<0?
311 004317 373 055          JCY EXP30    JIF EXP(X)<0.
312 004321 170 024 245        LDND R70,R24 GET LN(10).
313 004324 070          ARP 70        SET ARP.
314 004325 130 202          ERB R30      SAVE E.
315 004327 114 205          LLM R14      FOR NEXT DECADE IN EXP(ANS).
316 004331 130 204          LLB R30      RESTORE E.
317 004333 260 003          JMF EXP24    TO CAST OUT LN(10)'S.
318 004335 233          DCE          BORROW FROM E.
319 004336 114 211          ICM R14      K=K+1.
320 004340 140 305          SSH R40,R#   X-LN(10).
321 004342 373 372          JCY EXP25    LOOP TILL OVERDRAFT.
322 004344 370 367          JEH EXP26    LOOP TILL E=0.
323 004346 303          ADX R40,R#      RECOVER FROM OVERDRAFT.
324 004347 205          LLM R40      LEFT DIGIT TO E; 0 IN FROM RT.
325 004350 114 311 231        CMH R14=99C,9C EXP(ANS)>999?
325 004353 011
326 004354 372 012          JNC EXP28    JIF EXP(ANS) TOO BIG.
327 004356 251 230 011        LDH R14=99C,9C LARGE EXP FOR OVF.
328 004361 136 223          CLM R36      SET EXP(R40)=0.
329 004363 140 223          CLM R40      READY FOR #0 MANTISSA.
330 004365 142 217          NCM R42      #0 MANTISSA.
331 004367 236          RTH          RTN TO OUTER LOOP.
332 004370 136 213          DCM R36      DECR EXP(X) EACH LOWER DECADE.
333 004372 372 023          JNC EXP31    JIF EXP(X) DEPLETED.
334 004374 260 327          JMF EXP66    LOOP & SUB LN10'S NEXT LWR DEC.
335 004376 126 036 241 EXP30  LDH R26,R36  COPY EXP(X).
336 004401 217          NCM R26      J=ABS(EXP(X))-1.
337 004402 311 010 000        CMH R26=8C,0  J<=7?
338 004405 373 040          JCY EXP34    JIF J>7.
339 004407 230          BIN          ELSE SET MODE FOR SHIFTS.
340 004410 205          LLM R26      GIVES 2J.
341 004411 205          LLM R26      GIVES 4J.
342 004412 205          LLM R26      GIVES 8J.
343 004413 124 026 305        SSH R24,R26  POINT TO JTH LN CONSTANT.
344 004416 231          BCD          RESET MODE.

```

```

>>>      MAIN EXP(X) SUBROUTINE                                     <<<<<<<
345 004417 170 024 343 EXP31 PCMD R70-R24 GET NEXT LN CONSTANT.
346 004422 316 154 010      JSB= PQ010 GET NEXT PQ DIGIT.
347 004425 230              BIN      SET MODE FOR COMPARE.
348 004426 124 311 141      CMM R24=TBL1 LAST TABLE ENTRY YET?
348 004431 011
349 004432 231              BCD      RESET MODE.
350 004433 266 362          JNE EXP31 LOOP TILL ALL PQ DIGITS FOUND.
351 004435 147 203          ERM R47 RECOVER FROM LAST SHIFT IN PQ0.
352 004437 126 251 007      LDM R26=7C,0 J=7.
352 004442 000
353 004443 136 251 222      LDM R36=92C,99C EXP(R) = -8.
353 004446 231
354 004447 150 070 243 EXP34 STM R50,R70 SAVE PSEUDO QUOTIENT.
355 004452 316 333 000      JSB= NEWR GO CALCULATE R/(1-.5R).
356 004455 133 032 242      STB R33,R32 RESTORE S(X).
357 004460 260 002          JMP EXP39 JMP TO PSEUDO-MULTIPLY LOOP.
358 004462 126 213          EXP38 DCM R26 NO. OF RT SHIFTS DESIGNATOR.
359 004464 174 207          EXP39 LRM R74 NEXT PQ DIGIT TO RT END OF R70.
      004466 170 221          TSM R70 PSEUDO-QUOTIENT=0?
361 004470 367 275          JZR EXP29 EXIT WHEN PQ EXHAUSTED.
362 004472 147 207          LRM R47 LEAD 0.
363 004474 136 211          ICM R36 ADJUST EXP(EXP(X)-1).
364 004476 260 014          JMP EXP42 GO SEE IF CURRENT PQ DIGIT=0.
365 004500 157 207          EXP40 LRM R57 SHIFT P.
366 004502 124 213          EXP41 DCM R24 DECR SHIFT COUNTER.
367 004504 373 372          JCY EXP40 LOOP TILL CTR<0.
368 004506 140 050 303      ADM R40,R50 NEW P=P+SHIFTED P.
369 004511 147 313 020      ADM R47=10C NEW P=NEW P+(10 TO J).
370 004514 170 220          EXP42 TSB R70 CURRENT PQ DIGIT = 0?
371 004516 376 342          JRZ EXP38 JIF DONE WITH CURRENT PQ DIGIT.
372 004520 212              DCB R70 ELSE DECR CURRENT PQ DIGIT.
373 004521 140 050 243      STM R40,R50 COPY P=PARTIAL PRODUCT.
374 004524 124 026 241      LDM R24,R26 COPY SHIFT COUNTER.
375 004527 260 351          JMP EXP41 GO SHIFT P.

```

```

>>>>          TABLES AND CONSTANTS          <<<<<<<<
1377 004531 200 104 227 DCON1 DEC 7253981633974480
1377 004534 063 026 230
1377 004537 123 170
1378 004541 064 000 000 TBL1 DEC 9999995000000034
1378 004544 000 225 231
1378 004547 231 231
1379 004551 064 063 000 DEC 9999995000000334
1379 004554 000 120 231
1379 004557 231 231
1380 004561 061 063 063 DEC 9999950000333331
1380 004564 000 000 225
1380 004567 231 231
1381 004571 064 010 063 DEC 9999500033330834
1381 004574 063 000 120
1381 004577 231 231
1382 004601 062 123 203 RND3 DEC 9995003330835332
1382 004604 060 063 000
1382 004607 225 231
1383 004611 203 200 026 DEC 9950330853168083
1383 004614 123 010 063
1383 004617 120 231
1384 004621 207 044 103 DEC 9531017980432487
1384 004624 200 171 001
1384 004627 061 225
1385 004631 124 224 131 DEC 6931471805599454
1385 004634 005 030 107
1385 004637 061 151
1386 004641 106 100 231 TBL2 DEC 2302585092994046
1386 004644 222 120 130
1386 004647 002 043
1387 004651 003 142 021 TBL3 DEC 9966865249116203
1387 004654 111 122 206
1387 004657 146 231
1388 004661 070 122 146 DEC 9999666686665238
1388 004664 206 146 146
1388 004667 231 231
1389 004671 147 206 146 DEC 9999996666668667
1389 004674 146 146 231
1389 004677 231 231
1390 004701 147 146 146 DEC 9999999966666667
1390 004704 146 231 231
1390 004707 231 231
1391 004711 147 146 146 DEC 9999999999666667
1391 004714 231 231 231
1391 004717 231 231
1392 004721 147 146 231 DEC 9999999999996667
1392 004724 231 231 231
1392 004727 231 231
1393 004731 147 231 231 DEC 9999999999999967
1393 004734 231 231 231
1393 004737 231 231
1394 004741          TBL3B BSS 0
  
```

```

>>>> REAL/INTEGER DATA FETCH
1397 004741 ENT INTORL
1397 004741 ENT ONEX
1398 004741 ENT ONEX+
1399 004741 ENT ONEI
1400 004741 ENT ONER
1401 004741 ENT ONER+
1402 004741 ENT ONEROI
1403 004741 ENT TWOSEP
1404 004741 * ENT TWOI
1405 004741 ENT TWOB
1406 004741 ENT TWOR
1407 004741 ENT TWOROI
1408 004741 * DATA LOCATION AT ROUTINE EXIT:
1409 004741 * ONER: REAL IN R40
1410 004741 * ONEI: BCD INT IN R40
1411 004741 * ONEROI: REAL IN R40 OR INTEGER IN R40
1412 004741 * ONEB: BINARY INT IN 40
1413 004741 * TWO---: B=1ST OFF STACK IN R40, A=2ND IN R50
1414 004741 * SOME EXTRA STUFF TO GET ONE INTEGER
1416 004741 * ONEB - GET ONE BIN INT, 32767 IF TOO BIG
1417 004741 * RETURNS E<>0 FOR OVF,UNDERFLOW
1418 004741 231 ONEX BCD
1419 004742 160 012 343 POND R60,-R12 GET VAL FROM STAK
1420 004745 164 310 377 ONEX+ CMB R64,-377 INT?
1421 004750 372 025 JNC REALB JIF REAL
1422 004752 165 206 LRB R65 LSB TO E
1423 004754 221 TSM R65
1424 004755 266 013 JNZ ONEB1 JIF NOT SMALL INT
1425 004757 132 222 CLB R32 SIGN = POSITIVE
1426 004761 176 223 CLM R76
1427 004763 200 ELB R76 ANS TO 46
1428 004764 235 CLE
1429 004765 176 046 243 ONEBU STM R76,R46
1430 004770 230 BIN
1431 004771 236 RTN
1432 004772 165 200 ONEB1 ELB R65
1433 004774 316 173 012 JSB =INTORL CONVERT TO REAL
1434 004777 316 377 377 REALB JSB =CONINT CONV 60 TO BIN INT IN 76
1435 005002 260 361 JMP ONEBU
1436 005004 * ONEI - GET ONE BCD INT, 99999 IF TOO BIG
1437 005004 316 103 012 ONEI JSB= ONEROI GET ONE OF EITHER
1438 005007 370 053 JEN ONEIOK JIF INTEGER FOUND
1439 005011 160 040 241 LDM R60,R40 MOVE ANSWER TO R60.
1440 005014 316 377 377 JSB= RTOIN CONVERT TO BCD INTEGER.
1441 005017 144 250 377 LDB R44,-377
1442 005022 145 065 241 LDM R45,R65
1443 005025 236 RTN DONE.
1444 005026 * END OF EXTRA CODE
1445 005026
1446 005026 * TWOB - GET TWO BIN INT, EXIT IN BIN
1447 005026 316 377 377 TWOB JSB =ONEB
1448 005031 146 026 243 STM R46,R26

```

```

>>>> REAL/INTEGER DATA FETCH <<<<<<<<
005034 316 377 377 JSB =ONE9 GET THE 2ND ONE
450 005037 146 056 243 STM 46,56
451 005042 026 241 LD1 46,26
452 005044 236 RTN
453 005045
454 005045
455 005045 * TWOI - GET TWO BCD INTEGERS, EXIT IN BCD
456 005045 *TWOI JSB= ONEI GET FIRST INTEGER.
457 005045 * PUMD R44,+R5
458 005045 * JSB= ONEI GET 2ND INTEGER.
459 005045 * STM 44,54 MOVE ANSWER TO 54
460 005045 * PUMD R44,-R5 RESTORE FIRST OFF
461 005045 *ROIRTH RTN DONE.
462 005045
463 005045 160 012 343 ONER ONER - GET ONE REAL, CONVERT IF NECESSARY
464 005050 164 310 377 ONER+ PCMD R60,-R12 GET TAG.
465 005053 372 003 CM3 R64=377 IS NO ON STACK REAL?
466 005055 316 173 012 RIDF20 JNC FLOR1N JIF YES.
467 005060 231 FLOR1N BCD CONVERT IT TO FLOATING.
468 005061 160 040 243 STN R60,R40 REALS MUST BE BCD
469 005064 235 ONEIOK CLE
470 005065 236 RTN NEEDED FOR TWOROI AND ONEI
471 005066 END.
472 005066
473 005066 316 045 012 * TWOR - GET TWO REALS, CONVERT IF NECESSARY
474 005071 140 050 243 TUOR JSB =ONER GET ONE OF THEM
475 005074 316 045 012 STM R40,R50 SAVE FIRST RESULT
476 005077 316 141 010 JSB =ONER NOW GET SECOND ONE
477 005102 236 JSB= UT210 A TO R50, B TO R40
478 005103 RTN DONE
479 005103
480 005103 235 * ONEROI - FETCH ONE REAL OR INTEGER
481 005104 140 012 343 ONEROI CLE RESET FLAG
482 005107 144 310 377 PCMD R40,-R12 GET VALUE
483 005112 372 001 CM3 R44=377 IS NO ON STACK REAL?
484 005114 234 JNC ROIRTH JIF YES (DONE)
485 005115 236 ICE FLAG INTEGER
ROIRTH RTN DONE
  
```

```

ARMA 03/14/81=====
ITEM    LOC  OBJECT CODE  SRC=OCHRX  OBJ=CEWHRX  9/11/1981 11:40 AM  PG 38
=====
>>>>  TWOROI - GET 2 REAL OR INTEGER                                     <<<<<<
487 005116      *THERE ARE FOUR CASES OF TWOROI:
488 005116      *
489 005116      *           B           A
490 005116      * CASE 1:    R44    R54
491 005116      * CASE 2:    377    377    BOTH INTEGERS
492 005116      * CASE 3:    BCD    BCD    B REAL, A INTEGER
493 005116      * CASE 4:    BCD    377    A REAL, B INTEGER
494 005116      *           BCD    BCD    BOTH REAL
495 005116 235   * MODE ON RETURN IS BCD
496 005117 140 012 343  TWOROI CLE          FOR INCREMENT
497 005122 150 343      PCMD R40,-R12 GET B
498 005124 154 310 377  PCMD R50,-R12 GET A WHILE ARP = 12
499 005127 372 022      CMB R54,=377 IS A REAL?
500 005131 144 310 377  JNC CASE34    JIF YES
501 005134 372 002      CMB R44,=377 ARE BOTH INTEGERS
502 005136 234        JNC CASE2    JIF NO, CASE 2
503 005137 236        ICE          FLAG INTEGERS
504 005140 160 050 241 CASE2 RTH          DONE, CASE 1 IS EASY
505 005143 316 173 012      LDM R60,R50
506 005146 160 050 243      JSB =INTORL  CONVERT A
507 005151 235          STM R60,R50
508 005152 236          CLE          FLAG REALS
509 005153 144 310 377 CASE34 RTH          DONE
510 005156 372 012      CMB R44,=377 IS B INTEGER?
511 005160          JNC TWORTH    JIF CASE 4, DONE
512 005160 160 040 241      *CASE 3 - B IS INTEGER, SO CONVERT IT
513 005163 316 173 012      LDM R60,R40  GET B TO 60 FOR INTORL
514 005166 160 040 243      JSB =INTORL  CONVERT B
515 005171 235          STM R60,R40
516 005172 236          CLE          FLAG REALS
517 005173 231          TWORTH RTH          DONE
518 005174 136 223          INTORL BCD
519 005176 062 243          CLM R36      EXP/SIGN
520 005200 164 222          STM R36,R62  CLEAR CRAP
521 005202 167 220          CLB R64      MORE CRAP
522 005204 365 005          TSB R67
523 005206 165 215          JPS INTOR1    JIF MANT POS
524 005210 137 250 011      TCM R65      COMPL
525 005213 165 221          LD3 R37=9C    SET SIGN NEG
526 005215 367 015          INTOR1 TSM R65
527 005217 136 250 005      JZR INTORX    JIF ZERO
528 005222 167 220          LD3 R36=5C    EXP
529 005224 375 006          INTOR2 TSB R67
530 005226 165 205          JLN INTORX    JIF NORMALIZED.
531 005230 136 212          LLM R65      SHIFT
532 005232 260 366          DCB R36      DEC EXP
533 005234 136 060 243 INTORX JMP INTOR2  LOOP
534 005237 236          STM R36,R60  SIGN/EXP
535 005240          RTH
536 005240          * UNIL FOR FIN
                    FIN

```

```

=====
ARMA 03/14/81=====
SYMBOL  VALUE      TYPE      COUNT      Symbol Table      9/11/1981 11:40 AM  PG 39
=====
+B#0    001127      LCL       1
+      002573      ENT       1
+      002601      LCL       1
+DD10    001076      ENT       1
+DD15    001101      LCL       2
+DD20    001106      ENT       8
+DD40    001176      LCL       1
+DD50    001214      LCL       1
+DD52    001211      LCL       1
+DD8     001072      LCL       2
+DD9     001073      ENT       1
+DD95    001205      LCL       3
+DONOW   001155      LCL       2
+DDROI   001013      ENT       1
+DLOOP   001664      LCL       1
+DRI10   001035      LCL       0      <--- NOT REFERENCED??
+DROI+   001016      ENT       1
+DSU     001033      ENT       2
+DSUEX   001021      LCL       1
+RTS35   177777      EXT       1
+      177777      EXT       0      <--- NOT REFERENCED??
+      005140      LCL       1
+ASE34   005153      LCL       1
+CEIL10  002460      ENT       1
+CFP     003776      LCL       1
+CHS10   000750      ENT       1
+CHS10X  000764      LCL       1
+CHS11   000760      LCL       2
+      )OI 000740      ENT       3
+CONINT  177777      EXT       1
+COS10   002421      ENT       1
+COT10   002401      ENT       1
+CSEC10  002346      ENT       1
+CSEC11  002375      LCL       1
+CSEC30  002372      LCL       2
+CON1    004531      ENT       5
+EG10    003004      ENT       1
+DIV10   000507      ENT       1
+DIV11   000515      LCL       1
+      000512      ENT       2
+      000421      LCL       2
+DIV2    000504      ENT       1
+DIV20   000537      ENT       8
+DIV30   000447      LCL       1
+DIV39   000456      LCL       1
+DIV40   000461      LCL       1
+DIV50   000462      LCL       1
+DIV77   000434      ENT       2
+DIV79   000453      LCL       1
+DIV88   000521      ENT       3
+DIV89   000531      LCL       0      <--- NOT REFERENCED??
+DIV99   000501      LCL       0      <--- NOT REFERENCED??
+ONTAD   001770      LCL       1
+OROM    002272      ENT       3

```

=====				
EXP	100160	G 5AD	1	
EXP0	002779	ENT	1	
EXP0R	177777	EXT	7	
EXP0R+	177777	EXT	5	
EXP11	001250	LCL	1	
EXP17	004276	ENT	2	
EXP18	004246	LCL	1	
EXP20	004251	ENT	2	
EXP22	004275	LCL	1	
EXP24	004340	LCL	1	
EXP25	004336	LCL	1	
EXP26	004335	LCL	1	
EXP28	004370	LCL	1	
EXP29	004367	LCL	1	
EXP30	004376	LCL	1	
EXP31	004417	LCL	2	
EXP34	004447	LCL	1	
EXP38	004462	LCL	1	
EXP39	004464	LCL	1	
EXP40	004500	LCL	1	
EXP41	004502	LCL	1	
EXP42	004514	LCL	1	
EXP5	001242	ENT	1	
EXP66	004325	LCL	1	
EXP=	001141	LCL	1	
FLORTN	005060	LCL	1	
FP5	002733	ENT	1	
FTR11	002433	LCL	3	
FTR12	003075	LCL	3	
FTR14	003122	LCL	2	
FTR15	002436	LCL	1	
FTR16	003204	LCL	1	
FTR17	003223	LCL	1	
FTR18	003141	LCL	1	
FTR19	003120	LCL	1	
FTR20	003163	LCL	3	
FTR24	003212	LCL	1	
FTR25	003224	LCL	2	
FTR26	003253	LCL	2	
FTR28	003301	LCL	1	
FTR29	003270	LCL	1	
FTR30	003312	LCL	1	
FTR32	003315	LCL	2	
FTR34	003340	LCL	2	
FTR36	003327	LCL	1	
FTR38	003274	LCL	1	
FTR40	003343	LCL	0	<--- NOT REFERENCED??
FTR42	003375	LCL	1	
FTR46	003415	LCL	1	
FTR53	003443	ENT	4	
FTR54	003451	LCL	3	
FTR60	003560	LCL	1	
FTR62	003452	LCL	2	
FTR64	003456	LCL	1	

TR66	003527	LCL	2	
	003510	LCL	1	
	003515	LCL	1	
TR72	003544	LCL	1	
TR73	003650	ENT	1	
TR74	003573	LCL	1	
TR76	003613	LCL	1	
TR77	003637	ENT	3	
TR78	003633	LCL	2	
TR79	003661	LCL	1	
TR84	003722	LCL	2	
TR88	003642	ENT	2	
TR99	003640	ENT	1	
NF10	002367	ENT	2	
NFR1	002664	ENT	1	
NFR10	002635	LCL	4	
NFR2	002666	LCL	1	
NFR3	002606	ENT	1	
NFR4	002611	ENT	2	
NFR5	002622	LCL	1	
	002712	LCL	1	
NT5	002640	ENT	2	
HTDIV	002647	ENT	1	
HTFRA	002603	ENT	2	
HTM2	001776	LCL	1	
HTMUL	001741	ENT	1	
HTOR1	005213	LCL	1	
HTOR2	005222	LCL	1	
RL	005173	ENT	5	
HTORX	005234	LCL	2	
P3	003050	ENT	2	
P4	003040	LCL	1	
P5	003036	ENT	1	
COM1	002317	LCL	3	
COM2	002323	LCL	0	<--- NOT REFERENCED??
N20	000000	ENT	3	
N22	000006	ENT	1	
N24	000016	LCL	1	
N30	000076	ENT	3	
M	000104	ENT	1	
N31	000135	LCL	1	
N34	000172	LCL	1	
N36	000026	LCL	1	
N38	000052	ENT	1	
N40	000057	LCL	1	
N5	000414	ENT	1	
N50	000176	LCL	1	
N52	000207	LCL	1	
N54	000217	LCL	1	
N56	000221	LCL	1	
N58	000227	LCL	1	
N60	000235	LCL	1	
N62	000173	LCL	1	
N80	000257	LCL	2	

SYMBOL	VALUE	TYPE	COUNT	Symbol Table	9/11/1981 11:40 AM PG 42
LN94	000376	LCL	1		
LN95	000390	LCL	1		
LN96	000306	LCL	2		
LN98	000312	LCL	1		
LN98	000402	LCL	1		
LN99	000410	LCL	1		
LOGCOM	002306	LCL	1		
LOGT5	000562	ENT	1		
TAX10	004212	ENT	1		
TAX11	004221	LCL	1		
TAX30	004245	LCL	1		
MIN10	004173	ENT	1		
LOOP	001761	LCL	1		
MOD10	000607	ENT	1		
MOD20	000612	LCL	1		
MOD30	000624	LCL	1		
MOD35	000705	LCL	1		
MOD40	000630	LCL	2		
MOD50	000634	LCL	1		
MOD60	000640	LCL	2		
MOD70	000701	LCL	3		
MPY10	001425	ENT	1		
MPY11	001430	LCL	1		
MPY28	001432	ENT	2		
MPY30	001435	ENT	7		
MPY31	001453	LCL	1		
MPY32	001477	LCL	1		
MPY36	001550	LCL	3		
MPY50	001523	LCL	1		
MPY52	001531	LCL	1		
MPY60	001532	LCL	1		
MPY70	001554	LCL	1		
MPY77	001514	LCL	1		
MPYEX	001573	LCL	1		
MPYLOP	001656	LCL	1		
MPYR	001604	ENT	2		
MPYROI	001565	ENT	1		
NEWR	000333	LCL	2		
NEWR2	000375	LCL	1		
NEWR3	002744	ENT	2		
NEWR4	002747	LCL	3		
NEWR5	002750	LCL	3		
NEWR6	002753	ENT	2		
NOFLIP	001640	LCL	1		
ON?R	177777	EXT	1		
ONEB	177777	EXT	2		
ONEB1	004772	LCL	1		
ONEBJ	004765	LCL	1		
ONEI	005004	ENT	1		
ONEIOK	005064	LCL	1		
ONER	005045	ENT	11		
ONER+	005050	ENT	1		
ONEROI	005103	ENT	3		
ONEX	004741	ENT	1		

=====					
INEX+	004745	ENT	1		
INEX	001725	LCL	1		
INEX	001735	LCL	1		
INPD=0	001126	LCL	1		
IVF10	002476	ENT	2		
IVF19	002472	LCL	1		
PI10	002442	ENT	1		
PI11	002454	LCL	0	<--- NOT REFERENCED??	
POS	001721	LCL	2		
POSGN	001062	LCL	1		
Q010	004154	LCL	2		
Q020	004152	LCL	1		
Q030	004151	LCL	1		
RAD10	002540	ENT	1		
RAD11	002567	LCL	1		
REALB	004777	LCL	1		
REM10	000601	ENT	1		
RIDF20	005055	LCL	0	<--- NOT REFERENCED??	
RND10	002007	ENT	1		
RND30	002045	LCL	1		
RND31	002047	LCL	1		
RND3	002040	LCL	1		
RNDINI	003752	ENT	1		
RNDIZ.	003761	ENT	1		
RNDS	004601	LCL	1		
RNDSET	004001	ENT	1		
ENTIME	003756	LCL	1		
ROIRTN	005115	LCL	1		
R)	002741	ENT	5		
RQU10	004027	ENT	2		
RSET10	004015	LCL	1		
RSET20	004023	LCL	2		
RSET9	004005	LCL	1		
RTDIN	177777	EXT	2		
SAP40	004115	LCL	1		
SBR01+	000775	ENT	1		
SEC10	002326	ENT	1		
SEP10	004102	ENT	8		
SEP15	004076	ENT	9		
SEP20	004047	ENT	4		
SEP3	004062	LCL	1		
SEP35	004079	LCL	1		
SEP41	004120	LCL	2		
SGN2	001626	LCL	1		
SGN3	001630	LCL	1		
SGN5	002250	ENT	1		
SGN6	002265	LCL	3		
SHF10	000542	ENT	13		
SHF20	000552	LCL	1		
SHF30	000556	LCL	1		
SHRONF	002736	ENT	3		
SIN10	002411	ENT	1		
SQR11	001326	LCL	2		
SQR15	001320	LCL	1		
=====					

```

=====
ARMA 03/14/81=====
SYMBOL  VALUE      TYPE      COUNT      Symbol Table      9/11/1981 11:40 AM  PG 44
=====
BQR30    001332      ENT       3
BQR35    001353      LCL       1
BQR36    001362      LCL       1
BQR37    001402      LCL       1
BQR40    001420      LCL       1
BQR45    001255      LCL       1
BQR5     001305      ENT       1
BQR65    001252      LCL       1
BQR99    001301      LCL       0      <--- NOT REFERENCED??
SUB10    001002      ENT       1
SUB11    001005      LCL       0      <--- NOT REFERENCED??
SUBIL    001025      ENT       2
SUBROI    000772      ENT       1
SWAP+-    001146      LCL       1
TAN10    002431      ENT       1
TBL1     004541      LCL       2
TBL2     004641      LCL       3
TBL3     004651      LCL       1
TBL3B    004741      ENT       1
TEMP2    101731      G DAD     2
TIME     177777      EXT       1
TRC1     003740      ENT       4
TRC7     003726      ENT       2
TRCS1    003742      ENT       2
TRUNC    002722      LCL       3
TSTEZ    001665      LCL       1
TWOB     005026      ENT       1
TWOR     005066      ENT       8
TWOROI    005116      ENT       2
TWORTHN  005172      LCL       1
TWOSEP    004071      ENT       4
UT110    004125      LCL       1
UT110F   004131      LCL       1
UT210    004141      ENT      10
XCK-1)   102002      G DAD     3
YTX11    002211      LCL       0      <--- NOT REFERENCED??
YTX12    002213      LCL       1
YTX14    002161      LCL       2
YTX16    002242      LCL       2
YTX18    002145      LCL       1
YTX20    002140      LCL       1
YTX22    002163      LCL       1
YTX26    002230      LCL       1
YTX5     002105      ENT       1

```

TABLE OF CONTENTS	1
TABLE OF CONTENTS	2
TRY POINTS, EXTERNALS	3
MAIN LNK(X) SUBROUTINE	5
DIVIDE AND SHIFT ROUTINES	8
LOG BASE 10, MOD/REM	9
REAL/INTEGER CHS, SUB, ADD	11
SQUARE ROOT ROUTINE	14
MULTIPLY, INTEGER OR REAL	15
INTEGER MULTIPLY INTM AND INTMUL	17
RANDOM NUMBER GENERATOR ROUTINE	18
CHANGE SIGN (UNARY MINUS) ROUTINE	19
Y*X ROUTINE	19
SGN, FTR ENTRY, DRCOM, & LOGCOM	20
PI, CEIL, OVF/UF, DEG-RAD	22
ABS, FLOOR(=INT), & A/B	23
FP, NFR, EPS, RAD-DEG, & IP	24
MAIN FORWARD TRIG SUBROUTINE	26
RANDOMIZE, ROUND, & SEPARATE	30
UTILITIES, PQO, AND MIN/MAX	32
EXP(X) SUBROUTINE	33
TABLES AND CONSTANTS	35
REAL/INTEGER DATA FETCH	36
TWORDI - GET 2 REAL OR INTEGER	38

CHRXX HAD 0 ERRORS 0 WARNINGS 314 LABELS LAST ERROR AT 0

)