```
ARMA 03/14/81 ================================================================================
ITEM    LOC    OBJECT CODE  SRC=CCALLO OBJ=GEXALL   9/11/1981 11:30 AM  PG  1
================================================================================
>>>>      MEMORY ALLOCATION                                              <<<<<<<<
  2 000000                        ABS
  3 000000                        LOC 40000
  4 040000              *
  5 040000              *
  6 040000              *      I N D E X
  7 040000              *
  8 040000              *
  9 040000              *        INDEX                                  PAGE 1
 10 040000              *        ENTRY POINTS                           PAGE 2
 11 040000              *        ALLOC                                  PAGE 3
 12 040000              *        VALOC                                  PAGE 6
 13 040000              *        VSRCH                                  PAGE 8
 14 040000              *        VSTOR                                  PAGE 9
 15 040000              *        COMMON ALLOC(STORIY)                   PAGE 11
 16 040000              *        LINEAL                                 PAGE 13
 17 040000              *        DEFFN                                  PAGE 13
 18 040000              *        FUNCAL                                 PAGE 15
 19 040000              *        COMM                                   PAGE 15
 20 040000              *        DIM/INT/SHORT/OPTION                   PAGE 16
 21 040000              *        ASCNAM                                 PAGE 20
 22 040000              *        ALINI                                  PAGE 22
 23 040000              *        SKPPAR                                 PAGE 22
 24 040000              *        TSTALO                                 PAGE 24
 25 040000              *        GETNXT                                 PAGE 24
 26 040000              *        NXTLIN                                 PAGE 25
 27 040000              *        SKPCON                                 PAGE 25
 28 040000              *        FRMNAM                                 PAGE 26
 29 040000              *        CALCSZ                                 PAGE 27
 30 040000              *        INIVAR                                 PAGE 27
 31 040000              *        ZROMEM                                 PAGE 29
 32 040000              *        FNDLIN                                 PAGE 29
 33 040000              *        RTOIN                                  PAGE 30
 34 040000              *        CONINT                                 PAGE 31
 35 040000              *        CUROPT                                 PAGE 33
 36 040000              *        SETOPT                                 PAGE 33
 37 040000              *        FETSV/FETSVA/FNUM                      PAGE 34
 38 040000              *        FETAV/FETAVA                           PAGE 35
 39 040000              *        FETST                                  PAGE 37
 40 040000              *        STOSV                                  PAGE 38
 41 040000              *        STOST                                  PAGE 40
 42 040000              *        EXTERNALS                              PAGE 42
```

>>>> ENTRY POINTS <<<<<<<<

```
040000                          ENT ALLOC
040000                          ENT BCDINT
040000                          ENT COMM
040000                          ENT CONINT
040000                          ENT CRDF?
040000                          ENT CUROP1
040000                          ENT CUROPS
040000                          ENT CUROPT
040000                          ENT DIMAL
040000                          ENT ELSZ
040000                          ENT FETAV
040000                          ENT FETAVA
040000                          ENT FETIND
040000                          ENT FETSV
040000                          ENT FETSVA
040000                          ENT FETSVX
040000                          ENT FNDL2
040000                          ENT FNUM
040000                          ENT FRMNAM
040000                          ENT INIEND
040000                          ENT INIPGM
040000                          ENT INIVAR
040000                          ENT INIVLP
040000                          ENT INT
040000                          ENT JATAB
040000                          ENT MATCH?
040000                          ENT NXTONE
040000                          ENT OPTIO.
040000                          ENT REAL
040000                          ENT RELJMP
040000                          ENT ROMCLA
040000                          ENT ROMCL-
040000                          ENT RTOIN
040000                    *     ENT SETAL0
040000                          ENT SETOPT
040000                          ENT SHORT
040000                          ENT SKP2NA
040000                          ENT SKPVR
040000                          ENT STOST
040000                          ENT STOSTA
040000                          ENT STOSTM
040000                          ENT STOR1
040000                          ENT STOSV
040000                          ENT STOSVM
040000                          ENT VSRCH
040000                          ENT ZERO1-
040000                          ENT ZERO11
040000                          ENT ZERO10
040000                          ENT ZROM-
040000                          ENT ZROMEM
040000
```

```
>>>>>>      ALLOCATION ROUTINES                                    <<<<<<<<
    1 040000                    *
  132 040000                    *  THIS ROUTINE ALLOCATES ALL VARIABLES IN THE
  133 040000                    *  CURRENT PROGRAM (E.G. THE ONE WHOSE PCB IS
  134 040000                    *  POINTED TO BY FWCURR).
  135 040000                    *  THE ROUTINE IS ENTERED BY JSB AND SAVES AND
  136 040000                    *  RESTORES THE  DCM FF STATUS. IT REQUIRES NO
  137 040000                    *  INPUT OTHER THAN THE GLOBAL SYSTEM POINTERS ,
  138 040000                    *  HOWEVER ALL REGISTERS BEGINNING WITH R22 ARE
  139 040000                    *  CONSIDERED VOLATILE.
  140 040000                    *
  141 040000                    *
  142 040000                    *****************************************
  143 040000                    *                                       *
  144 040000                    *  ROMTOK/ROM#/TOKEN                    *
  145 040000                    *                                       *
  146 040000                    *****************************************
  147 040000                    EROM    BSS  0
  148 040000                            DRP  !36
  149 040000 006 343                    POMD R36,-R6        TRASH ONE RETURN
  150 040002 316 014 100                JSB =ROMCLA         GET ROM CLASS
  151 040005 117 310 300                CMB R17,=300        ROM MISSING?
  152 040010 373 077                    JCY NXTBAD          JIF YES
  153 040012 360 037                    JMP NXTGUD          ELSE LOOP
  154 040014 136 251 377 ROMCLA         LDM R36,=GETNXT     JUMP ADDR
  154 040017 377
  155 040020 316 377 377                JSB =EXTRJM         GET ROM CLASS
  156 040023 136 310 057 ROMCL-         CMB R36,=57         ROM CLASS?
  157 040026 372 002                    JNC ROMCLR          JIF NO
  158 040030 223                        CLM R36             ELSE = -1
  159 040031 213                        DCM R36
  160 040032                    * NOTE EXTRJM MAY SET ERRORS
  161 040032 236                ROMCLR  RTN                 AND RETURN
  162 040033                    *****************************************
  163 040033                    *                                       *
  164 040033                    *  INITIALIZATION PART                  *
  165 040033                    *                                       *
  166 040033                    *****************************************
  167 040033                    ALLOC   BSS  0
  168 040033 230                        BIN
  169 040034 262 001 377                STBD R#,=GINTDS     DISABLE INTERRUPTS
  170 040037                    ALL01   BSS  0              BINARY MODE
  171 040037 124 250 010                LDB R24,=10         ROMFL ALLOC
  172 040042 316 377 377                JSB =ALINI          INIT ROUTINE
  173 040045                    *
  174 040045                    *****************************************
  175 040045                    *                                       *
  176 040045                    *  MAIN LOOP                            *
  177 040045                    *                                       *
  178 040045                    *****************************************
  179 040045                    *
  180 040045                    NXTONE  BSS  0
  181 040045 316 377 377                JSB =GETNXT         GET NEXT TOKEN
    2 040050 316 023 100                JSB =ROMCL-         FOR CLASS > 56
```

```
>>>>>>     ALLOCATION ROUTINES                                       <<<<<<<
183 040053             NXTGUD  BSS  0
184 040053 175 261 310         LDMD R75,=PTR1       SAVE FOR ERR HANDLING
184 040056 377
185 040057 263 003 207         STMD R75,=ERRSTP
186 040062 117 310 300         CMB R17,=300         ERR CHK FOR BIN PROG
187 040065 373 022             JCY NXTBAD
188 040067 136 310 030         CMB R36,=30
189 040072 365 351             JPS NXTONE           JIF NOT ALLOCATABLE
190 040074 205                 LLM R36
191 040075 036 265 210         LDMD R36,X36,JATAB GET JUMP ADDR
191 040100 100
192 040101 306 000 000         JSB X36,ZRO          GO ALLOCATE
193 040104 117 310 300         CMB R17,=300         TEST ERRORS
194 040107 372 334             JNC NXTONE           LOOP IF NO ERRORS
195 040111             NXTBAD  BSS 0
196 040111 175 261 172         LDMD R75,=DFPAR1     ERROR IN DEF FN
196 040114 210
197 040115 367 016             JZR  XALL0           IF NO
198 040117 230                 BIN
199 040120 261 160 215         LDMD R75,=YLPEN      ELSE DEALLOC DEF FN
200 040123 315 004 000         SBM  R75,=4,0,0      PT TO 1ST TOKEN
200 040126 000
201 040127 263 310 377         STMD R75,=PTR1
202 040132 316 377 377         JSB  =NEXTD          DEALLOC DEF FN
203 040135 235         XALL0   CLE
204 040136 233                 DCE
205 040137 360 034             JMP XALL2
206 040141
207 040141             *SETALO  BSS  0
208 040141             *        LDB  R40,=40
209 040141             *        LDMD R45,=FWCURR  GET FWCURR
210 040141             *        SBM  R45,=P.TYPE  BACK TO POINT TO P.TYPE
211 040141             *        OCT  0
212 040141             *        STMD R45,=PTR2    SETUP POINTER
213 040141             *        LDBI R45,=PTR2-   FETCH P.TYPE
214 040141             *        ORB  R45,R40      SET ALLOC BIT
215 040141             *        STBI R45,=PTR2    UPDATE P.TYPE
216 040141             *        RTN
217 040141
218 040141             ********************************************
219 040141             *                                          *
220 040141             *         END OF CURRENT LINE              *
221 040141             *         OR DATA STMT                     *
222 040141             *                                          *
223 040141             ********************************************
224 040141 316 377 377 SKPNXT  JSB =SKIPL          SKIP THIS LINE
225 040144 136 006 343 XALL1   POMD R36,-R6         RETURN ADDRESS
226 040147 316 377 377         JSB =NXTLIN
227 040152 370 271             JEN NXTONE           JIF NOT LAST LINE
228 040154 116 220             TSB R16
229 040156 362 014             JOD XALL             JIF CALC
230 040160 175 261 172         LDMD R75,=DFPAR1     DEF ACTIVE?
230 040163 210
```

```
>>>>>>      ALLOCATION ROUTINES                                    <<<<<<<<
   040164 267 006              JZR XALL1A          JIF NO
   040166 316 377 377          JSB =ERROR          NO FNEND
   040171 046                  OCT 38D
   040172 260 315              JMP NXTBAD
   040174
   040174              XALL1A  BSS  0
   040174 235          XALL    CLE
   040175              XALL2   BSS  0
   040175 175 223              CLM  R75            RESET ERRSTP
   040177 263 003 207          STMD R75,=ERRSTP
   040202 262 000 377          STBD R#,=GINTEN     ENABLE INTERRUPTS
   040205 236                  RTN
   040206
   040206              *COMN    BSS  0
   040206              *        LDMD R75,=FWCURR   PT TO P.COM
   040206              *        SBM  R75,=P.COMG
   040206              *        OCT  0
   040206              *        STMD R75,=PTR2
   040206              *        LDMD R75,=FWPRGM   CALC COM SIZE
   040206              *        SBMD 75,=NXTCOM
   040206              *        STMI R75,=PTR2-    STORE IN PLACE
   040206              *        RTN
   040206              ********************************************
   040206              *                                          *
   040206              *            ALLOCATE JUMP TABLE           *
   040206              *                                          *
   040206              ********************************************
   040206 377 377              DEF ROMINI          -1  ROM CLASS > 56
   040210 144 100      JATAB   DEF XALL1           0  EOL
   040212 123 101              DEF VALOC           1  FETCH VAR
   040214 377 377              DEF BININT          2  INTEG CONST
   040216 106 101              DEF SVAL            3  STORE VAR
   040220 377 377              DEF SKPCON          4  REAL CONST
   040222 377 377              DEF SKPNM+          5  STR CONST
   040224 007 111              DEF FUNCAL          6  USER FN CALL
   040226 020 107              DEF LINEAL          7  JTRU LINE #
   040230 020 107              DEF LINEAL          10 GOTO GOSUB
   040232 270 100              DEF RELJMP          11 JMP REL
   040234 370 107              DEF DEFFN           12 USER DEF
   040236 275 100              DEF DEFEND          13 FNEND
   040240 000 100              DEF EROM            14 EXT ROM
   040242 172 111              DEF OPTION          15 OPTION BASE
   040244 275 100              DEF DEFEND          16 FN RET
   040246 371 100              DEF FNASN           17 FN LET
   040250 141 100              DEF SKPNXT          20 DATA
   040252 110 111              DEF DIMAL           21 DIM
   040254 121 111              DEF SHORT           22 SHORT
   040256 114 111              DEF INT             23 INTEGER
   040260 050 111              DEF COMM            24 COM
   040262 020 107              DEF LINEAL          25 ELSE JMP#
   040264 270 100              DEF RELJMP          26 ELSE JMP REL
   040266 020 107              DEF LINEAL          27 USING LINE #
   040270
```

```
>>>>>>          ALLOCATION ROUTINES                                              <<<<<<<<
284 040270                        *****************************************
285 040270                        *                                       *
286 040270                        *  RELJMP TOKEN/VAL/VAL                 *
287 040270                        *                                       *
288 040270                        *****************************************
289 040270                        RELJMP   BSS   0
290 040270 175 271 311                    LDMI R75,=PTR1-    SKIP REL ADDR
290 040273 377
291 040274 236                            RTN
292 040275                        *****************************************
293 040275                        *                                       *
294 040275                        *  FNEND TOKEN/NAME/NAME                *
295 040275                        *                                       *
296 040275                        *****************************************
297 040275 145 261 172 DEFEND     LDMD R45,=DFPAR1   FUNC DEFINED?
297 040300 210
298 040301 367 062                        JZR DFERR          JIF NOT
299 040303 261 310 377                    LDMD R45,=PTR1     SAVE PTR1
300 040306 316 377 377                    JSB  =SKPLN        SKIP TO EOL
301 040311                                DRP  !75
302 040311 261 310 377                    LDMD R75,=PTR1     GET EOL ADDR
303 040314 065 243                        STM  R75,R65       SAVE A COPY
304 040316 261 167 210                    LDMD R75,=JMPADR   GET PTR TO JMP ADR FIELD
305 040321 263 314 377                    STMD R75,=PTR2
306 040324 145 263 310                    STMD R45,=PTR1     RESTORE PTR1
306 040327 377
307 040330 165 325 006                    SBMD R65,=FWCURR   MAKE IT REL ADDR
307 040333 200
308 040334 273 315 377                    STMI R65,=PTR2-    STORE IT
309 040337 261 164 200                    LDMD R65,=CURFUN   GET CUR FUNC ADR
310 040342 273 311 377                    STMI R65,=PTR1-    STORE IT
311 040345 223                            CLM  R65           RESET DEF FUN FLG
312 040346 263 172 210                    STMD R65=DFPAR1
313 040351 263 003 207                    STMD R65,=ERRSTP   RESET ERRSTP BEFORE EXIT
314 040354 260 047 200                    LDBD R65,=CONTOK   ALLOC ON THE FLY ?
315 040357 366 003                        JNZ  DEFEN1        JIF NOT
316 040361 120 006 343                    POMD R20,-R6       ELSE RTN TO INTERPRETER
317 040364 236                 DEFEN1     RTN
318 040365 316 377 377 DFERR      JSB =ERROR+
319 040370 047                            OCT 39D             NOT IN ACTIVE DEF
320 040371
321 040371                        *****************************************
322 040371                        *                                       *
323 040371                        *  FN LET/VAR TOK/NAME/NAME             *
324 040371                        *                                       *
325 040371                        *****************************************
326 040371
327 040371                        FNASN    BSS   0
328 040371 175 261 172                    LDMD R75,=DFPAR1   FUN DEFINED?
328 040374 210
329 040375 367 366                        JZR  DFERR         JIF NOT
330 040377 123 270 311                    LDBI R23,=PTR1-    GET TOKEN
330 040402 377
```

```
    040403 312 200                ADB  R23,=200       SET FUNC
332 040405 316 273 111            JSB  =FRMNAM        GO FORM HEADER
333 040410 175 261 164            LDMD R75,=CURFUN    GET CUR FUNC ADR
333 040413 200
334 040414 323 006 200            ADMD R75,=FWCURR    MAKE IT ABS
335 040417 263 314 377            STMD R75,=PTR2
336 040422 270 315 377            LDBI R75,=PTR2-     GET HEADER
337 040425 037 300                CMB  R75,R37        SAME TYPE?
338 040427 366 334                JNZ  DFERR          JIF NOT
339 040431 271 311 377            LDMI R75,=PTR1-     SKIP ADDR
340 040434 271 315 377            LDMI R75,=PTR2-     GET FUNC NAME PTR
341 040437 323 006 200            ADMD R75,=FWCURR    MAKE IT ABS
342 040442 263 314 377            STMD R75,=PTR2
343 040445 261 310 377            LDMD R75,=PTR1      SAVE PTR1
344 040450 316 311 107            JSB  =MATCH?        SAME NAME?
345 040453 371 310                JEZ  DFERR          JIF NOT
346 040455                        DRP  !65
347 040455 263 310 377            STMD R65,=PTR1      RESTORE PTR1
348 040460 261 164 200            LDMD R65,=CURFUN    GET CURR FUN ADDR
349 040463 273 310 377            STMI R65,=PTR1      SAVE IN PLACE
350 040466                 LABEL  BSS  0
351 040466                 VALRTN BSS  0
352 040466 316 377 377            JSB  =SKPNM+        SKIP THE FUN NAME
353 040471 236                    RTN
354 040472                 ***************************************************
355 040472                 *                                                 *
356 040472                 *      VARIABLE ALLOCATED SO STORE ADDR           *
357 040472                 *                                                 *
358 040472                 ***************************************************
359 040472                 FNDIT  BSS  0
360 040472 316 274 101            JSB  =RNMOD1        RESTORE NXT TOK ADDR
361 040475 145 325 006            SBMD R45,=FWCURR    MAKE ADDR RELATIVE
361 040500 200
362 040501 273 310 377            STMI R45,=PTR1      STORE ADDR
363 040504 360 360                JMP  VALRTN         PASS OVER NAME
364 040506
365 040506                 ***************************************************
366 040506                 *                                                 *
367 040506                 *  SUBROUTINE VALOC                               *
368 040506                 *                                                 *
369 040506                 *                                                 *
370 040506                 *  THIS ROUTINE IS CALLED TO ALLOCATE A VARIABLE. *
371 040506                 *  IT WILL BE ALLOCATED IN THE CURRENT PROGRAM    *
372 040506                 *  AREA IF RUN, ELSE IT WILL BE ALLOCATED IN      *
373 040506                 *  THE  AREA POINTED TO BY LAVAIL. IF RUN, THE    *
374 040506                 *  VARIABLE WILL BE ASSIGNED AN ADDRESS RELATIVE  *
375 040506                 *  TO FWCURR . OTHERWISE AN ABSOLUTE ADDRESS WILL *
376 040506                 *  BE ASSIGNED.                                   *
377 040506                 *                                                 *
378 040506                 *                                                 *
379 040506                 *  INPUT REGISTERS                                *
380 040506                 *                                                 *
  1 040506                 *    R16     =  CALC MODE FLAG                    *
```

```
>>>>>       ALLOCATION ROUTINES                                          <<<<<<<<
040506                    *     R20       =  SCRATCH                                 *
040506                    *     R22       =  ZERO IF DIM                             *
040506                    *     R23       =  TOKEN                                   *
040506                    *     R24       =  PTR TO NAME IN STACK                    *
040506                    *     R26       =  FWA VARIABLES(IF RUN)                   *
040506                    *     R30       =  NEXT VARIABLE(IF RUN)                   *
040506                    *     R40-R44   =  PCB 7 - 11( IF RUN)                     *
040506                    *     R45-R47   =  LINE # AND BYTES(IF RUN)                *
040506                    *                                                          *
040506                    *  OTHER REGISTERS USED                                    *
040506                    *                                                          *
040506                    *     R32       =  CURRENT VRBL PTR FOR SEARCH             *
040506                    *     R34       =  SCRATCH                                 *
040506                    *     R36       =  VARIABLE NAME FORM                      *
040506                    *     R50-R77   =  SCRATCH                                 *
040506                    *                                                          *
040506                    ************************************************************
040506                    *  NOTE - A( ) AND A( , ) ALSO COME HERE WITH              *
040506                    *  TOKENS 37 AND 40 RESPECTIVELY. FOR THESE                *
040506                    *  TWO CASES THE TOKEN IS FORCED TO 2 FOR THE              *
040506                    *  FRMNAM CALL AND THEN RESTORED. IF THE                   *
040506                    *  ARRAY HAS NOT ALREADY BEEN DECLARED,e.g.VSRCH           *
040506                    *  FAILS, THEN NARDIM CHECKS FOR TOKENS 37                 *
040506                    *  AND 40 AND ALLOCATES THE DEFAULT ARRAY.                 *
040506                    ************************************************************
040506                    *                                                          *
040506                    *  VAR TOKEN/NAME/NAME                                     *
040506                    *                                                          *
040506                    ************************************************************
040506                    SVAL    BSS  0
040506 123 310 257                CMB  R23,=257      LABEL TOKEN ?
040511 367 353                    JZR  LABEL         JIF YES
040513 310 040                    CMB  R23,=40       STRING ARRAY?
040515 366 002                    JNZ  SVAL1         JIF NOT
040517 250 024                    LDB  R23,=24       BUILD A PSEUDO IMAGE
040521                    SVAL1   BSS  0
040521                            DRP  !23
040521 314 020                    SBB  R23,=20
040523                    VALOC   BSS  0
040523 145 271 311                LDMI R45,=PTR1-    SKIP ADDR
040526 377
040527 364 335                    JNG  VALRTN        JIF ALREADY ALLOCATED
040531 123 006 344                PUBD R23,+R6       SAVE TOKEN
040534 310 200                    CMB  R23,=200      A$( ),A$( , ),A( ),A( , ) ?
040536 372 013                    JNC  VALOC2        JIF NOT
040540 310 340                    CMB  R23,=340      A( ) OR A( , ) ?
040542 373 004                    JCY  VALOC1        JIF YES
040544 250 004                    LDB  R23,=4        FOR A$( ) OR A$( , )
040546 360 003                    JMP  VALOC2
040550                    VALOC1  BSS  0
040550 123 250 002                LDB  R23,=2        FOR A( ) OR A( , )
040553 316 273 111        VALOC2  JSB  =FRMNAM       GO FORM NAME
040556 123 006 342                POBD R23,-R6       RESTORE TOKEN
```

```
>>>>>>      ALLOCATION ROUTINES                                    <<<<<<<
    040561 145 261 310          LDMD R45,=PTR1     SAVE NAME ADDR
434 040564 377
435 040565 263 164 210          STMD R45,=SAVLAV
436 040570 261 172 210          LDMD R45,=DFPAR1   CHECK IF IN DEF FN
437 040573 367 016              JZR NTDEF    JIF NOT IN DEF
438 040575                  *************************************************
439 040575                  *                                               *
440 040575                  *            VARIABLES WITHIN DEFS              *
441 040575                  *                                               *
442 040575                  *************************************************
443 040575 155 261 175          LDMD R55,=DFPAR2   GET END OF PAR ADDR
443 040600 210
444 040601 040 243              STM  R55,R40
445 040603 163 222              CLB  R63           FLAG FN PARS
446 040605 210                  ICB  R63
447 040606 316 304 101          JSB  =VSRCH        SEARCH FUNC PARS
448 040611 370 257              JEN  FNDIT         JIF FOUND
449 040613                  *************************************************
450 040613                  *                                               *
451 040613                  *       SEE IF THERE IS AN ALLOCATED PROGRAM    *
452 040613                  *                                               *
453 040613                  *************************************************
454 040613                  NTDEF   BSS  0
455 040613 145 261 017          LDMD R45,=EOVAR    GET END OF VAR ADDR
455 040616 200
456 040617 040 243              STM  R45,R40
457 040621 261 014 200          LDMD R45,=BOVAR    GET BEGIN OF VAR ADDR
458 040624                  *************************************************
459 040624                  *                                               *
460 040624                  *       SEARCH FWCURR VARIABLE AREA             *
461 040624                  *                                               *
462 040624                  *************************************************
463 040624 163 222              CLB R63              FLG NOT FN PARS
464 040626 316 304 101          JSB =VSRCH           SEARCH FOR VARIABLE
465 040631 371 012              JEZ NOFIND           JIF NOT FOUND
466 040633 122 260 160 TSTDUP   LDBD R22,=DIMDIM     CIF DIM
466 040636 210
467 040637 366 231              JNZ FNDIT            JIF NOT
468 040641 316 377 377          JSB =ERROR+
469 040644 043                  OCT 35D              COM OR DIM EXISTING VRBL
470 040645
471 040645 116 220      NOFIND  TSB R16
472 040647 363 020              JEV RNMOD            JIF NOT CALC
473 040651                  *************************************************
474 040651                  *                                               *
475 040651                  *       SEARCH CALC MODE VARIABLE AREA          *
476 040651                  *                                               *
477 040651                  *************************************************
478 040651 145 261 030          LDMD R45,=CALVRB   GET END OF CAL VAR ADDR
478 040654 200
479 040655 040 243              STM  R45,R40
480 040657 261 025 200          LDMD R45,=LAVAIL   GET BEGIN OF CAL VAR ADDR
  1 040662 163 222              CLB  R63           FLG NOT FN PARS
```

```
>>>>>       ALLOCATION ROUTINES                                          <<<<<<<<<
040664 316 304 101          JSB =VSRCH
040667                  *  NOW IF FOUND, GO CHECK DUP DIM
040667                  *  TO CATCH INTEGER A  SHORT A IN CALC MODE
040667 370 342              JEN TSTDUP          JIF FOUND
040671                  ********************************************************
040671                  *                                                      *
040671                  *       VARIABLE NOT FOUND SO ALLOCATE                 *
040671                  *                                                      *
040671                  ********************************************************
040671 316 164 102 $NMOD    JSB  =VSTOR         STORE VARIABLE
040674                  $NMOD1  BSS  0
040674 175 261 164          LDMD R75,=SAVLAV    RESTORE NXT TOK ADDR
040677 210
040700 263 310 377          STMD R75,=PTR1
040703 236                  RTN
040704
040704                  ********************************************************
040704                  *                                                      *
040704                  *  SUBROUTINE VSRCH                                    *
040704                  *                                                      *
040704                  *    IN:                                               *
040704                  *      PTR1=POINTS TO NAME                             *
040704                  *      R36=CLASS                                       *
040704                  *      R45=BEGINING OF VAR SPACE                       *
040704                  *      R40=END OF VAR SPACE                            *
040704                  *                                                      *
040704                  *    OUT:                                              *
040704                  *      IF FOUND THEN                                   *
040704                  *           E#0                                        *
040704                  *           R45=POINTS TO THE VAR                      *
040704                  *           PTR1=POINTS TO AFTER NAME                  *
040704                  *                                                      *
040704                  *      IF NOT FOUND THEN                               *
040704                  *           E=0                                        *
040704                  *           R45=POINTS TO NEXT SPACE                   *
040704                  *           PTR1=POINTS TO NAME                        *
040704                  *                                                      *
040704                  ********************************************************
040704                  VSRCH   BSS  0
040704 137 317 311          ANM R37,=311        CLEAR INT & SHORT
040707 235              VLOOP   CLE                 SET NOT FOUND
040710 145 040 301          CMM  R45,R40        SEE IF DONE
040713 367 076              JZR VSRTN           EXIT IF DONE
040715 372 074              JNC VSRTN
040717 234                  ICE                 SET FOUND
040720 263 314 377          STMD R45,=PTR2      SET NEXT VAR LOCATION
040723 167 270 315          LDBI R67,=PTR2-     GET HEADER
040726 377
040727 317 311              ANM  R67,=311       OFF REAL,INT,SHORT,LOC,TRACE
040731 037 300              CMB  R67,R37        SAME CLASS?
040733 366 051              JNZ  SKIP1          JIF NO
040735 163 220              TSB  R63            SRCH FN PAR LISTS ?
040737 367 011              JZR  VSR1           JIF NO
```

```
>>>     ALLOCATION ROUTINES                                        <<<<<<<<
040741 165 271 315           LDMI R65,=PTR2-   GET NAME POINTER
040744 377
040745 323 160 215           ADMD R65,=YLPEN   MAKE IT ABSOLUTE
040750 760 024               JMP  VSR2
040752               VSR1    BSS  0
040752 165 271 315           LDMI R65,=PTR2-   GET NAME POINTER
040755 377
040756 145 321 022           CMMD R45,=NXTMEM  PRGM VARIABLE ?
040761 200
040762 373 006               JCY  VSR1.5       JIF YES
040764 165 323 030           ADMD R65,=CALVRB  CALC VAR IS REL TO CALVRB
040767 200
040770 760 004               JMP  VSR2
040772               VSR1.5  BSS  0
040772 165 323 006           ADMD R65,=FWCURR  GET ABS ADDR
040775 200
040776               VSR2    BSS  0
040776                       DRP  !65
040776 263 314 377           STMD R65,=PTR2    SET UP POINTER
041001 316 311 107           JSB  =MATCH?      NAME MATCH?
041004 370 005               JEN  VSRTN        JIF YES
041006               SKIP1   BSS  0
041006 316 020 102           JSB  =SKPVR       SKIP THIS VAR
041011 760 274               JMP  VLOOP        LOOP BACK
041013               VSRTN   BSS  0
041013 137 322 015           ADBD R37,=DIMFLG  RESTORE CLASS
041016 204
041017 236                   RTN
041020
041020                       ***********************************************************
041020                       *                                                         *
041020                       *     SKIP CURRENT VARIABLE                               *
041020                       *                                                         *
041020                       *        IN:                                              *
041020                       *          R45=POINTS TO CURRENT VAR                      *
041020                       *                                                         *
041020                       *        OUT:                                             *
041020                       *          R45=POINTS TO NEXT VAR OR SPACE                *
041020                       *                                                         *
041020                       ***********************************************************
041020               SKPVR   BSS  0
041020 165 223               CLM  R65          PRESET FOR REMOTE
041022 145 263 314           STMD R45,=PTR2
041025 377
041026 171 271 315           LDMI R71,=PTR2-   GET TOT,NAME PTR,CLASS
041031 377
041032 177 220               TSB  R77
041034 362 011               JOD  YESFN        JIF FUNC?
041036 135 077 240           LDB  R35,R77      COPY HEADER
041041 317 004               ANM  R35,=4       ISOLATE REMOTE
041043 366 110               JNZ  VRTN         JIF REMOTE
041045 760 015               JMP  NOTFN
041047               YESFN   BSS  0
```

```
041047 177 006 344              PUBD R77,+R6      SAVE HEADER
041052 170 271 315              LDMI R70,=PTR2-   TRACH RTN,PCR,TOS,MEM,CSTAT
041055 377
041056 171 271 315              LDMI R71,=PTR2-   AND GET TOT
041061 377
041062 177 342                  POBD R77,-R6      RESTORE HEADER
041064                  NOTFN   BSS  0
041064 177 065 242              STB  R77,R65      SAVE CLASS
041067 310 100                  CMB  R77,=100     STRING OR ARRAY?
041071 373 022                  JCY  SARRAY       JIF YES
041073 231                      BCD               GET TYPE
041074 165 206                  LRB  R65
041076 317 003 000              ANM  R65,=3,0,0   ISOLATE
041101 000
041102 230                      BIN
041103 366 004                  JNZ  NOTRL        JIF NOT REAL
041105 250 005                  LDB  R65,=5       5 BYTES MORE TO SKIP
041107 360 043                  JMP  VRTN+
041111                  NOTRL   BSS  0
041111 165 212                  DCB  R65
041113 360 037                  JMP  VRTN+
041115                  SARRAY  BSS  0
041115 165 071 241              LDM  R65,R71      GET TOT
041120 177 310 200              CMB  R77,=200     STRING/STRING FUNC?
041123 373 007                  JCY  STRNG        JIF YES
041125 165 313 004              ADM  R65,=4,0,0   SKIP 4 MORE
041130 000 000
041132 360 020                  JMP  VRTN+
041134                  STRNG   BSS  0
041134                          DRP  !77
041134 310 300                  CMB  R77,=300     STRING ARRAY?
041136 373 007                  JCY  STRARY       JIF YES
041140 165 313 002              ADM  R65,=2,0,0   SKIP 2 MORE
041143 000 000
041145 360 005                  JMP  VRTN+
041147                  STRARY  BSS  0
041147 165 313 006              ADM  R65,=6,0,0   SKIP 6 MORE
041152 000 000
041154                  VRTN+   BSS  0
041154                          DRP  !65
041154 215                      TCM  R65          NEGATE IT
041155                  VRTN    BSS  0
041155 165 323 314              ADMD R65,=PTR2    PASS OVER VAR
041160 377
041161 045 243                  STM  R65,R45
041163 236                      RTN
041164
041164                  ***********************************************
041164                  *                                             *
041164                  *  SUBROUTINE VSTOR                           *
041164                  *                                             *
041164                  *  THIS ROUTINE STORES THE VARIABLE NAME      *
041164                  *  AND  DESCRIPTION IN THE VARIABLE AREA.     *
```

```
>>>>         ALLOCATION ROUTINES                                            <<<<<<<<
 041164                        *                                                 *
 041164                        ***************************************************
 041164                        VSTOR BSS 0
 041164 106 263 150                  STMD R6,=XLPEN     SAVE R6 FOR ERR HANDLING
 041167 215
 041170 145 261 310                  LDMD R45,=PTR1     GET NAME ADDR
 041173 377
 041174 325 006 200                  SBMD R45,=FWCURR   GET REL ADDR
 041177 044 243                      STM  R45,44        MOVE UP ONE BYTE
 041201 147 037 240                  LDB  R47,R37       GET FLAGS
 041204 130 240                      LDB  R30,R37       COPY HEADER
 041206 231                          BCD
 041207 206                          LRB  R30           SHIFT 5 BITS
 041210 230                          BIN
 041211 206                          LRB  R30
 041212 317 006 000                  ANM  R30,=6,0      ISOLATE TYPE
 041215 030 265 225                  LDMD R30,X30,STRTBL STORE TABLE
 041220 102
 041221 306 000 000                  JSB  X30,ZRO       JUMP TO STORE
 041224 236                          RTN
 041225                        STRTBL  BSS  0
 041225 101 103                      DEF  STSNUM        STORE SIMPLE NUM
 041227 143 103                      DEF  STNUAR        STORE NUM ARRAY
 041231 100 104                      DEF  STSSTR        STORE SIMPLE STRING
 041233 143 103                      DEF  STSTAR        STORE STRING ARRAY
 041235
 041235                        ******************************************************
 041235                        *                                                    *
 041235                        *    CALCULATE TOTAL # OF BYTES IN ARRAY             *
 041235                        *                                                    *
 041235                        *       IN:                                          *
 041235                        *         R54=TOTAL # OF ELEMENTS (4 BYTES)          *
 041235                        *         R66=LENGTH OF EACH ELEMENT                 *
 041235                        *                                                    *
 041235                        *       OUT:                                         *
 041235                        *         R55=TOTAL # OF BYTES  (3 BYTES)            *
 041235                        *                                                    *
 041235                        *       NOTE:                                        *
 041235                        *         ERROR FLAG MAY BE SET IF ARRAY TOO LARGE   *
 041235                        *                                                    *
 041235                        ******************************************************
 041235
 041235                        ARYSPC  BSS  0
 041235 154 074 243                  STM  R54,R74       SAVE A COPY
 041240 316 377 377                  JSB  =INTMUL       MUL HI ORDER
 041243 155 221                      TSM  R55           TOO LARGE?
 041245 366 030                      JNZ  MOVFL         JIF YES
 041247 154 070 242                  STB  R54,R70       SAVE IT
 041252 176 074 241                  LDM  R76,R74       GET LOW ORDER 2 BYTES
 041255 316 377 377                  JSB  =INTMUL       MUL LOW ORDER
 041260 157 220                      TSB  R57           TOO LARGE?
 041262 366 013                      JNZ  MOVFL         JIF YES
 041264 175 054 241                  LDM  R75,R54       SHIFT LOW ORDER UP
```

```
>>>>>      ALLOCATION ROUTINES                                    <<<<<<<<
041267 055 243              STM  R75,R55
041271 157 070 302          ADB  R57,R70       ADD HI ORDER BYTE
041274 373 001              JCY  MOVFL         JIF TOO LARGE
041276 236                  RTN
041277                MOVFL BSS  0
041277 106 261 150          LDMD R6,=XLPEN     CLEAN UP R6 STACK
041302 215
041303 316 377 377          JSB  =ERROR+       MEM OVERFLOW
041306 023                  OCT  19D

041307                      ********************************************************
041307                      *                                                      *
041307                      *    GET SUBSCRIPT                                     *
041307                      *                                                      *
041307                      *      IN:                                             *
041307                      *        PTR1=POINTS TO THE SUBSCRIPT                  *
041307                      *                                                      *
041307                      *      OUT:                                            *
041307                      *        R76=SUBSCRIPT IN BINARY                       *
041307                      *        PTR1=POINTS TO NEXT ITEM                      *
041307                      *                                                      *
041307                      ********************************************************

041307                GSBSCR BSS  0
041307 316 204 104          JSB  =INTG?        NXT TOKEN INTEGER ?
041312 266 016              JNZ  GSBSR         JIF NOT
041314 165 270 311          LDBI R65,=PTR1-    SKIP INT TOKEN
041317 377
041320 271 311 377          LDMI R65,=PTR1-    GET INT VALUE
041323                BCDINT BSS  0
041323 316 377 377          JSB  =INTORL       CONVERT TO REAL
041326 230                  BIN
041327 316 116 112          JSB  =CONINT       CONVERT TO BINARY
041332 236                GSBSR RTN

041333                      ********************************************************
041333                      *                                                      *
041333                      *    ALLOCATE DEFAULT ARRAY                            *
041333                      *                                                      *
041333                      *      ENTRY NAME IS "NARDIM"                          *
041333                      *                                                      *
041333                      *      IN:                                             *
041333                      *        PTR1=POINTS TO NEXT TOKEN                     *
041333                      *                                                      *
041333                      *      OUT:                                            *
041333                      *        R30=DEFAULT # OF ROW                          *
041333                      *        R76=DEFAULT # OF COL ( IF TWO DIM )           *
041333                      *        R65=ADDR OF NEXT TOKEN                        *
041333                      *                                                      *
041333                      ********************************************************

041333
041333
041333                SAVIT BSS  0
```

```
>>>>      ALLOCATION ROUTINES                                    <<<<<<<<
    041333 175 261 171          LDMD R75,=RSTAR    ALREADY FOUND A VAR?
713 041336 200
714 041337 766 007              JNZ  SAVIT1        JIF YES
715 041341 261 310 377          LDMD R75,=PTR1     SET FOUND
716 041344 211                  ICM  R75           BACKUP TO THE TOKEN
717 041345 263 171 200          STMD R75,=RSTAR    SAVE IT
718 041350             SAVIT1   BSS  0
719 041350 175 271 311          LDMI R75,=PTR1-    SKIP ADDR
719 041353 377
720 041354 316 377 377          JSB  =SKPNM+       SKIP NAME
721 041357 123 310 002          CMB  R23,=2        ARRAY?
722 041362 766 014              JNZ  NARD1         JIF NOT
723 041364             NARDIM   BSS  0
724 041364 123 310 300          CMB  R23,=300      A() OR A(,) ?
725 041367 373 060              JCY  STOR()        JIF YES
726 041371 260 174 200          LDBD R23,=CSTAR    INC COUNT
727 041374 210                  ICB  R23
728 041375 262 174 200          STBD R23,=CSTAR
    041400             NARD1    BSS  0
73  041400 316 377 377          JSB  =GETNXT       GET NEXT TOKEN
731 041403 136 221              TSM  R36           TST CLASS
732 041405 367 173              JZR  ILLDIM        JIF ILLEGAL
733 041407             SFSG     BSS  0
734 041407 310 032              CMB  R#,=32        CIF SUBS DIM 11-14?
735 041411 367 024              JZR  TSTINX        JIF YES
736 041413 310 002              CMB  R#,=2         INTEGER?
    041415 366 005              JNZ  NOTIN         JIF NOT
738 041417 316 377 377          JSB  =BININT       SKIP INTEGER
739 041422 360 354              JMP  NARD1
740 041424             NOTIN    BSS  0
741 041424 310 004              CMB  R#,=4         VARIABLE
742 041426 364 303              JNG  SAVIT         JIF YES
743 041430 366 346              JNZ  NARD1         JIF NOT CONSTANT
744 041432 316 377 377          JSB  =SKPCON       SKIP THE CONSTANT
745 041435 360 341              JMP  NARD1
746 041437             TSTINX   BSS  0
747 041437 167 260 174          LDBD R67,=CSTAR    GET COUNT
    041442 200
74  041443 212                  DCB  R67           DEC COUNT
749 041444 262 174 200          STBD R67,=CSTAR    UPDATE IT
750 041447 366 327              JNZ  NARD1         JIF NOT MATCH
751 041451             STOR()   BSS  0
752 041451 130 251 012          LDM  R30,=12,0     DEFAULT 10 ROWS
752 041454 000
753 041455 076 243              STM  R30,R76       DEFAULT 10 COLS
754 041457 123 220              TSB  R23           ONE DIM?
755 041461 363 004              JEV  STOR0         JIF NOT
756 041463 176 251 377          LDM  R76,=377,377  SET ONE DIM
756 041466 377
757 041467             STOR0    BSS  0
758 041467 165 261 171          LDMD R65,=RSTAR    FIND A VAR
758 041472 200
  9 041473             ********JNZ  STORTN        JIF YES
```

```
> >>>>          ALLOCATION ROUTINES                                                   <<<<<<<<
    041473                     ********LDMD R65,=PTR1     ELSE USE POINTER
    041473 367 003                     JZR  STORTN        5-21-81 PAR IN PTR1
    041475 263 310 377                 STMD R65,=PTR1     5-21-81 INSTEAD OF R65
    041500 236                 STORTN  RTN
    041501
    041501                     **********************************************************
    041501                     *                                                        *
    041501                     *    STORE SIMPLE NUMERICAL VARIABLES                    *
    041501                     *                                                        *
    041501                     *      IN:                                               *
    041501                     *        R44-46=POINTER TO NAME                          *
    041501                     *        R47=HEADER FLAGS                                *
    041501                     *        PTR1=POINTS TO NAME                             *
    041501                     *                                                        *
    041501                     *      OUT:                                              *
    041501                     *        VAR SPACE ALLOCATED & HEADER STORED             *
    041501                     *        PTR1=POINTS TO NEXT TOKEN                       *
    041501                     *                                                        *
    041501                     **********************************************************
    041501
    041501                     STSNUM  BSS  0
    041501 175 261 310                 LDMD R75,=PTR1     SAVE CURRENT POINTER
    041504 377
    041505 006 345                     PUMD R75,+R6
    041507 316 377 377                 JSB  =SKPNM+       SKIP NAME
    041512                             DRP  !75           POINTS TO NEXT TOKEN
    041512 263 164 210                 STMD R75,=SAVLAV   SAVE IT
    041515 006 343                     POMD R75,-R6       RESTORE NAME ADDR
    041517 263 310 377                 STMD R75,=PTR1
    041522 316 213 104                 JSB  =ELSZ         GET ELEMENT SIZE
    041525 155 223                     CLM  R55           RESET TOTAL SIZE
    041527 170 223                     CLM  R70           RESET FOR SIZE
    041531 166 055 243                 STM  R66,R55       SET TOTAL SIZE
    041534 070 243                     STM  R66,R70       SET SIZE
    041536 316 150 105                 JSB  =STOR1        STORE THE VAR
    041541 360 133                     JMP  STNUAS        INITIALIZE TO NULL
    041543
    041543                     **********************************************************
    041543                     *                                                        *
    041543                     *    STORE NUMERICAL ARRAY                               *
    041543                     *                                                        *
    041543                     *      IN:                                               *
    041543                     *        R44-46=POINTER TO NAME                          *
    041543                     *        R47=HEADER OF FLAGS                             *
    041543                     *        PTR1=POINTS TO NAME                             *
    041543                     *                                                        *
    041543                     *      OUT:                                              *
    041543                     *        ARRAY ALLOCATED                                 *
    041543                     *                                                        *
    041543                     **********************************************************
    041543
    041543                     STSTAR  BSS  0             STORE STRING ARRAY
    041543                     STNUAR  BSS  0             STORE NUMERIC ARRAY
```

```
>>>>>      ALLOCATION ROUTINES                                    <<<<<<<<<
   041543 316 377 377          JSB  =SKPNM+        SKIP NAME
   041546 316 140 111          JSB  =SETOPT        DEFAULT OPT BAS IF NECESSARY
   041551              STNUA1  BSS  0
   041551 174 223              CLM  R74            RESET RSTAR & CSTAR
   041553 263 171 200          STMD R74,=RSTAR
   041556 122 260 160          LDBD R22,=DIMDIM    DIM?
   041561 210
   041562 266 116              JNZ  STNUA+         JIF NOT
   041564 316 307 102          JSB  =GSBSCR        GET 1ST SUBSCRIPT
   041567 176 030 243          STM  R76,R30        SAVE IT
   041572 251 377 377          LDM  R76,=377,377   PRESET FOR 1 DIM
   041575 316 307 102          JSB  =GSBSCR        GET 2ND SUBSCRIPT
   041600 360 103              JMP  STNUA2
   041602
   041602              ILLDIM  BSS 0
   041602 316 377 377          JSB =ERROR+
   041605 044                  OCT 36D              ILLEGAL DIM
   041606
   041606              STNUA3  BSS  0
   041606                      DRP  176
   041606 006 345              PUMD R76,+R6        SAVE MAX COL
   041610 130 345              PUMD R30,+R6        SAVE MAX ROW
   041612 323 175 200          ADMD R30,=OPTBAS
   041615 054 243              STM  R30,R54        # OF ELEMENT=MAX ROW + BASE
   041617 156 223              CLM  R56
   041621              STNUA4  BSS  0
   041621 147 220              TSB  R47            STRING ARRAY ?
   041623 364 125              JNG  STSTA1         JIF YES
   041625 165 261 310          LDMD R65,=PTR1      GET NXT TOKEN ADDR
   041630 377
   041631 316 274 101          JSB  =RNMOD1        RESTORE NAME ADDR
   041634 165 263 164          STMD R65,=SAVLAV    SAVE NXT TOKEN ADDR
   041637 210
   041640 316 213 104          JSB  =ELSZ          GET ELEMENT SIZE
   041643 316 235 102          JSB  =ARYSPC        CALC ARRAY SIZE
   041646 175 055 241          LDM  R75,R55        COPY SIZE
   041651 313 007 000          ADM  R75,=7,0,0     SIZE OF TOT,ROW,COL
   041654 000
   041655 070 243              STM  R75,R70        SAVE IT
   041657 316 150 105          JSB  =STOR1         STORE ARRAY
   041662 175 055 241          LDM  R75,R55        GET TOT BYTES
   041665 164 006 343          POMD R64,-R6        RESTORE MAX ROW +COL
   041670 071 243              STM  R64,R71        PACK IN PLACE
   041672 171 273 315          STMI R71,=PTR2-     STORE IN PLACE
   041675 377
   041676              STNUA5  BSS  0
   041676 316 237 104          JSB  =INIV          INITIALIZE ARRAY
   041701 236                  RTN
   041702
   041702              STNUA+  BSS  0
   041702 316 364 102          JSB  =NARDIM        ALLOCATE DEFAULT ARRAY
   041705
   041705              STNUA2  BSS  0
```

```
>>>>      ALLOCATION ROUTINES                                    <<<<<<<<
041705 166 261 175          LDMD R66,=OPTBAS
041710 200
041711 215                  TCM  R66
041712 030 301              CMM  R66,R30        WITH SUBSCRIPT=0
041714 367 264              JZR  ILLDIM         JIF YES
041716 176 311 377          CMM  R76,=377,377 ONE DIM?
041721 377
041722 367 262              JZR  STNUA3         JIF YES
041724 166 076 301          CMM  R66,R76        2ND SUBS=0
041727 367 251              JZR  ILLDIM         JIF YES
041731                BASE0 BSS  0
041731 176 006 345          PUMD R76,+R6        SAVE MAX COL
041734 130 345              PUMD R30,+R6        SAVE MAX ROW
041736 166 030 241          LDM  R66,R30
041741 323 175 200          ADMD R66,=OPTBAS    ADD BASE
041744 176                  DRP  76
041745 316 141 105          JSB  =TOTEL1
041750 360 247              JMP  STNUA4
041752
041752                STSTA1 BSS  0
041752 176 251 022          LDM  R76,=18D,0     DEFAULT LEN
041755 000
041756 122 260 160          LDBD R22,=DIMDIM    DEFAULT ARRAY ?
041761 210
041762 366 021              JNZ  STSTA2         JIF YES
041764 270 311 377          LDBI R22,=PTR1-     SKIP TOKEN (11-14)
041767 316 204 104          JSB  =INTG?         NEXT TOKEN AN INT?
041772 366 011              JNZ  STSTA2         JIF NOT
041774 154 006 345          PUMD R54,+R6        SAVE REG
041777 316 307 102          JSB  =GSBSCR        GET MAX LEN
042002 154 006 343          POMD R54,-R6        RESTORE REG
042005
042005                STSTA2 BSS  0
042005 176 036 243          STM  R76,R36        SAVE MAX LEN
042010 165 261 310          LDMD R65,=PTR1      GET NXT TOKEN ADDR
042013 377
042014 316 274 101          JSB  =RNMOD1        RESTORE NAME ADDR
042017 165 263 164          STMD R65,=SAVLAV    SAVE NXT TOKEN ADDR
042022 210
042023 166 036 241          LDM  R66,R36        GET MAX LEN
042026 313 002 000          ADM  R66,=2,0       +2 BYTES FOR ACT
042031 316 235 102          JSB  =ARYSPC        CALC TOT # OF BYTES
042034 175 055 241          LDM  R75,R55        COPY IT
042037 313 011 000          ADM  R75,=9D,0,0    FOR TOT,ROW,COL,MAX LEN
042042 000
042043 070 243              STM  R75,R70        SETUP
042045 316 150 105          JSB  =STOR1         ALLO FOR THE ARRAY
042050 155 273 315          STMI R55,=PTR2-     STORE TOT # OF BYTES
042053 377
042054 164 006 343          POMD R64,-R6        RESTORE ROW,COL
042057 273 315 377          STMI R64,=PTR2-     STORE IT
042062 175 223              CLM  R75            RESET
042064 136 273 315          STMI R36,=PTR2-     STORE MAX LEN
```

```
042067 377
042070 211                     ICM  R36            ELEM LEN = MAX LEN + 2
042071 211                     ICM  R36
042072 075 243                 STM  R36,R75        ELEM LEN
042074                 STSTA3  BSS  0
042074 316 311 104             JSB  =INISTR        INIT STRING ARRAY
042077 236                     RTH
042100
042100                 ***********************************************************
042100                 *                                                         *
042100                 *     STORE SIMPLE STRING VARIABLE                        *
042100                 *                                                         *
042100                 ***********************************************************
042100
042100                 STSSTR  BSS  0
042100 175 261 310             LDMD R75,=PTR1      SAVE POINTER
042103 377
042104 006 345                 PUMD R75,+R6
042106 316 377 377             JSB  =SKPNM+        SKIP NAME
042111 176 251 022             LDM  R76,=18D,0     DEFAULT SIZE
042114 000
042115 122 260 160             LDBD R22,=DIMDIM    DIM?
042120 210
042121 366 003                 JNZ  STSST1         JIF NOT
042123 316 307 102             JSB  =GSBSCR        GET SPECIFIED SIZE
042126                 STSST1  BSS  0
042126 165 261 310             LDMD R65,=PTR1      SAVE NEXT TOKEN ADDR
042131 377
042132 263 164 210             STMD R65,=SAVLAV
042135 006 343                 POMD R65,-R6        RESTORE NAME ADDR
042137 263 310 377             STMD R65,=PTR1
042142 176 345                 PUMD R76,+R6        SAVE MAX LEN
042144 313 002 000             ADM  R76,=2,0       + 2 BYTES OF ACT LEN
042147 055 243                 STM  R76,R55        SETUP
042151 157 222                 CLB  R57
042153 175 241                 LDM  R75,R55
042155 313 005 000             ADM  R75,=5,0,0     LEN OF TOT,MAX
042160 000
042161 070 243                 STM  R75,R70        SETUP
042163 316 150 105             JSB  =STOR1         STORE IT
042166 155 075 243             STM  R55,R75        SAVE IT
042171 130 006 343             POMD R30,-R6        POP MAX LEN
042174 053 243                 STM  R30,R53        SETUP HEADER
042176 153 273 315             STMI R53,=PTR2-     STORE IT
042201 377
042202 360 270                 JMP  STSTA3         INIT STRING TO NULL
042204
042204                 ***********************************************************
042204                 *                                                         *
042204                 *     CHECK IF NEXT TOKEN IS AN INTEGER                   *
042204                 *                                                         *
042204                 *       IN:                                               *
042204                 *         PTR1=POINTS TO NEXT TOKEN                       *
```

```
>>>>>>>      ALLOCATION ROUTINES                                        <<<<<<<<
042204                       *
042204                       *      OUT:                                       *
042204                       *        Z=1 IF IT IS INTEGER                     *
042204                       *          0 IF IT IS NOT                         *
042204                       *        PTR1 NOT CHANGED                         *
042204                       *                                                 *
042204                       ***************************************************
042204
042204                 INTG?   BSS  0              CIF NXT TOK IS INT
042204 170 270 313             LDBI R70,=PTR1-+    GET NXT TOK
042207 377
042210 310 032                 CMB  R70,=32
042212 236                     RTN
042213
042213                       ***************************************************
042213                       *                                                 *
042213                       *    CALCULATE ELEMENT SIZE FOR NUMERICAL DATA    *
042213                       *                                                 *
042213                       *      IN:                                        *
042213                       *        DIMFLG=TYPE OF NUMERICAL VAR             *
042213                       *                                                 *
042213                       *      OUT:                                       *
042213                       *        R66=CLEMENT SIZE                         *
042213                       *                                                 *
042213                       ***************************************************
042213
042213                 ELSZ    BSS  0
042213 166 251 010             LDM  R66,=10,0      ASSUME REAL
042216 000
042217 103 260 015             LDBD R3,=DIMFLG     GET FLG
042222 204
042223 317 060                 ANM  R3,=60         MASK OUT TYPE
042225 367 007                 JZR  ELSZR          JIF REAL
042227 231                     BCD
042230 206                     LRB  R3             SHIFT 4 BITS
042231 230                     BIN
042232 312 002                 ADB  R3,=2          3 FOR INT & 4 FOR SHORT
042234 066 242                 STB  R3,R66
042236 236             ELSZR   RTN
042237
042237                       ***************************************************
042237                       *                                                 *
042237                       *    INITIALIZE NUMERICAL VARIABLES               *
042237                       *                                                 *
042237                       *      IN:                                        *
042237                       *        R55-57=TOTAL SIZE OF STORAGE             *
042237                       *        PTR2=POINTS TO 1ST ELEMENT ADDR          *
042237                       *        DIMFLG=HEAD OF THE VARIABLE              *
042237                       *                                                 *
042237                       *      OUT:                                       *
042237                       *        VAR INITIALIZED                          *
042237                       *                                                 *
042237                       ***************************************************
```

```
>>>>      ALLOCATION ROUTINES                                    <<<<<<<<
    042237
984 042237                  INIV    BSS  0
985 042237 316 213 104              JSB  =ELSZ          CALC ELEMENT SIZE
986 042242 175 223                  CLM  R75            MAKE IT 3-BYTE IN R75
987 042244 166 075 243              STM  R66,R75
988 042247 160 223                  CLM  R60            SET NULL PATTERN
989 042251 167 250 377              LDB  R67,=377
990 042254 130 261 015              LDMD R30,=DIMFLG    GET TYPE OF NUM VAR
990 042257 204
991 042260 317 060 000              ANM  R30,=60,0      ISOLATE TYPE
992 042263 207                      LRM  R30            SHIFT 3 BITS
993 042264 207                      LRM  R30
994 042265 207                      LRM  R30
995 042266                  INILOP  BSS  0
996 042266 030 306 303              JSB  X30,INITBL     SET DRP
996 042271 104
997 042272 273 315 377              STMI R#,=PTR2-      INIT ONE ELEMENT
998 042275 155 075 305              SBM  R55,R75        SUB ONE ELEM SIZE
999 042300 366 364                  JNZ  INILOP         JIF MORE
000 042302 236                      RTN
001 042303
002 042303                  INITBL  BSS  0
003 042303 160              INIREL  DRP  60             8 BYTES FOR REAL
004 042304 236                      RTN
005 042305 165              INIINT  DRP  65             3 BYTES FOR INTEGER
006 042306 236                      RTN
007 042307 164              INISHT  DRP  64             4 BYTES FOR SHORT
008 042310 236                      RTN
009 042311
010 042311                  *****************************************************
010 042311                  *                                                   *
011 042311                  *    INITIALIZE STRING VARIABLES                    *
012 042311                  *                                                   *
013 042311                  *      IN:                                          *
014 042311                  *        R55-57=TOTAL SIZE OF STORAGE               *
015 042311                  *        R75-77=LENGTH OF EACH STRING ELEMENT       *
016 042311                  *        PTR2=POINTS TO 1ST ELEMENT                 *
016 042311                  *                                                   *
017 042311                  *****************************************************
017 042311
018 042311                  INISTR  BSS  0
019 042311 130 251 377              LDM  R30,=377,377   SET ACT LEN=-1
019 042314 377
020 042315                  INIST+  BSS  0
021 042315 130 273 317              STMI R30,=PTR2-+    INIT ONE STRING ELEMENT
021 042320 377
022 042321 316 332 104              JSB  =SKP2ST        PASS OVER VALUE PART
023 042324 155 075 305              SBM  R55,R75        DEC ONE ELEM SIZE
024 042327 366 364                  JNZ  INIST+         LOOP IF MORE
025 042331 236                      RTN
026 042332
027 042332                  SKP2ST  BSS  0
028 042332 175 006 345              PUMD R75,+R6        SAVE R75
```

```
>>>>       ALLOCATION ROUTINES                                            <<<<<<<<
029 042335 316 365 113          JSB   =SKPBY2       SKIP OVER
030 042340                      DRP   !75
031 042340                      ARP   !6
032 042340 343                  POMD  R75,-R6       RESTORE R75
033 042341 236                  RTN
034 042342
```

```
>>>>         INITIALIZE VARIABLES                                          <<<<<<<<
1036 042342
1037 042342                     ****************************************************
1037 042342                     *                                                  *
1038 042342                     *    INITIALIZE VARIABLES                          *
1038 042342                     *                                                  *
1038 042342                     ****************************************************
1039 042342
1040 042342                     INIVAR  BSS  0
1041 042342 230                         BIN
1042 042343 175 261 014                 LDMD R75,=BOVAR   START ADDR OF VAR
1042 042346 200
1043 042347 263 314 377                 STMD R75,=PTR2    SET PTR
1044 042352 261 017 200                 LDMD R75,=EOVAR   END ADDR OR VAR
1045 042355 263 150 215                 STMD R75,=XLPEN
1046 042360 360 062                     JMP  INIEND
1047 042362
1048 042362                     ****************************************************
1048 042362                     *                                                  *
1049 042362                     *    INIVLP                                        *
1050 042362                     *                                                  *
1051 042362                     *       IN:                                        *
1052 042362                     *         PTR2=1ST LOCATION OF VAR                 *
1053 042362                     *         XLPEN=LAST LOCATION OF VAR               *
1053 042362                     *                                                  *
1053 042362                     ****************************************************
1054 042362
1055 042362                     INIVLP  BSS  0
1056 042362 171 271 315                 LDMI R71,=PTR2-   GET HEADER
1056 042365 377
1057 042366 177 033 242                 STB  R77,R33      SAVE A COPY
1058 042371 262 015 204                 STBD R77,=DIMFLG  SAVE ONE MORE
1059 042374 317 004                     ANM  R77,=4       MASK OUT REMOTE BIT
1060 042376 366 044                     JNZ  INIEND       JIF REMOTE
1061 042400                     NOTRMT  BSS  0
1062 042400 133 220                     TSB  R33          TEST THE HEADER
1063 042402 363 015                     JEV  NOTFNC       JIF NOT FUNCTION
1064 042404 170 223                     CLM  R70          RESET RTN,..., ETC.
1065 042406 273 315 377                 STMI R70,=PTR2-
1065 042411 174 273 317                 STMI R74,=PTR2-+
1066 042414 377
1067 042415 171 271 315                 LDMI R71,=PTR2-
1067 042420 377
1068 042421                     NOTFNC  BSS  0
1069 042421 155 071 241                 LDM  R55,R71      GET TOTAL SIZE
1070 042424 133 200                     ELB  R33          NUM/STR BIT
1071 042426 200                         ELB  R33          ARAY/SIMPLE BIT
1072 042427 372 032                     JNC  NOTARY       JIF NOT ARRAY
1073 042431 174 271 315                 LDMI R74,=PTR2-   SKIP MAX ROW, COL
1073 042434 377
1074 042435 133 220                     TSB  R33
1075 042437 362 041                     JOD  INIST        JIF STRING
1076 042441
1077 042441                     ININUM  BSS  0
```

```
>>>>        INITIALIZE VARIABLES                                    <<<<<<<<
078 042441 316 237 104          JSB   =INIV          INIT NUM VARS
079 042444
080 042444                INIEND BSS   0
081 042444 175 261 314          LDMD  R75,=PTR2
081 042447 377
082 042450 321 150 215          CMMD  R75,=XLPEN
083 042453 766 305              JNZ   INIVLP
084 042455 102 222              CLB   R2
085 042457 262 015 204          STBD  R2,=DIMFLG
086 042462 236                  RTN
087 042463
088 042463                NOTARY BSS   0
089 042463 362 015              JOD   INIST          JIF SIMPLE STRING VAR
090 042465 316 213 104          JSB   =ELSZ          CALC ELEMENT SIZE
091 042470 155 271 316          LDMI  R55,=PTR2+     BACK UP 3 BYTES
091 042473 377
092 042474 223                  CLM   R55            RESET TOTAL SIZE REG
093 042475 166 055 243          STM   R66,R55        PUT TOTAL SIZE IN REG
094 042500 760 337              JMP   ININUM         GO INITIALIZE IT
095 042502
096 042502                INIST  BSS   0
097 042502 176 271 315          LDMI  R76,=PTR2-     GET MAX LENGTH
097 042505 377
098 042506 211                  ICM   R76            ELEM LEN = MAX LEN + 2
099 042507 211                  ICM   R76
100 042510 075 243              STM   R76,R75        MAKE IT 3 BYTES
101 042512 177 222              CLB   R77
102 042514 316 311 104          JSB   =INISTR        INIT STRING
103 042517 760 323              JMP   INIEND
104 042521
```

```
>>>>>>         CALCULATE ARRAY SIZE                                          <<<<<<<<<
042521
042521                              *******************************************************
042521                              *                                                     *
042521                              *    CALCULATE ARRAY SIZE                             *
042521                              *                                                     *
042521                              *      IN:                                            *
042521                              *        R74=MAX ROW                                  *
042521                              *        R76=MAX COL                                  *
042521                              *                                                     *
042521                              *      OUT:                                           *
042521                              *        R55=# OF BYTES OF THE ARRAY                  *
042521                              *                                                     *
042521                              *******************************************************
042521
042521                              *CALCSZ  BSS  0
042521                              *        LDM  R66,R74       COPY MAX ROW
042521                              *        JSB  =TOTELM       CALC TOT# ELEMENT
042521                              *        JSB  =ELSZ         CALC ELEMENT SIZE
042521                              *        JSB  =ARYSPC       CALC TOT SIZE
042521                              *        RTN
042521
042521                              TOTELM   BSS  0
042521 166 323 175                           ADMD R66,=OPTBAS
042524 200
042525 176 311 377                           CMM  R76,=377,377 VECTOR?
042530 377
042531 366 006                               JNZ  TOTEL1        JIF NOT
042533 156 223                               CLM  R56
042535 166 054 243                           STM  R66,R54
042540 236                                   RTN
042541                              TOTEL1   BSS  0
042541                                       DRP  !76
042541 323 175 200                           ADMD R76,=OPTBAS
042544 316 377 377                           JSB  =INTMUL
042547 236                                   RTN
042550
```

```
>>>>>>>        STORE ROUTINES                                                    <<<<<<<
 042550
 042550                       ************************************************************
 042550                       *                                                          *
 042550                       *     STORE HEADER & GET SPACE                             *
 042550                       *                                                          *
 042550                       *       IN:                                                *
 042550                       *         R44-46=REL POINTER TO NAME                       *
 042550                       *         R47=HEADER                                       *
 042550                       *         R70=SIZE OF DATA + (TOT,MAX ROW,MAX COL,ETC)     *
 042550                       *         PTR1=POINTS TO VAR NAME                          *
 042550                       *                                                          *
 042550                       *       OUT:                                               *
 042550                       *         PTR2=WHERE (TOT,MAX ROW,ETC. + DATA) TO BE STOR
 042550                       *                                                          *
 042550                       ************************************************************
 042550                       STOR1    BSS  0
 042550 100 260 264                    LDBD R0,=COMFLG     COMMON DECLARATION?
 042553 203
 042554 367 004                        JZR  STOR1X         JIF NO
 042556 104 251 052                    GTO  STOR1Z
 042561 106
 042562
 042562                       STOR1X   BSS  0
 042562 175 070 241                    LDM  R75,R70        GET SPACE NEEDED
 042565 313 004 000                    ADM  R75,=4,0,0     4 MORE BYTES FOR HEADER
 042570 000
 042571 243                            STM  R75,R70        SAVE TOTAL SPACE NEEDED
 042572 116 220                        TSB  R16            CALC MODE ?
 042574 362 060                        JOD  STOR1Y         JIF YES
 042576 316 250 111                    JSB  =TSTRM         ENOUGH ROOM?
 042601 372 077                        JNC  NORM1          JIF NO
 042603                                DRP  !75
 042603 261 017 200                    LDMD R75,=EOVAR     GET ADDR OF VAR
 042606 325 006 200                    SBMD R75,=FWCURR    MAKE IT REL ADDR
 042611 273 310 377                    STMI R75,=PTR1      STORE IN PLACE
 042614 144 012 345                    PUMD R44,+R12       SAVE HEADER
 042617                       STR1X2   BSS  0
 042617 145 261 017                    LDMD R45,=EOVAR     MOVE EOVAR-NXTMEM UP
 042622 200
 042623 316 345 106                    JSB  =MOVE
 042626 175 261 017                    LDMD R75,=EOVAR
 042631 200
 042632 263 314 377                    STMD R75,=PTR2      PT TO VAR STORE
 042635 102 250 002                    LDB  R2,=2          UPDATE 2 PTRS
 042640 114 251 017                    LDM  R14,=EOVAR     ADDR OF 1 PTR
 042643 200
 042644 316 373 106                    JSB  =UPDPTR        UPDATE PTRS
 042647 144 012 343                    POMD R44,-R12       RESTORE HEADER
 042652 273 315 377                    STMI R44,=PTR2-     STORE HEADER
 042655 236                            RTN
 042656
 042656                       STOR1Y   BSS  0
 042656 175 006 345                    PUMD R75,+R6        SAVE LENGTH
```

```
>>>>>>>        STORE ROUTINES                                                  <<<<<<<<
182 042661 223                      CLM  R75           GET NAME LENGTH
183 042662 270 313 377              LDBI R75,=PTR1-+
184 042665 313 013 000              ADM  R75,=13,0,0   ADD LENGTH OF HEADER
184 042670 000
185 042671 345                      PUMD R75,+R6       SAVE IT TOO
186 042672 070 303                  ADM  R75,R70       TOTAL SIZE FOR CALC
187 042674 243                      STM  R75,R70       SAVE IT
188 042675 316 250 111              JSB  =TSTRM2       ENOUGH ROOM ?
189 042700 373 002                  JCY  ROOMOK        JIF YES
190 042702 360 141          NORM1   JMP  NOROOM        BRIDGE JUMP
191 042704
192 042704                  ROOMOK  BSS  0
193 042704                          DRP  !75
194 042704 006 343                  POMD R75,-R6       GET LENGTH OF NAME BLK
195 042706 070 243                  STM  R75,R70       SAVE IT
196 042710 323 025 200              ADMD R75,=LAVAIL   GET SPACE FOR NAME BLK
197 042713 263 025 200              STMD R75,=LAVAIL   UPDATE POINTER
198 042716 263 314 377              STMD R75,=PTR2     SET PTR
199 042721 144 006 345              PUMD R44,+R6       SAVE HEADER
200 042724 147 250 210              LDB  R47,=210      FORM DUMMY HEADER
201 042727 144 273 315              STMI R44,=PTR2-    STORE DUMMY HEADER
201 042732 377
202 042733 175 070 241              LDM  R75,R70       LENGTH OF NAME BLK
203 042736 315 011 000              SBM  R75,=11,0,0   TOT LENGTH
203 042741 000
204 042742 172 273 315              STMI R72,=PTR2-    STORE IT
204 042745 377
205 042746 145 261 314              LDMD R45,=PTR2     SAVE POINTERS
205 042751 377
206 042752 006 345                  PUMD R45,+R6
207 042754 261 310 377              LDMD R45,=PTR1
208 042757 345                      PUMD R45,+R6
209 042760                  MOVNM   BSS  0             STORE NAME
210 042760 175 213                  DCM  R75
211 042762 367 011                  JZR  NMDONE
212 042764 170 270 311              LDBI R70,=PTR1-
212 042767 377
213 042770 272 315 377              STBI R70,=PTR2-
214 042773 360 363                  JMP  MOVNM
215 042775                  NMDONE  BSS  0
216 042775 175 006 343              POMD R75,-R6       RESTORE POINTERS
217 043000 263 310 377              STMD R75,=PTR1
218 043003 343                      POMD R75,-R6       RESTORE NAME ADDR
219 043004 325 030 200              SBMD R75,=CALVRB   MAKE IT REL TO CALVRB
220 043007 144 343                  POMD R44,-R6       RESTORE HEADER
221 043011 175 044 243              STM  R75,R44       CHANGE NAME POINTER
222 043014 006 343                  POMD R75,-R6       RESTORE LENGTH
223 043016 070 243                  STM  R75,R70
224 043020 261 025 200              LDMD R75,=LAVAIL   UPDATE POINTER
225 043023 303                      ADM  R75,R70
226 043024 263 025 200              STMD R75,=LAVAIL
227 043027 263 314 377              STMD R75,=PTR2     PT TO CALC VAR ADDR
    043032 325 006 200              SBMD R75,=FWCURR
```

```
ARMA 03/14/81 ==================================================================
ITEM    LOC    OBJECT CODE  SRC=CCALLO OBJ=GEXALL     9/11/1981 11:30 AM   PG 29
================================================================================
>>>>>>        STORE ROUTINES                                             <<<<<<<<
229 043035 273 310 377              STMI R75,=PTR1     STORE REL ADDR
230 043040 144 273 315              STMI R44,=PTR2-    STORE HEADER
230 043043 377
231 043044 236                      RTN
232 043045
233 043045               NOROOM     BSS  0
234 043045 316 377 377              JSB  =ERROR        MEM OVFL ERROR
235 043050 023                      OCT  19D
236 043051 360 165                  JMP  COMER+        CLEAN R6 STACK
237 043053
238 043053               STOR1Z     BSS  0
239 043053 144 064 243              STM  R44,R64       SAVE HEADER & NAME PTR
240 043056 147 312 004              ADB  R47,=4        SET REMOTE BIT
241 043061 175 261 011              LDMD R75,=NXTCOM   GET ADDR OF COM VAR
241 043064 200
242 043065 041 243                  STM  R75,R41       SAVE IT
243 043067 260 263 203              LDBD R75,=DCLCOM   ORIGIN DECLARATION?
244 043072 367 151                  JZR  STOR1W        JIF NOS
245 043074 070 241                  LDM  R75,R70       GET SIZE
246 043076 313 013 000              ADM  R75,=13,0,0   FOR HEADERS
246 043101 000
247 043102 243                      STM  R75,R70       SAVE IT
248 043103 316 250 111              JSB  =TSTRM        ENOUGH ROOM?
249 043106 372 335                  JNC  NOROOM        JIF NO
250 043110                          DRP  !75
251 043110                          ARP  !70
252 043110 241                      LDM  R75,R70       GET SIZE
253 043111 315 007 000              SBM  R75,=7,0,0    DIST TO BE MOVED
253 043114 000
254 043115 012 345                  PUMD R75,+R12      SAVE IT
255 043117 325 164 210              SBMD R75,=SAVLAV   UPDATE SAVED PTR1
256 043122 215                      TCM  R75
257 043123 263 164 210              STMD R75,=SAVLAV
258 043126 251 007 000              LDM  R75,=7,0,0    # BYTES FOR LOCAL COPY
258 043131 000
259 043132 070 243                  STM  R75,R70       SAVE IT
260 043134 261 017 200              LDMD R75,=EOVAR    GET REL ADDR
261 043137 325 006 200              SBMD R75,=FWCURR
262 043142 273 310 377              STMI R75,=PTR1     STORE IT
263 043145 141 012 345              PUMD R41,+R12      SAVE HEADER
264 043150 316 217 105              JSB  =STR1X2       MOVE SPACE FOR LOCAL COPY
265 043153                          ARP  !12
266 043153 145 343                  POMD R45,-R12      POP 3 BYTES MORE
267 043155 273 315 377              STMI R45,=PTR2-    STORE PTR TO REMOTE
268 043160 343                      POMD R45,-R12      GET SIZE OF COMMON VAR
269 043161 070 243                  STM  R45,R70       SAVE IT
270 043163 261 003 200              LDMD R45,=FWPRGM   MOVE FWPRGM-NXTMEM UP
271 043166 316 345 106              JSB  =MOVE
272 043171 102 250 006              LDB  R2,=6         UPDATE 6 PTRS
273 043174 114 251 003              LDM  R14,=FWPRGM   ADDR OF 1ST PTR
273 043177 200
274 043200 316 373 106              JSB  =UPDPTR
275 043203                          DRP  !75
```

```
>>>>          STORE ROUTINES                                                  <<<<<<<<
043203                           ARP  !70
043203 261 115 202               LDMD R75,=PCR       ADJUST PCR FOR ERR RPRT
043206 305                       SBM  R75,R70
043207 263 115 202               STMD R75,=PCR
043212 261 257 203               LDMD R75,=OLDPCR    ADJUST OLDPCR FOR TRACE
043215 305                       SBM  R75,R70
043216 263 257 203               STMD R75,=OLDPCR
043221 261 310 377               LDMD R75,=PTR1      PTR TO COM VAR
043224 263 314 377               STMD R75,=PTR2
043227 164 273 315               STMI R64,=PTR2-     STORE HEADER + NAME PTR
043232 377
043233 236                       RTN
043234
043234                  COMERR   BSS  0
043234 316 377 377               JSB  =ERROR         COM ITEM MISMATCH
043237 040                       OCT  32D
043240                  COMER+   BSS  0
043240 106 261 150               LDMD R6,=XLPEN      CLEAN UP STACK
043243 215
043244 236                       RTN
043245
043245                  STOR1W   BSS  0
043245 175 261 011               LDMD R75,=NXTCOM
043250 200
043251 263 314 377               STMD R75,=PTR2
043254 127 270 315               LDBI R27,=PTR2-     GET FLAGS
043257 377
043260 047 226                   XRB  R27,R47        SAME TYPE?
043262 317 360                   ANM  R27,=360
043264 366 346                   JNZ  COMERR         JIF NOT
043266 175 271 315               LDMI R75,=PTR2-     SKIP 3 DUMMY BYTES
043271 377
043272 147 310 100               CMB  R47,=100       SIMPLE VAR
043275 372 010                   JNC  ADDCOM         JIF YES
043277 175 271 317               LDMI R75,=PTR2-+    GET TOT
043302 377
043303 055 301                   CMM  R75,R55        SAME SIZE?
043305 366 325                   JNZ  COMERR         JIF NOT
043307                  ADDCOM   BSS  0
043307 175 044 241               LDM  R75,R44        COPY NAME PTR
043312 273 314 377               STMI R75,=PTR2      INTO COM VAR
043315 261 314 377               LDMD R75,=PTR2      UPDATE NXTCOM
043320 070 305                   SBM  R75,R70
043322 263 011 200               STMD R75,=NXTCOM
043325 251 003 000               LDM  R75,=3,0,0     NEED 3 BYTES
043330 000
043331 243                       STM  R75,R70        SAVE IT.
043332 316 162 105               JSB  =STOR1X        ALLOCATE FOR IT
043335 175 041 241               LDM  R75,R41        GET PTR TO RMT VAR
043340 273 315 377               STMI R75,=PTR2-     STORE IN PLACE
043343 760 273                   JMP  COMER+         CLEAN R6 STACK
043345
043345                  ***************************************************************
```

```
>>>>          STORE ROUTINES                                                  <<<<<<<<<
321 043345                  *
322 043345                  *     R45=LOWEST ADDR OF BLOCK                              *
323 043345                  *     R70=DISTANCE TO MOVE                                  *
324 043345                  *     NXTMEM IS ALWAYS THE LAST BYTE TO MOVE                *
324 043345                  *                                                           *
324 043345                  *************************************************************
325 043345
326 043345                  MOVE    BSS  0
327 043345 145 325 022              SBMD R45,=NXTMEM  BYTE COUNT FOR MOVE
327 043350 200
328 043351 367 017                 JZR  MDONE        JIF NOTHING TO MOVE
329 043353 175 261 022              LDMD R75,=NXTMEM  SOURCE
329 043356 200
330 043357 263 310 377              STMD R75,=PTR1
331 043362 070 305                  SBM  R75,R70      DESTINATION
332 043364 263 314 377              STMD R75,=PTR2
333 043367 316 377 377              JSB  =EMOVDN
334 043372 236              MDONE   RTN
[illegible] 043373
[illegible] 043373                  *************************************************************
336 043373                  *                                                           *
337 043373                  *    R2 =#OF PTRS TO BE UPDATED                             *
338 043373                  *    R14=ADDR OF 1ST PTR                                    *
339 043373                  *    R70=AMOUNT TO ADJUST                                   *
339 043373                  *                                                           *
339 043373                  *************************************************************
340 043373
341 043373                  UPDPTR  BSS  0
342 043373 175 014 245              LDMD R75,R14      GET A PTR
343 043376 070 305                  SBM  R75,R70      UPDATE
344 043400 014 345                  PUMD R75,+R14     PUSH BACK
345 043402 102 212                  DCB  R2           DEC COUNT
346 043404 366 365                  JNZ  UPDPTR       LOOP IF MORE
347 043406 175 261 154              LDMD R75,=SAVPT2  UPDATE SAVPT2
347 043411 210
348 043412 070 305                  SBM  R75,R70
349 043414 263 154 210              STMD R75,=SAVPT2
[illegible] 043417 236                  RTN
[illegible] 043420
352 043420                  *************************************************************
352 043420                  *                                                           *
353 043420                  *   SUBROUTINE LINEAL                               *       *
354 043420                  *                                                   *       *
355 043420                  *   THIS ROUTINE ALLOCATES LINE # REFERENCES*       *
356 043420                  *   BY REPLACING THE LINE # REFERENCE WITH  *       *
357 043420                  *   ITS ADDRESS RELATIVE TO FWCURR          *       *
358 043420                  *                                           *       *
359 043420                  *    JMP TOKEN/LINE#/LINE#                  *       *
360 043420                  *                                           *       *
361 043420                  *********************************************       *
361 043420                  *                                                   *
361 043420                  *************************************************************
362 043420                  LINEAL  BSS  0
```

```
043420 316 377 377            JSB  =LABTK?       ONE OF THE LABEL TOKEN?
043423 367 112                JZR  LABLAL        JIF YES
043425 175 271 311            LDMI R75,=PTR1-    GET LINE #
043430 377
043431 364 026                JNG  LINEA1        JIF ALREADY ALLOCATED
043433 316 062 107            JSB  =FNDL2        FIND LINE #
043436 371 004                JEZ  FNDLN         JIF FOUND
043440 316 377 377            JSB  =ERROR+       ELSE ERROR
043443 071                    OCT  57D           LINE# UNDEFINED
043444
043444             FNDLN      BSS  0
043444 175 271 316            LDMI R75,=PTR2+    BACKUP 3 BYTES
043447 377
043450 261 314 377            LDMD R75,=PTR2     GET ABS ADDR
043453 325 006 200            SBMD R75,=FWCURR   GET REL ADDR
043456 273 310 377            STMI R75,=PTR1     STORE IN PLACE
043461 236         LINEA1     RTN
043462
043462             ********************************************************
043462             *                                                      *
043462             *    FIND A LINE # BY USING POINTER2                   *
043462             *                                                      *
043462             *       IN:                                            *
043462             *         R75=LINE#                                    *
043462             *                                                      *
043462             *       OUT:                                           *
043462             *         E=0 IF FOUND                                 *
043462             *             & PTR2 POINTS TO STSIZE                  *
043462             *         E#1 IF NOT FOUND                             *
043462             *                                                      *
043462             ********************************************************
043462
043462             FNDL2      BSS  0
043462 165 006 345            PUMD R65,+R6       SAVE REGS
043465 316 377 377            JSB  =BOP          POINT TO 1ST LINE
043470                        DRP  !65
043470 263 314 377            STMD R65,=PTR2     SET UP POINTER
043473             FND2LP     BSS  0
043473 235                    CLE                RESET FLAG
043474 165 271 315            LDMI R65,=PTR2-    FETCH LINE#
043477 377
043500 231                    BCD
043501 075 301                CMM  R65,R75       FOUND?
043503 230                    BIN
043504 367 010                JZR  FND2RT        JIF YES
043506 373 005                JCY  FND2EX        JIF PASSED
043510 316 122 107            JSB  =SKP2LN       SKIP THIS LINE
043513 360 356                JMP  FND2LP        LOOP BACK
043515 233         FND2EX     DCE                SIGNAL NOT FOUND
043516             FND2RT     BSS  0
043516 165 006 343            POMD R65,-R6       RESTORE WORKING REGS
043521 236                    RTN
043522
```

```
ARMA 03/14/81=========================================================================
ITEM     LOC    OBJECT CODE  SRC=CCALLO OBJ=GEXALL    9/11/1981 11:30 AM   PG 33
======================================================================================
>>>>>          STORE ROUTINES                                               <<<<<<<<
41   043522                   ************************************************************
410  043522                   *                                                          *
411  043522                   *     SKIP A STRING                                        *
412  043522                   *                                                          *
413  043522                   *       IN:                                                *
414  043522                   *         PTR2=POINTS TO BYTE COUNT                        *
415  043522                   *                                                          *
416  043522                   *       OUT:                                               *
417  043522                   *         PTR2=POINTS TO AFTER THE STRING                  *
417  043522                   *                                                          *
417  043522                   ************************************************************
418  043522
419  043522                   SKP2NA  BSS  0              SKIP A NAME
420  043522                   SKP2LN  BSS  0              SKIP A LINE
421  043522 175 006 345               PUMD R75,+R6        SAVE REGS
422  043525 223                       CLM  R75
423  043526 270 315 377               LDBI R75,=PTR2-     GET BYTE COUNT
424  043531 230                       BIN
  1  043532 316 365 113               JSB  =SKPBY2        SKIP
42   043535                           DRP  !75
427  043535                           ARP  !6
428  043535 343                       POMD R75,-R6        RESTORE REGS
429  043536 236                       RTN
430  043537
431  043537
432  043537                   ************************************************************
432  043537                   *
433  043537                   *     ALLOCATION FOR LABELS
433  043537                   *                                                          *
433  043537                   ************************************************************
434  043537
435  043537                   LABLAL  BSS  0
436  043537 175 271 311               LDMI R75,=PTR1-     ALREADY ALLOCATED?
436  043542 377
437  043543 364 033                   JNG  LRTN2          JIF YES
438  043545 261 310 377               LDMD R75,=PTR1      SAVE POINTER
439  043550 006 345                   PUMD R75,+R6
  0  043552 316 204 107               JSB  =FNDLAB        SEARCH FOR LABEL
44   043555 165 006 343               POMD R65,-R6        RESTORE POINTER
442  043560 263 314 377               STMD R65,=PTR2
443  043563 117 310 300               CMB  R17,=300       ERROR?
444  043566 373 007                   JCY  LRTN           JIF YES
445  043570 175 325 006               SBMD R75,=FWCURR    GET REL ADDR
445  043573 200
446  043574 273 314 377               STMI R75,=PTR2      STORE IN PLACE
447  043577 236                LRTN   RTN
448  043600
449  043600                   LRTN2   BSS  0
450  043600 316 377 377               JSB  =SKPNM+        SKIP LABEL
451  043603 236                       RTN
452  043604
453  043604
454  043604                   ************************************************************
```

```
043604                   *                                                             *
043604                   *     FIND A LABEL                                            *
043604                   *                                                             *
043604                   *       IN:                                                   *
043604                   *         PTR1=POINTS TO TARGET LABEL                         *
043604                   *                                                             *
043604                   *       OUT:                                                  *
043604                   *         R75=ABS ADDR OF LINE CONTAINS THE LABEL             *
043604                   *                                                             *
043604                   ***************************************************************
043604
043604                   FNDLAB  BSS  0
043604 316 377 377               JSB  =BOP           POINT TO 1ST LINE
043607                   FLLOOP  BSS  0
043607                           DRP  !65
043607 263 314 377               STMD R65,=PTR2      SETUP POINTER
043612 006 345                   PUMD R65,+R6
043614 271 315 377               LDMI R65,=PTR2-     SKIP LINE#
043617 311 231 231               CMM  R65,=231,231,12   LINE A9999?
043622 012
043623 367 056                   JZR  FLERR          JIF YES
043625 270 315 377               LDBI R65,=PTR2-     GET STSIZE
043630 344                       PUBD R65,+R6        SAVE IT
043631                   FLOOP   BSS  0
043631 102 270 315               LDBI R2,=PTR2-      GET NEXT TOKEN
043634 377
043635 310 367                   CMB  R2,=367        INDENT?
043637 366 006                   JNZ  LABEL?         JIF NOT
043641 270 315 377               LDBI R2,=PTR2-      SKIP BYTE COUNT
043644 270 315 377               LDBI R2,=PTR2-      GET NEXT TOKEN
043647                   LABEL?  BSS  0
043647                           DRP  !2
043647 310 257                   CMB  R2,=257        LABEL TOKEN?
043651 366 005                   JNZ  NEXTL          JIF NOT
043653 316 311 107               JSB  =MATCH?        GO COMPARE
043656 370 016                   JEN  FOUND          JIF MATCH
043660 175 223           NEXTL   CLM  R75
043662 006 342                   POBD R75,-R6        RESTORE STSIZE
043664 165 343                   POMD R65,-R6        RESTORE LINE ADDR
043666 075 305                   SBM  R65,R75        SKIP CURR LINE
043670 315 004 000               SBM  R65,=4,0,0     BACKUP 4 MORE
043673 000
043674 360 311                   JMP  FLLOOP
043676
043676                   FOUND   BSS  0
043676 175 006 342               POBD R75,-R6        THROW AWAY STSIZE
043701 343                       POMD R75,-R6        RETURN ABS ADDR OF LABEL LINE
043702 236                       RTN
043703
043703                   FLERR   BSS  0
043703                           DRP  !65
043703 006 343                   POMD R65,-R6        THROW AWAY POINTER
043705 316 377 377               JSB  =ERROR+        LABEL NOT FOUND ERROR
```

```
>>>>>>      STORE ROUTINES                                                  <<<<<<<
502 043710 071                   OCT   570
503 043711
504 043711              **************************************************************
504 043711              *                                                            *
505 043711              *    COMPARE TWO STRINGS                                     *
506 043711              *                                                            *
507 043711              *      IN:                                                   *
508 043711              *        PTR1=POINTS TO TARGET STRING                        *
509 043711              *        PTR2=POINTS TO INPUT STRING                         *
510 043711              *                                                            *
511 043711              *      OUT:                                                  *
512 043711              *        IF STRING MATCH THEN                                *
513 043711              *            E=1 ,                                           *
514 043711              *            PTR1=POINTS TO PASS NAME                        *
515 043711              *                                                            *
516 043711              *        IF THEY DON'T MATCH THEN                            *
517 043711              *            E=0 ,                                           *
518 043711              *            PTR1=STILL POINTS TO NAME                       *
5   043711              *                                                            *
51  043711              **************************************************************
519 043711
520 043711              MATCH?  BSS   0
521 043711 165 261 310          LDMD R65,=PTR1     SAVE POINTER
521 043714 377
522 043715 006 345              PUMD R65,+R6
523 043717 235                  CLE                RESET FLAG
524 043720 160 270 311          LDBI R60,=PTR1-    GET BYT CNT OF TARGET LABEL
524 043723 377
525 043724 161 270 315          LDBI R61,=PTR2-    GET BYT CNT OF CURR LABEL
525 043727 377
526 043730 060 300              CMB  R61,R60       LENGTH MATCH?
527 043732 366 025              JNZ  MCHRTN        JIF NO
528 043734              MINLOP  BSS  0             INNER LOOP
529 043734 161 270 311          LDBI R61,=PTR1-    FETCH 1 CHAR
529 043737 377
530 043740 162 270 315          LDBI R62,=PTR2-
530 043743 377
5   043744 061 300              CMB  R62,R61       SAME CHAR?
53  043746 366 011              JNZ  MCHRTN        JIF NO
533 043750 160 212              DCB  R60           DEC BYTE COUNT
534 043752 366 360              JNZ  MINLOP        LOOP IF MORE
535 043754 234                  ICE                FLAG FOUND
536 043755 165 006 343          POMD R65,-R6       THROW AWAY OLD PTR1
537 043760 236                  RTN
538 043761 165 006 343  MCHRTN  POMD R65,-R6       RESTORE POINTER
539 043764 263 310 377          STMD R65,=PTR1
540 043767 236                  RTN
541 043770
542 043770
543 043770
```

```
>>>>>>>        FN ALLOCATION                                                  <<<<<<<<
1545 043770                 **************************************************
1546 043770                 *                                                *
1547 043770                 *  FN ALLOCATION                                 *
1548 043770                 *                                                *
1549 043770                 *  THIS ROUTINE ALLOCATES USER DEFINED FUNCTIONS*
1550 043770                 *                                                *
1551 043770                 *   DEF TOKEN/NAME/NAME/JMP ADDR/JMP ADDR/       *
1552 043770                 *           2*PARAMS+STR/---PARAMS---            *
1553 043770                 *                                                *
1554 043770                 **************************************************
1555 043770                 DEFFN   BSS  0
1556 043770 165 271 311            LDMI R65,=PTR1-   ALREADY ALLOCATED?
1556 043773 377
1557 043774 367 004                JZR  DEFFN1       JIF NOT
1558 043776 316 377 377            JSB  =SKPDEF      SKIP FUNC DEFINITION
1559 044001 236                    RTN
1560 044002                 DEFFN1  BSS  0
1561 044002                         DRP  !65
1562 044002 261 172 210            LDMD R65,=DFPAR1  WITHIN FUN DEF?
1563 044005 366 172                JNZ DEFDEF        JIF YES
1564 044007 210                    ICB  R65          SIGNIFY IN DEF FN
1565 044010 263 172 210            STMD R65,=DFPAR1
1566 044013 261 115 202            LDMD R65,=PCR     SAVE PCR OF DEF FN
1567 044016 263 160 215            STMD R65,=YLPEN   FOR PAR NAME SEARCH
1568 044021 261 310 377            LDMD R65,=PTR1    SAVE POINTER
1569 044024 316 377 377            JSB  =SKPNM+      SKIP NAME
1570 044027 175 271 311            LDMI R75,=PTR1-   SKIP JUMP ADDR
1570 044032 377
1571 044033 123 270 311            LDBI R23,=PTR1-   GET PAR CNT/STR FLG
1571 044036 377
1572 044037 317 001                ANM R23,=1         1 IF STR
1573 044041 204                    LLB R23            2 IF STR
1574 044042 312 201                ADB R23,=201       201 OR 203
1575 044044 165 263 310            STMD R65,=PTR1    RESTORE POINTER
1575 044047 377
1576 044050 316 205 110            JSB =CRDF?        ALLOC DEF IF NOT ALREADY
1577 044053 117 311 300            CMM  R17,=300     ANY ERROR
1578 044056 373 021                JCY  DEFFNE       JIF YES
1579 044060 371 024                JEZ  DEFFN2       JIF NOT FOUND
1580 044062 145 315 004            SBM  R45,=4,0,0   PT TO FUNC DEF ADR
1580 044065 000 000
1581 044067 263 314 377            STMD R45,=PTR2    SET POINTER
1582 044072 271 317 377            LDMI R45,=PTR2-+  GET FUNC DEF ADR
1583 044075 366 076                JNZ  DUPDEF       JIF ALREADY DEFINED
1584 044077 360 011                JMP  DEFFN3
1585 044101                 DEFFNE  BSS  0
1586 044101 316 377 377            JSB  =SKPNM+      SKIP FUNC NAME
1587 044104 360 035                JMP  DEFFN4
1588 044106                 DEFFN2  BSS  0
1589 044106 175 263 314            STMD R75,=PTR2    PT TO FUNC DEF ADR FIELD
1589 044111 377
1590 044112                 DEFFN3  BSS  0
1591 044112 175 271 310            LDMI R75,=PTR1    GET CUR FN ADDR
```

```
>>>>>>        FN ALLOCATION                                          <<<<<<<
591  044115 377
592  044116 263 164 200          STMD R75,=CURFUN   SAVE FOR FN END
593  044121 316 377 377          JSB  =SKPNM+       SKIP FUNC NAME
594  044124                      OPP  !75           SAVE PTR OF JMP ADR
595  044124 263 167 210          STMD R75,=JMPADR
596  044127 271 311 377          LDMI R75,=PTR1-    SKIP JUMP ADR
597  044132 261 310 377          LDMD R75,=PTR1     GET ADDR
598  044135 325 006 200          SBMD R75,=FWCURR   MAKE IT REL
599  044140 273 315 377          STMI R75,=PTR2-    SAVE IT
600  044143                DEFFN4 BSS  0
601  044143 170 270 311          LDBI R70,=PTR1-    GET PAR CNT
601  044146 377
602  044147 206                  LRB  R70           DIV BY 2 TO GET CNT
603  044150 316 377 377          JSB  =SKPANM       SKIP PAR NAMES
604  044153 175 261 310          LDMD R75,=PTR1     GET POINTER
604  044156 377
605  044157 263 172 210          STMD R75,=DFPAR1   FWA OF PAR STORAGE
606  044162 316 377 377          JSB  =SKPRS        SKIP PARS
607  044165 175 261 310          LDMD R75,=PTR1     GET POINTER
608  044170 377
608  044171 263 175 210          STMD R75,=DFPAR2   LWA OF PAR STORAGE
609  044174 236                  RTN
610  044175
611  044175 316 377 377 DUPDEF   JSB =ERROR+
612  044200 045                  OCT 37D             DUPLICATE DEF
613  044201 316 377 377 DEFDEF   JSB =ERROR+
614  044204 046                  OCT 38D                DEF WITHIN DEF
615  044205
616  044205
617  044205                *****************************************************
617  044205                *                                                   *
618  044205                *     ALLOCATE FOR FUNCTIONS                        *
619  044205                *                                                   *
620  044205                *       IN:                                         *
621  044205                *         PTR1=POINTS TO FUNC NAME                  *
622  044205                *         R23=FUNC TYPE                             *
623  044205                *                                                   *
624  044205                *       OUT:                                        *
625  044205                *         IF FOUND THEN                             *
626  044205                *             E#0                                   *
627  044205                *             PTR1=POINTS TO NAME                   *
628  044205                *             R45=POINTS TO FUNC STORAGE            *
629  044205                *                                                   *
630  044205                *         IF NOT FOUND THEN                         *
631  044205                *             ALLOCATE STORAGE FOR FUNC             *
632  044205                *             E=0                                   *
633  044205                *             PTR1=POINTS TO NAME                   *
634  044205                *             R75=ADDR OF FUNC ADDR FIELD OF FUNC STORE *
634  044205                *                                                   *
634  044205                *****************************************************
635  044205
636  044205                CRDF?  BSS  0
637  044205 316 273 111          JSB  =FRMNAM       FORM HEADER
```

```
>>>>        FN ALLOCATION                                        <<<<<<<<
  3 044210 145 261 310          LDMD R45,=PTR1     SAVE POINTER
538 044213 377
539 044214 263 164 210          STMD R45,=SAVLAV
540 044217 261 017 200          LDMD R45,=EOVAR    GET LWA OF VAR
541 044222 040 243              STM  R45,R40       SETUP
542 044224 261 014 200          LDMD R45,=BOVAR    GET FWA OF VAR
543 044227 163 222              CLB  R63           FLG NOT FN PARS
544 044231 316 304 101          JSB  =VSRCH        SEARCH VAR TABLE
545 044234                ******* LDMD R65,=SAVLAV RESTORE NAME ADDR
546 044234                ******* STMD R65,=PTR1
547 044234 316 274 101          JSB  =RHMOD1       RESTORE NAME ADDR
548 044237 371 012              JEZ  CRDF0         JIF NOT FOUND
549 044241 175 045 241          LDM  R75,R45       COPY FUNC ADDR
550 044244 325 006 200          SBMD R75,=FWCURR   MAKE IT RELATIVE
551 044247 273 310 377          STMI R75,=PTR1     STORE IN PLACE
552 044252 236                  RTN
553 044253                CRDF0 BSS  0
554 044253                ****** ORP  !65
555 044253                ****** SBMD R65,=FWCURR  GET REL ADDR
556 044253                ****** STM  R65,R44      SETUP HEADER
557 044253
558 044253                      ORP  !75
559 044253 325 006 200          SBMD R75,=FWCURR   MAKE IT REL ADR
560 044256 044 243              STM  R75,R44       SAVE IT
561 044260 147 037 240          LDB  R47,R37       GET FLAGS
562 044263 106 263 150          STMD R6,=XLPEN     FOR ERROR HANDLING
  2 044266 215
563 044267 123 310 203          CMB  R23,=203      STRING FUNC?
564 044272 367 052              JZR  STRFUN        JIF YES
565 044274 155 251 010          LDM  R55,=10,0,0   ;8 BYTES FOR REAL
565 044277 000 000
566 044301 175 055 241          LDM  R75,R55       COPY IT
567 044304 313 017 000          ADM  R75,=17,0,0   MORE BYTES FOR HEADER
567 044307 000
568 044310 316 323 110          JSB  =DFCOMN       DO COMN PART
569 044313 167 212              DCB  R67           INIT FUNC VALUE
570 044315 160 273 315          STMI R60,=PTR2-
570 044320 377
571 044321 360 062              JMP  GOSTOS
572 044323
573 044323                DFCOMN BSS  0
574 044323                      ORP  !75           ORP MUST BE 75
575 044323 070 243              STM  R75,R70
576 044325 316 150 105          JSB  =STOR1        ALLOCATE FOR FUNC
577 044330 175 261 314          LDMD R75,=PTR2     SAVE ADR OF FUN ADR FIELD
577 044333 377
578 044334 160 223              CLM  R60
579 044336 273 315 377          STMI R60,=PTR2-    INIT HEADER INFO
580 044341 161 273 315          STMI R61,=PTR2-
580 044344 377
581 044345 236                  RTN
582 044346
  3 044346                STRFUN BSS  0
```

```
>>>>        FN ALLOCATION                                          <<<<<<<
684 044346 155 251 024          LDM  R55,=20D,0,0 DEFAULT STRING ELEM LEN
684 044351 000 000
685 044353 175 055 241          LDM  R75,R55      COPY IT
686 044356 313 024 000          ADM  R75,=20D,0,0 MORE BYTES FOR HEADER INFO
686 044361 000
687 044362 316 323 110          JSB  =OFCOMN      DO COMN PART
688 044365 155 273 315          STMI R55,=PTR2-   TOT LEN
688 044370 377
689 044371 156 251 022          LDM  R56,=18D,0
689 044374 000
690 044375 273 315 377          STMI R56,=PTR2-   MAX LEN
691 044400 166 213              DCM  R66          NULL STR LEN = 377,377
692 044402 273 315 377          STMI R66,=PTR2-   INIT ACT LEN
693 044405              GOSTOS  BSS  0
694 044405 235                  CLE               SIGNAL NOT FOUND
695 044406              CRDF1?  BSS  0
696 044406 236,                 RTN
697 044407                      ********************************************
698 044407                      *                                          *
699 044407                      *  ALLOCATE FN CALL                        *
700 044407                      *                                          *
701 044407                      *  THIS ROUTINE ALLOCATES USER FUNCTION CALLS *
702 044407                      *                                          *
703 044407                      *  FN CALL TOKEN/NAME/NAME/# PARAMS/--PARAM TYPES *
704 044407                      *                                  1 BYTE EACH *
705 044407                      *                                          *
706 044407                      ********************************************
707 044407              FUNCAL  BSS  0
708 044407 165 271 311          LDMI R65,=PTR1-   SKIP ADDR
708 044412 377
709 044413 364 020              JNG  FUNCA1       JIF ALREADY ALLOCATED
710 044415 116 310 001          CMB  R16,=1       CALC MODE?
711 044420 367 022              JZR FNUNDF        JIF YES
712 044422 123 314 025          SBB R23,=25        MAKE IT RIGHT
713 044425 362 001              JOD NTSTRF
714 044427 210                  ICB R23
715 044430 312 200      NTSTRF  ADB R#,=200        FUNC FLAG
716 044432 316 205 110          JSB =CRDF?        ALLOC FUNC
717 044435 316 377 377 FUNCA1   JSB  =3KFNM+       SKIP FUNC NAME
718 044440 316 377 377          JSB  =3KFNM+       SKIP PAR TYPES
719 044443 236                  RTN
720 044444
721 044444              FNUNDF  BSS 0
722 044444 316 377 377 ILLCOM   JSB =ERROR+
723 044447 130                  OCT 98D            ILLEGAL STATEMENT
724 044450
725 044450 116 220      COMM    TSB R16
726 044452 362 370              JOD ILLCOM         COM IN CALC MODE
727 044454 175 261 172          LDMD R75,=DFPAR1   DEF?
727 044457 210
728 044460 366 362              JNZ ILLCOM         COM IN DEF ILLEGAL
729 044462 261 000 200          LDMD R75,=FWUSER   IS IT ORIG DECL ?
730 044465 321 003 200          CMMD R75,=FWPRGM
```

```
1     044470 366 005                    JNZ   COMRTN          JIF NOT
1732 044472 250 001                    LDB   R75,=1          FLAG YES
1733 044474 262 263 203                STBD  R75,=DCLCOM
1734 044477                    COMRTN  BSS   0
1735 044477 122 250 001                LDB   R22,=1          SET COMMON FLAG
1736 044502 262 264 203                STBD  R22,=COMFLG
1737 044505 222                        CLB   R22             DEFAULT TO REAL
1738 044506 360 014                    JMP   DIMCOM          SET DIM FLAG
1739 044510
1740 044510                    REAL    BSS 0
1741 044510 176 222            DIMAL   CLB R76               DIMFLG = 0
1742 044512 360 010                    JMP DIMCOM
1743 044514 176 250 020 INT             LDB R76,=20           DIMFLG = 20
1744 044517 360 003                    JMP DIMCOM
1745 044521 176 250 040 SHORT           LDB R76,=40           DIMFLG = 40
1746 044524 262 015 204 DIMCOM          STBD R#,=DIMFLG
1747 044527 122 222                    CLB R22               DIM FLAG
1748 044531 262 160 210                STBD R22,=DIMDIM
   9 044534 236                        RTN
17   044535
1751 044535                    CUROPT  BSS   0
1752 044535 316 152 111                JSB   =CUROP1         GET CUR OPTION BASE
1753 044540                    SETOPT  BSS   0
1754 044540 176 261 175                LDMD  R76,=OPTBAS     OPTION BASE SET?
1754 044543 200
1755 044544 365 071                    JPS   OPTEX           JIF YES
   6 044546 223                        CLM   R76             DEFAULT OPT BAS 0
1 57 044547 211                        ICM   R76
1758 044550 360 055                    JMP   OPT1
1759 044552
1760 044552                    *
1761 044552                    *
1762 044552                    *  SUBROUTINE CUROPT
1763 044552                    *
1764 044552                    *  THIS ROUTINE WILL SET THE OPTION
1765 044552                    *  BASE TO THAT OF THE PROGRAM POINTED TO
1766 044552                    *  BY FWCURR.
   7 044552                    CUROP1  BSS   0
1    044552 230                        BIN
1769 044553 176 223                    CLM   R76
1770 044555 210                        ICB   R76             ASSUME OPTION BASE 0
1771 044556 116 220                    TSB   R16             CALC MODE ?
1772 044560 362 051                    JOD   OPT2            JIF YES
1773 044562                    CUROPS  BSS   0
1774 044562 176 261 214                LDMD  R76,=PGMOPT     GET PRGM OPTBAS
1774 044565 210
1775 044566 360 043                    JMP   OPT2
1776 044570
1777 044570
1778 044570                    ******************************************************
1778 044570                    *                                                    *
1779 044570                    *                                                    *
   0 044570                    *    SET OPTBAS=0 IF OPTION BASE 1                   *
```

```
ARMA 03/14/81 ===========================================================================
ITEM    LOC    OBJECT CODE   SRC=CCALLO OBJ=GEXALL   9/11/1981 11:30 AM   PG 41
=========================================================================================
>>>>>>          FN ALLOCATION                                                   <<<<<<<<
7   044570                  *                  1 IF OPTION BASE 0                      *
782 044570                  *                                                          *
782 044570                  *                                                          *
782 044570                  ************************************************************
783 044570
784 044570 315                       OCT  315             ATTRIBUTE
785 044571
786 044571                  OPTIO.   BSS  0
787 044571 230                       BIN
788 044572                  OPTION   BSS  0
789 044572 116 220                   TSB  R16              CALC MODE ?
790 044574 362 046                   JOD  OPTERR           OPT BAS ILLEGAL
791 044576 176 260 346               LDBD R76,=OPTB        GET OPTB  FLAG     1 bei 1
791 044601 212
792 044602 366 040                   JNZ  OPTERR           JIF NOT THE 1ST TIME
793 044604 210                       ICB  R76
794 044605 262 346 212               STBD R76,=OPTB        FLAG WE ALREADY SEEN ONE
795 044610 174 271 311               LDMI R74,=PTR1-       FETCH OPTION BASE
 5  044613 377
    044614 176 074 241               LDM  R76,R74
79  044617 213                       DCM  R76
798 044620 215                       TCM  R76
799 044621 102 261 214               LDMD R2,=PGMOPT       GET OLD OPTBAS   0,0 bei 1
799 044624 210
800 044625 365 011                   JPS  OPTCHK           JIF DEFINED
801 044627                  OPT1     BSS  0
802 044627 176 263 214               STMD R76,=PGMOPT
802 044632 210
803 044633                  OPT2     BSS  0
804 044633 176 263 175               STMD R76,=OPTBAS       ELSE, SET OPTBAS   1,0 bei 1
804 044636 200
805 044637 236              OPTEX    RTN
806 044640
807 044640                  OPTCHK   BSS  0
808 044640                           DRP  !2
809 044640 076 301                   CMM  R2,R76           SAME OPTBAS ?
810 044642 367 373                   JZR  OPTEX            JIF YES
 11 044644                  OPTERR   BSS  0
    044644 316 377 377               JSB  =ERROR+
81  044647 036                       OCT  30D              ILLEGAL OPTION BASE
814 044650
815 044650                  ************************************************************
815 044650                  *                                                          *
816 044650                  *     SUBROUTINE TSTRM                                     *
817 044650                  *                                                          *
818 044650                  *       IN:                                                *
819 044650                  *         R70-72=ROOM REQUIRED                             *
820 044650                  *                                                          *
821 044650                  *       SCRATCH:                                           *
822 044650                  *         R75-77=WORKING REGISTERS                         *
823 044650                  *                                                          *
824 044650                  *       OUT:                                               *
825 044650                  *         CY=1 IF ROOM OK
```

```
>>>>>>        FN ALLOCATION                                                    <<<<<<<<
044650                           *        CY=0 IF NOT ENOUGH                           *
044650                           *                                                     *
044650                           *******************************************************
044650
044650                           TSTRM2   BSS   0
044650                           TSTRM    BSS   0
044650 175 261 022                        LDMD  R75,=NXTMEM
044653 200
044654 175 070 305 TSTRM1        SBM   R75,R70
044657 372 011                            JNC   TSTRMX       RTN IF NOT ENOUGH SPACE
044661 315 000 001                        SBM   R75,=0,1,0   LEAVE 256 BYTES FOR PARSER
044664 000
044665 372 003                            JNC   TSTRMX       JIF IF NOT ENOUGH SPACE
044667 321 025 200                        CMMD  R75,=LAVAIL  CHECK IF ENOUGH SPACE
044672 236                       TSTRMX   RTN
044673
044673                           ****TSTRM2  BSS  0
044673                           ****        LDMD R75,=SAVPT2
044673                           ****        JMP  TSTRM1
044673
```

```
>>>>>>      GLOBAL SUBROUTINES                                          <<<<<<<
1843 044673                *
1844 044673                *  SUBROUTINE FRMNAM
1845 044673                *
1846 044673                *  THIS ROUTINE PUT THE VARIABLE NAME IN
1847 044673                *  R36-R37 IN THE FORM
1848 044673                *
1849 044673                *    XYTTNNNN
1850 044673                *    RDFAAAAA
1851 044673                *
1852 044673                *  WHERE:   X      = STRING FLAG
1853 044673                *           Y      = ARRAY FLAG
1854 044673                *           TT     = TYPE
1855 044673                *                0 - REAL
1856 044673                *                1 - INTEGER
1857 044673                *                2 - SHORT
1858 044673                *                3 - AVAIL
1859 044673                *           AAAAA  = 1ST CHR NAME - 100 OCT
1860 044673                *           R      = REMOTE FLAG
1861 044673                *           D      = TRACE FLAG
1862 044673                *           F      = FUNCTION FLAG
1863 044673                *           NNNN   = 2ND CHR NAME - 60 OCT
1864 044673                *                    OR 12 IF BLANK
1865 044673                *    INPUT:
1866 044673                *           R23    = TOKEN OR TOKEN + 200 IF FUNC
1867 044673                *           DIMFLG SET    0   -  REAL
1868 044673                *                         20  -  INTEGER
1869 044673                *                         40  -  SHORT
1870 044673                *
1871 044673                *    OUTPUT:
1872 044673                *           R37=HEADER
1873 044673                *
1874 044673                FRMNAM  BSS  0
1875 044673                FRMNA#  BSS  0
1876 044673 137 023 240           LDB R37,R23          TOKEN
1877 044676 212                   DCB R37              TOKEN - 1
1878 044677 231                   BCD
1879 044700 204                   LLB R37              POSITION ARRAY,STR,FUNC
1880 044701 230                   BIN
1881 044702 204                   LLB R37
1882 044703 204                   LLB R37
1883 044704 322 015 204           ADBD R37,=DIMFLG     ADD DIM FLAG
1884 044707 123 220               TSB  R23            FUNC TOKEN?
1885 044711 365 002               JFS  FRM1           JIF NOT
1886 044713 137 210               ICB  R37            SET FUNC FLAG
1887 044715 236           FRM1    RTN
1888 044716                *************************************************
1888 044716                *                                                *
1889 044716                *  SUBROUTINE ZROMEM                             *
1890 044716                *                                                *
1891 044716                *  THIS ROUTINE WILL ZERO OR BLANK               *
1892 044716                *  A SPECIFIED AREA OF MEMORY.                   *
1893 044716                *                                                *
1894 044716                *  INPUTS
```

```
>>>>>>          GLOBAL SUBROUTINES                                                  <<<<<<<<
1895 044716                      *                                                          *
1896 044716                      *  R36     =  FWA                                          *
1897 044716                      *  R56     = # BYTES                                       *
1898 044716                      *  R23     = 3 THEN BLANKS ELSE ZEROS                      *
1898 044716                      *                                                          *
1898 044716                      ************************************************************
1899 044716
1900 044716 160 250 040 ZROMEM  LDB R60,=40
1901 044721 161 060 241         LDM R61,R60         blank R60-R67
1902 044724 123 310 003         CMB R23,=3          string?
1903 044727 367 002             JZR ZROM1-          jif yes
1904 044731 160 223     ZROM-   CLM R60             zero R60
1905 044733 316 360 111 ZROM1-  JSB =ZERO11         go do 10's
1906 044736 156 313 010         ADM R56,=10,0       restore count
1906 044741 000
1907 044742 156 221     ZEROL2  TSM R56
1908 044744 367 007             JZR ZEROX           exit if done
1909 044746 160 036 344 ZEROL3  PUBD R60,+R36       blank some
1910 044751 156 213             DCM R56
1911 044753 366 371             JNZ ZEROL3          LOOP IF MORE
1912 044755 236         ZEROX   RTN
1913 044756 160 223     ZERO10  CLM R60
1914 044760 100 250 060 ZERO11  LDB R0,=60
1915 044763 176 251 010         LDM R76,=10,0
1915 044766 000
1916 044767 156 076 305 ZERO1-  SBM R56,R76
1917 044772 372 361             JNC ZEROX           JIF NOT R76 LEFT
1918 044774 101 036 345         PUMD R*,+R36
1919 044777 360 366             JMP ZERO1-          LOOP TO DO MORE
1920 045001                      *
1921 045001                      *  ROUTINE RTOIN
1922 045001                      *
1923 045001                      *  REAL TO BCD INTEGER CONVERSION
1924 045001                      *
1925 045001                      *  INPUT R60
1926 045001                      *
1927 045001                      *  OUTPUT R65
1928 045001                      *
1929 045001 160 070 243 RTOIN   STM R60,R70         SAVE ORIG FOR HOMER
1930 045004 231         RTOIN1  BCD                 ALTERNATE ENTRY
1931 045005 235                 CLE
1932 045006 164 310 377         CMB R64,=377        INT
1933 045011 373 102             JCY RTORTN          DONE IF YES
1934 045013 160 040 243         STM R60,R40
1935 045016 316 377 377         JSB =INFR3          SEP & INFRA
1936 045021 160 223             CLM R60             ANS IF SMALL
1937 045023 136 211             ICM R36             EXP = -1
1938 045025 367 025             JZR RNDIT           JIF YES
1939 045027 375 032             JLN TSTBIG          NEG EXP
1940 045031 150 221             TSM R50             ZERO MANTISSA?
1941 045033 367 051             JZR BIG1-           JIF YES
1942 045035 150 205     RNDSHF  LLM R50             BRING TRAILING ZERO
   3 045037 367 006             JZR BIG2            EXP IS TOO BIG
```

```
1944 045041 136 213                  DCM R36            DEC EXP
1945 045043 366 370                  JNZ RNDSHF         LOOP IF MORE TO BRING
1946 045045 360 014                  JMP TSTBIG
1947 045047 153 250 001 BIG2         LDB R53,=1         FORCE OVERFLOW
1948 045052 360 007                  JMP TSTBIG
1949 045054 147 312 120 RNDIT        ADB R47,=50C       ROUND
1950 045057 372 002                  JNC TSTBIG         JIF NO OVERFLOW
1951 045061 150 211                  ICM R50          , ELSE INCR ANS
1952 045063 235             TSTBIG   CLE
1953 045064 234                      ICE
1954 045065 165 251 231              LDM R65,=99C,99C,9C IN CASE OF ERROR
1954 045070 231 011
1955 045072 153 221                  TSM R53
1956 045074 366 011                  JNZ BIG1           TOO BIG
1957 045076 152 310 020              CMB R52,=10C
1958 045101 373 004                  JCY BIG1           JIF > 99999
1959 045103 165 050 241              LDM R65,R50        ANS TO 65
1960 045106 235             BIG1-    CLE
1961 045107 132 220         BIG1     TSB R32
1962 045111 376 002                  JRZ RTORTN         JIF POS
1963 045113 165 215                  TCM R65            ELSE COMPL
1964 045115 236             RTORTN   RTN
1965 045116                 *  SUBROUTINE CONINT
1966 045116                 *
1967 045116                 *  CONVERTS THE F.P. CONSTANT IN R60-R67
1968 045116                 *  TO AN OCTAL CONSTANT IN R76.
1969 045116                 *
1970 045116                 *  OTHER REEGISTERS USED
1971 045116                 *
1972 045116                 *  R32  R34 R36 R50
1973 045116                 *
1974 045116 232             CONINT   SAD                SAVE ARP DRP
1975 045117                 ****  FIRST TRY A QUICKIE FOR SMALL POS #'S  ****
1976 045117 231                      BCD
1977 045120 132 222                  CLB R32          RESET SIGN
1978 045122 176 223                  CLM R76
1979 045124 235                      CLE
1980 045125 167 200                  ELB R67          MSB
1981 045127 160 221                  TSM R60          SEE IF INTEGER 0 TO 9
1982 045131 366 004                  JNZ CONIN1       JIF NO
1983 045133 176 200                  ELB R76          ELSE ANS TO 76
1984 045135 237                      PAD
1985 045136 236                      RTN
1986 045137 167 202         CONIN1   ERB R67            RESTORE ORIG VAL
1987 045141                 ******* PUMD R32,+R6
1988 045141 140 006 345              PUMD R40,+R6
1989 045144 316 004 112              JSB =RTOIN1        REAL TO INT
1990 045147 176 223                  CLM R76
1991 045151 167 220                  TSB R67            NEG?
1992 045153 374 002                  JLZ CONIN2         JIF NO
1993 045155 165 215                  TCM R65            ELSE COMPL
1994 045157 136 251 000 CONIN2       LDM R36,=0,200     100000
1994 045162 200
```

```
>>>>>>      GLOBAL SUBROUTINES                                    <<<<<<<<
1995 045163 134 251 017          LDM R34,=17,0      GET LARGEST EXP
1995 045166 000
1996 045167              NXTVAL  BSS 0
1997 045167              ******* BIN
1998 045167              ******* LLB R#             EXP * 2
1999 045167 231                  BCD
2000 045170 155 034 265          LDMD R55,X34,TABD  GET BCD VAL
2000 045173 242 112
2001 045175              ******* STM R56,R55
2002 045175              ******* CLB R57
2003 045175 165 055 305 VALLOP   SBM R65,R55
2004 045200 372 022              JNC NXTCON         JIF VAL < TABD
2005 045202 230                  BIN
2006 045203 176 036 303          ADM R76,R36        INC ANS
2007 045206 231                  BCD
2008 045207 361 364              JNO VALLOP         REPEAT IF NO OVERFLOW
2009 045211 176 251 377 BIGNUM   LDM R76,=377,377
2009 045214 377
2010 045215 235                  CLE
2011 045216 233                  DCE
2012 045217              NGTST   BSS 0
2013 045217              ******* TSB R32
2014 045217              ******* JRZ CONOUT
2015 045217              ******* BIN
2016 045217              ******* TCM R76
2017 045217 140 006 343 CONOUT   POMD R40,-R6
2018 045222              ******* POMD R32,-R6       RESTORE R32
2019 045222 237                  PAD                RESTORE ARP DRP
2020 045223 236                  RTN
2021 045224              NXTCON  BSS 0
2022 045224                      DRP 165
2023 045224                      ARP 155
2024 045224 303                  ADM R65,R55        RESTORE
2025 045225 230                  BIN
2026 045226 137 207              LRM R37            OCT VAL \ 2
2027 045230 207                  LRM                OCT VAL \ 4
2028 045231 207                  LRM                OCT VAL \ 8
2029 045232 134 314 003          SBB R34,=3         DEC EXP BY 3
2030 045235 365 330              JPS NXTVAL         JIF NOT DONE
2031 045237 235                  CLE
2032 045240 360 355              JMP NGTST          GO SEE IF NEG
2033 045242 001 000 000 TABD     DEC 000001         2^0
2034 045245 010 000 000          DEC 000008         2^3
2035 045250 144 000 000          DEC 000064         2^6
2036 045253 022 005 000          DEC 000512         2^9
2037 045256 226 100 000          DEC 004096         2^12
2038 045261 150 047 003          DEC 032768
2039 045264
```

```
2041 045264
2042 045264                    ************************************************************
2042 045264                    *                                                          *
2043 045264                    *     FETCH SIMPLE VARIABLE                                *
2044 045264                    *                                                          *
2045 045264                    *       INPUTS                                             *
2046 045264                    *         R65     =  REL ADDR OF VARIABLE                  *
2047 045264                    *                                                          *
2048 045264                    *       OUTPUTS                                            *
2049 045264                    *         R75     = ABS ADDR OF VARIABLE VALUE             *
2050 045264                    *         R46     = HEADER                                 *
2051 045264                    *         R60     = VARIABLE VALUE                         *
2052 045264                    *         R70     = ABS ADDR OF VAR NAME                   *
2053 045264                    *         PTR2    = POINTS TO VARIABLE VALUE               *
2053 045264                    *                                                          *
2053 045264                    ************************************************************
2054 045264
2055 045264                    FETSV   BSS 0
2056 045264 316 305 112               JSB =FETSVA          GET ADDR
2057 045267 316 374 112 FETSVX  JSB =FNUM            GET NUM
2058 045272 167 310 377 FETIND  CMB R67,=377          INDEF?
2059 045275 366 074               JNZ FEXIT2           JIF NO
2060 045277 160 223        FEXER   CLM R60              ZERO RESULT
2061 045301 316 377 377            JSB =ERROR+
2062 045304 007                    OCT 7D               INDEF VARIABLE
2063 045305
2064 045305
2065 045305                    ************************************************************
2065 045305                    *                                                          *
2066 045305                    *    FETCH VARIABLE ADDRESS                                *
2067 045305                    *                                                          *
2068 045305                    *      IN:                                                 *
2069 045305                    *        R65=POINTER TO VAR AREA                           *
2070 045305                    *                                                          *
2071 045305                    *      OUT:                                                *
2072 045305                    *        R46=HEADER                                        *
2073 045305                    *        R70=ABSOLUTE ADDR OF VAR NAME                     *
2074 045305                    *        PTR2=POINTS TO VAR VALUE                          *
2074 045305                    *                                                          *
2074 045305                    ************************************************************
2075 045305
2076 045305 230             FETSVA  BIN
2077 045306 165 323 006             ADMD R65,=FWCURR   make it absolute
2077 045311 200
2078 045312 263 314 377             STMD R65,=PTR2     POINT TO VAR STORAGE
2079 045315 147 270 315             LDBI R47,=PTR2-    GET HEADER
2079 045320 377
2080 045321 165 271 315             LDMI R65,=PTR2-    GET NAME PTR
2080 045324 377
2081 045325
2082 045325                    ****************************************************
2083 045325                    ** FOR CALC VAR, THE PTR IS PTING TO WRONG PLACE *
2084 045325                    ** HOWERVER, THIS INFORMATION IS IGNORED ANYWAY. *
```

```
2085 045325                *********************************************
2086 045325                *****   JNG   PRGNM        JIF PROG VERIABLE
2087 045325                ******  ADMD  R65,=CALVRS  ABS ADR OF CALC VAR NAME
2088 045325                ******  JMP   FETSV1
2089 045325                *PRGNM  BSS   0
2090 045325                ******  DRP   !65
2091 045325
2092 045325 323 006 200           ADMD R65,=FWCURR  ABS ADR OF PRG VAR NAME
2093 045330                FETSV1  BSS   0
2094 045330                        DRP   !65
2095 045330 070 243                STM   R65,R70      SAVE IT
2096 045332 103 047 240            LDB   R3,R47       COPY HEADER
2097 045335 317 004                ANM   R3,=4        ISOLATE REMOTE
2098 045337 367 024                JZR   FEXIT        JIF NOT REMOTE
2099 045341 165 271 315            LDMI R65,=PTR2-    GET PTR TO REMOTE VALUE
2099 045344 377
2100 045345 263 314 377            STMD R65,=PTR2     SET POINTER
2101 045350 103 270 315            LDBI R3,=PTR2-     GET REMOTE HEADER
2101 045353 377
2102 045354 317 002                ANM   R3,=2        ISOLATE TRACE BIT
2103 045356 147 003 224            ORB   R47,R3       SET IN VAR HEADER
2104 045361 165 271 315            LDMI R65,=PTR2-    SKIP 3 DUMMY BYTES
2104 045364 377
2105 045365                FEXIT   BSS   0
2106 045365 146 260 057            LDBD R46,=TRCFL2  GET TRACE ALL FLAG
2106 045370 200
2107 045371 047 224                ORB   R46,R47      OR IN HEADER
2108 045373 236            FEXIT2  RTN
2109 045374
2110 045374                ***************************************************
2110 045374                *                                                 *
2111 045374                *    FETCH NUMERICAL VALUE                        *
2112 045374                *                                                 *
2113 045374                *      IN:                                        *
2114 045374                *        PTR2=POINTS TO VAR VALUE                 *
2115 045374                *        R46=HEADER                               *
2116 045374                *                                                 *
2117 045374                *      OUT:                                       *
2118 045374                *        R60=NUMERICAL VAR VALUE                  *
2119 045374                *        PTR2=POINTS TO AFTER VALUE               *
2119 045374                *                                                 *
2119 045374                ***************************************************
2120 045374
2121 045374                FNUM    BSS   0
2122 045374 231                    BCD
2123 045375 136 046 240            LDB R36,R46        COPY TYPE
2124 045400 206                    LRB R36             TYPE LOWER
2125 045401 230                    BIN
2126 045402 317 003 000            ANM R36,=3,0        AND ISOLATE
2127 045405 212                    DCB   R36          DEC TYPE
2128 045406 364 060                JNG   FREAL        JIF REAL
2129 045410 367 063                JZR   FINT          JIF INT
2130 045412                *  SHORTS TO REAL
```

```
>>>>>       VARIABLE FETCH STORE                                    <<<<<<<
   045412 164 271 315          LDMI R64,=PTR2-     GET SHORT
31 045415 377
32 045416 136 223               CLM R36             SIGN EXP
33 045420 060 243               STM R36,R60         CLEAR CRAP
34 045422 062 243               STM R36,R62         MORE CRAP
35 045424 064 240               LDB R36,R64         EXP
36 045426                   *   AND STILL MORE CRAP
37 045426 137 242               STB R37,R64
38 045430 231                   BCD
39 045431 167 310 377           CMB R67,=377        UNINITIALIZED DATA
40 045434 373 030               JCY FEXRTN          JIF YES
41 045436 160 205               LLM R60             SIGNS TO E
42 045440 132 200               ELB R32             TO 32
43 045442 220                   TSB R32
44 045443 263 002               JEV FEX3A           JIF POS
45 045445 136 215               TCM R36             ELSE COMPL SIGN
46 045447 137 204       FEX3A   LLB R37             POSITION UPPER EXP
47 045451 230                   BIN
48 045452 132 206               LRB R32             SIGN MANT
   045454 231                   BCD
50 045455 220                   TSB R32
51 045456 263 003               JEV FEX3B           JIF POS
52 045460 137 312 011           ADB R37,=9C         SET SIGN NEG
53 045463 136 060 243 FEX3B     STM R36,R60         SIGN/EXP TO 60
54 045466 230         FEXRTN    BIN
55 045467 236                   RTN
56 045470
57 045470               FREAL   BSS  0
58 045470 160 271 315           LDMI R60,=PTR2-     FETCH REAL VALUE
58 045473 377
59 045474 236                   RTN
60 045475
61 045475               FINT    BSS  0
62 045475 164 250 377           LDB  R64,=377       SET INT FLAG
63 045500 165 271 315           LDMI R65,=PTR2-     FETCH INT VALUE
63 045503 377
64 045504 236                   RTN
65 045505
   045505               *********************************************************
66 045505               *                                                       *
67 045505               *    FETCH ADDRESS OF ARRAY ELEMENT                     *
68 045505               *                                                       *
69 045505               *      IN:                                              *
70 045505               *        R12=STACK POINTER                              *
71 045505               *                                                       *
72 045505               *                                                       *
73 045505               *      STACK IN **                                      *
74 045505               *                  PTR TO VARIABLE AREA(3 BYTES)        *
75 045505               *                  ROW DIM(2 BYTES)                     *
76 045505               *                  COL DIM(2 BYTES OPTIONAL)            *
77 045505               *                  DIM FLAG (0 FOR 2D, 1 FOR 1D)        *
78 045505               *        R12------>                                     *
79 045505               *
```

```
>>>>>>        VARIABLE FETCH STORE                                              <<<<<<<<
2180 045505                    *                                                       *
2181 045505                    *        OUT:                                           *
2182 045505                    *          R75=ABS ADR OF ELEMENT VALUE                 *
2183 045505                    *          R46=HEADER                                   *
2184 045505                    *          R70=ABS ADR OF VAR NAME                      *
2185 045505                    *          PTR2=ADR OF VAR VALUE                        *
2186 045505                    *          R20=MAX LEN IF STRING ARRAY                  *
2186 045505                    *                                                       *
2186 045505                    *********************************************************
2187 045505
2188 045505                    FETAV    BSS 0
2189 045505
2190 045505
2191 045505 230                FETAVA   BIN
2192 045506 122 012 342                POBD R22,-R12       DIM FLAG
2193 045511 006 344                    PUBD R22,+R6        SAVE IT
2194 045513 126 223                    CLM  R26            RESET COLUMN
2195 045515
2196 045515                    *S/DLOP   JSB  =ONEX          MAX COL OR ROW
2197 045515                    *         JEN  SUBERR         JIF OVERFLOW
2198 045515                    *         TSB  R32            TST SIGN
2199 045515                    *         JRZ  SUBOK          JIF POSITIVE
2200 045515                    *SUBERR   BSS  0
2201 045515                    *         LDMD R12,=TOS       CLEAN UP STACK
2202 045515                    *         JMP  DIMER+         REPORT ERROR
2203 045515                    *SUBOK    BSS  0
2204 045515                    *         STM  R46,R40
2205 045515                    *         TSB  R22            1 DIM ?
2206 045515                    *         JOD  S/DON          JIF YES
2207 045515                    *         ICB  R22            SO WE EXIT NEXT TIME
2208 045515                    *         LDM  R26,R40        SAVE COL
2209 045515                    *         JMP  S/DLOP         GO DO ROW
2210 045515
2211 045515                    S/DLOP   BSS  0
2212 045515 120 222                    CLB  R20            RESET FLAG
2213 045517 122 220                    TSB  R22            1 DIM ?
2214 045521 362 005                    JOD  ONED           JIF YES
2215 045523 316 377 377                JSB  =TWOX          ROW IN 46, COL IN 26
2216 045526 360 003                    JMP  SUBOK?
2217 045530                    ONED     BSS  0
2218 045530 316 377 377                JSB  =TWOX0         ROW IN 46
2219 045533                    SUBOK?   BSS  0
2220 045533 370 124                    JEN  DIMER1         JIF OVFL/NEG
2221 045535 146 040 243                STM  R46,R40        SAVE ROW
2222 045540
2223 045540 126 042 243 S/DON          STM  R26,R42        SAVE COL
2224 045543 165 012 343                POMD R65,-R12
2225 045546 316 305 112                JSB  =FETSVA        GET VARIABLE ADDRESS
2226 045551                    *    NAME FORM IN R46     ****************
2227 045551 144 006 342                POBD R44,-R6        DIM FLAG
2228 045554 116 344                    PUBD R16,+R6        SAVE CURR STATUS
2229 045556 165 261 314                LDMD R65,=PTR2      GET VAR ADDR
2229 045561 377
```

```
>>>>>      VARIABLE FETCH STORE                                    <<<<<<<<
2230  045562 321 022 200          CMMD R65,=NXTMEM    IS IT A CALC VAR ?
2231  045565 372 005              JNC  CALCV          JIF YES
2232  045567 116 220              TSB  R16            IS IT PGM MODE ?
2233  045571 363 001              JEV  CALCV          JIF YES
2234  045573 222                  CLB  R16            PRETEND PGM MODE
2235  045574 316 135 111 CALCV    JSB  =CUROPT        SET OPTION BASE
2236  045577 116 006 342          POBD R16,-R6        RESTORE STATUS
2237  045602 132 261 175 NTRMT    LDMD R32,=OPTBAS
2237  045605 200
2238  045606 030 243              STM  R32,R30        COPY IT
2239  045610 213                  DCM  R32
2240  045611 215                  TCM  R32            ZERO IF OPT BAS 0 ELSE 1
2241  045612 165 271 315          LDMI R65,=PTR2-     TRASH TOT SIZE
2241  045615 377
2242  045616 166 271 315          LDMI R66,=PTR2-     POP MAX ROW
2242  045621 377
2243  045622 124 040 241          LDM  R24,R40
2244  045625 316 376 113          JSB  =TSTDIM
2245  045630 166 271 315          LDMI R66,=PTR2-     POP MAX COL
2245  045633 377
2246  045634 311 377 377          CMM  R66,=377,377   ONE DIM?
2247  045637 366 026              JNZ FETA1           JIF NOT VECTOR
2248  045641 144 220              TSB R44             ONE DIM ?
2249  045643 363 016              JEV DIMER+          JIF NO
2250  045645 156 040 241          LDM R56,R40       ROW
2251  045650 032 305              SBM R56,R32         MINUS OPTION BASE
2252  045652 054 243              STM  R56,R54
2253  045654 223                  CLM  R56
2254  045655 360 046              JMP VCTOR           GO MPY
2255  045657 106 213      DIMERR  DCM R6            TRASH ONE RETURN
2256  045661 106 213      DIMER1  DCM  R6
2257  045663             DIMER+   BSS  0
2258  045663 316 377 377          JSB =ERROR+
2259  045666 067                  OCT 55D             SUBSCRIPT OUT OF RANGE
2260  045667
2261  045667 124 042 241 FETA1    LDM R24,R42         COL#
2262  045672 144 220              TSB R44             TWO DIM ?
2263  045674 362 365              JOD DIMER+          JIF NO
2264  045676 316 376 113          JSB =TSTDIM
2265  045701 176 040 241          LDM R76,R40         ROW
2266  045704 032 305              SBM R76,R32         ROW - OPT BASE
2267  045706 124 305              SBM R24,R32         COL - OPT BASE
2268  045710 166 030 303          ADM R66,R30         MAX COL + OPTBAS
2269  045713 154 223              CLM R54
2270  045715 316 377 377          JSB =INTMUL         ROW * MAX COL
2271  045720 126 223              CLM R26
2272  045722 154 024 303          ADM R54,R24         + COL
2273  045725             VCTOR    BSS  0
2274  045725 157 046 240          LDB  R57,R46        TYPE
2275  045730 364 057              JNG  FESTVA         JIF STRING ARRAY
2276  045732 231                  BCD                 FOR FASTER SHIFT
2277  045733 206                  LRB  R57            TYPE RT
2278  045734 317 003              ANM  R57,=3         ISOLATE TYPE
```

```
>>>>>>      VARIABLE FETCH STORE                                       <<<<<<<<<
045736 230                          BIN
045737 175 054 241                  LDM  R75,R54        USE 3 BYTES
045742 065 243                      STM  R75,R65        SAVE A COPY
045744 205                          LLM  R75            * 2
045745 157 212                      DCB  R57
045747 366 004                      JNZ  VCTOR1         JIF NOT INT
045751 175 303                      ADM R75,R65         *3
045753 360 010                      JMP VCTOR2          GO FINISH
045755 175 205      VCTOR1          LLM R75             * 4
045757 157 212                      DCB R57
045761 367 002                      JZR VCTOR2          JIF SHORT
045763 175 205                      LLM R75             * 8
045765              VCTOR2          BSS  0
045765              SKPBY2          BSS  0
045765 175 215                      TCM  R75           NEGATE OFFSET
045767 323 314 377                  ADMD R75,=PTR2     ADD BASE
045772 263 314 377                  STMD R75,=PTR2     UPDATE POINTER
045775 236                          RTN
045776
045776 166 024 301  TSTDIM          CMM R66,R24         MAX-GOAL
046001 372 254                      JNC DIMERR
046003 124 032 301                  CMM R24,R32         GOAL-BASE
046006 372 247                      JNC DIMERR
046010 236                          RTN
046011
046011              FESTVA          BSS  0
046011                              DRP  !57
046011 222                          CLB  R57           FOR CALC OFFSET
046012 170 006 345                  PUMD R70,+R6       SAVE ADR OF NAME
046015 146 344                      PUBD R46,+R6       SAVE HEADER
046017 166 271 315                  LDMI R66,=PTR2-    GET MAX LEN
046022 377
046023 020 243                      STM  R66,R20       SAVE MAX LEN
046025 313 002 000                  ADM  R66,=2,0      ELEMENT SIZE
046030 316 235 102                  JSB  =ARYSPC       CALC OFFSET
046033 146 006 342                  POBD R46,-R6       RESTORE HEADER
046036 170 343                      POMD R70,-R6       RESTORE ADR OF NAME
046040 175 055 241                  LDM  R75,R55       COPY OFFSET
046043 215                          TCM  R75           NEGATE OFFSET
046044 323 314 377                  ADMD R75,=PTR2     ADD BASE
046047 213                          DCM  R75           PASS OVER ACT LEN
046050 213                          DCM  R75
046051 263 314 377                  STMD R75,=PTR2     SET POINTER
046054 236                          RTN
046055
046055
046055
046055              *
046055              *
046055              *
046055              *   ROUTINE STOSV
046055              *
046055              *   THIS ROUTINE WILL STORE A VALUE IN
```

```
ARMA 03/14/81 ================================================================================
 ITEM    LOC    OBJECT CODE  SRC=CCALL3 OBJ=GEXALL   9/11/1981 11:30 AM  PG 53
================================================================================
>>>>>       VARIABLE FETCH STORE                                          <<<<<<<<
2330  046055                *  THE  SPECIFIED SIMPLE VARIABLE.
2332  046055                *
2333  046055                *
2334  046055                *  INPUTS
2335  046055                *
2336  046055                *  R12   =  STACK IN
2337  046055                *
2338  046055                **  STACK IN **
2339  046055                *
2340  046055                *                          ABS ADDR OF VALUE (3 BYTES)
2341  046055                *                          ROW # (2 BYTES ONLY IF TRACE ARRAY
2342  046055                *                          COL # (2 BYTES ONLY IF TRACE ARRAY
2343  046055                *                          DIM FLG(1 BYTE ONLY IF TRACE ARRAY
2344  046055                *                                (ODD FOR 1-D, EVEN FOR 2-D)
2345  046055                *                          ABS ADDR OF NAME  (3 BYTES)
2346  046055                *                          HEADER (1 BYTE)
2347  046055                *                          VALUE TO STORE (8 BYTES)
2348  046055                *           R12------->
2349  046055                *
2350  046055                **  STACK OUT **
2351  046055                *                          VALUE JUST STORED IF MULT STORE
2352  046055                *           R12------->
2353  046055                *
2354  046055                *
2355  046055                ****  MULTI STORE NUM ATTRIBUTE TABLE  **********
2356  046055 043                    OCT 43
2357  046056                ********************************************************
2358  046056                STOSVM  BSS  0                  LET IT FALL THROUGH
2359  046056                ****  STORE NUM ATTRIBUTE TABLE  ********************
2360  046056 031                    OCT 31
2361  046057                ********************************************************
2362  046057                STOSV   BSS  0
2363  046057 160 012 343            POMD R60,-R12           GET REAL /INT
2364  046062 006 345                PUMD R60,+R5            SAVE IT
2365  046064 146 012 342            POBD R46,-R12           get header
2366  046067 344                    PUBD R46,+R12           SAVE IT
2367  046070 231                    BCD
2368  046071 206                    LRB R46
2369  046072 317 003 000            ANM R46,=3,0            ISOLATE TYPE
2370  046075 230                    BIN
2371  046076 006 344                PUBD R46,+R5            save it
2372  046100 367 014                JZR STSV1R              JIF REAL
2373  046102 212                    DCB R46
2374  046103 766 013                JNZ STSV3               JIF NOT INT
2375  046105 316 001 112            JSB =RTOIN              REAL TO INT
2376  046110 371 004                JEZ STOSVJ              JIF OK
2377  046112 316 377 377            JSB =ERROR
2378  046115 002                    OCT 2D                  OVERFLOW
2379  046116                STOSVJ  BSS 0
2380  046116 760 154        STSV1R  JMP STSV1
2381  046120                *  REAL TO SHORT
2382  046120 132 222        STSV3   CLB R32                 SIGNS FLAG
2383  046122 164 310 377            CMB R64,=377            INT?
```

```
>>>>>>        VARIABLE FETCH STORE                                              <<<<<<<<
2384 046125 231                       BCD
2385 046126 766 030                   JNZ STSV3Z          JIF NO
2386 046130 136 222                   CLB R36             EXP IF MANT 0
2387 046132 165 221                   TSM R65
2388 046134 374 005                   JLZ STSV3X          JIF NOT NEG
2389 046136 132 210                   ICB R32
2390 046140 210                       ICB                 MANT SIGN NEG
2391 046141 165 215                   TCM R65             COMPL
2392 046143 367 011           STSV3X  JZR STSV3Y          JIF MANT = 0
2393 046145 136 250 005               LDB R36,=5          EXP=5(MAX)
2394 046150 136 212           STSV3L  DCB R36             DEC EXP
2395 046152 165 205                   LLM R65
2396 046154 374 372                   JLZ STSV3L          JIF NOT NORM
2397 046156 760 105           STSV3Y  JMP STSV3D          GO FINISH
2398 046160 136 060 241       STSV3Z  LDM R36,R60         EXP/SIGN
2399 046163 137 206                   LRB R37             SIGN TO E
2400 046165 371 004                   JEZ STSV3A          JIF MANT POS
2401 046167 230                       BIN
2402 046170 132 210                   ICB R32             SET MANT NEG
2403 046172 210                       ICB R32
2404 046173 231               STSV3A  BCD
2405 046174 165 313 005               ADM R65,=5C,0,0        ROUND
2405 046177 000 000
2406 046201 372 005                   JNC STSV3B             JIF NO CARRY
2407 046203 167 250 020               LDB R67,=10C           ELSE ANS IS 1EN+1
2408 046206 136 211                   ICM R36                INCR EXP
2409 046210 137 310 005       STSV3B  CMB R37,=5C
2410 046213 372 011                   JNC STSV3C          JIF EXP POS
2411 046215 136 215                   TCM R36
2412 046217 230                       BIN
2413 046220 132 210                   ICB R32             SET SIGN NEG
2414 046222 231                       BCD
2415 046223 137 317 017               ANM R37,=17         TRASH EXTRA 9
2416 046226 136 311 000       STSV3C  CMM R36,=0,1
2416 046231 001
2417 046232 372 031                   JNC STSV3D          JIF EXP < 100
2418 046234 223                       CLM R36             ZERO EXP
2419 046235 165 223                   CLM R65             AND MANTISSA
2420 046237 132 220                   TSB R32             NEG EXP
2421 046241 363 006                   JEV OVFLOD          JIF OVERFLO
2422 046243 316 377 377               JSB =ERROR
2423 046246 001                       OCT 1D              UNDER FLOW
2424 046247 760 014                   JMP STS3C-          CONT IF DEFAULT
2425 046251 316 377 377       OVFLOD  JSB =ERROR
2426 046254 002                       OCT 2D              OVER FLOW
2427 046255 136 250 231               LDB R36,=99C
2428 046260 165 251 231               LDM R65,=99C,99C,99C
2428 046263 231 231
2429 046265                   STS3C-  BSS 0
2430 046265 132 203           STSV3D  ERM R32
2431 046267 167 203                   ERM R67             SIGNS TO UPPER 60
2432 046271 136 064 242               STB R36,R64         EXP TO 64
2433 046274                   STSV1   BSS  0
```

```
>>>>>       VARIABLE FETCH STORE                                    <<<<<<<
2435 046274 316 015 115            JSB  =TRACE0
2435 046277 155 012 343 TRAC-      POMD R55,-R12       VARIABLE ADDR
2436 046302 263 314 377            STMD R55,=PTR2      set pointer
2437 046305 132 223                CLM  R32            reset index
2438 046307 006 342                POBD R32,-R6        5 bytes per entry
2439 046311 034 243                STM  R32,R34        SAVE IT
2440 046313 205                    LLM  R32            *2
2441 046314 205                .   LLM  R32            *4
2442 046315 303                    ADM  R32,R34        *5
2443 046316 032 306 051            JSB  X32,STORV      store value
2443 046321 115
2444 046322 160 006 343            POMD R60,-R6        get value back
2445 046325 126 270 313            LDBI R26,=PTR1-+    NEXT TOKEN
2445 046330 377
2446 046331 310 024                CMB R26,=24         MULTI STORE?
2447 046333 266 003                JNZ SVXIT           JIF NO
2448 046335 160 012 345 PBAC       PUMD R60,+R12       FOR NEXT TIME
2449 046340 147 220     SVXIT      TSB R47             TRACE?
2450 046342 263 022                JEV SVRTN           JIF NO
2451 046344 316 367 114            JSB  =CALFNP        CAL VAR OR FN PAR ?
2452 046347 371 015                JEZ  SVRTN          JIF YES.  DON'T TRACE
2453 046351 146 036 240            LDB  R46,R36        PUT THE HEADER BACK
2454 046354 316 374 112            JSB  =FNUM          FETCH NUM AS STORED
2455 046357 146 036 242            STB  R46,R36        SAVE HEADER
2456 046362 204                    LLB  R46            POSITION ARRAY FLAG
2457 046363 316 377 377            JSB =TRCVBL
2458 046366             SVRTN      BSS  0
2459 046366 236                    RTN
2460 046367
2461 046367             *
2462 046367             *     E = 0  IF CALC VAR OR FUNC PARAMETER
2463 046367             *        # 0  IF PRGM VAR OF COM VAR
2464 046367             *
2465 046367
2466 046367             CALFNP  BSS  0                 CALC OF FN PAR ?
2467 046367 235                 CLE                    RESET FLAG
2468 046370 155 261 314         LDMD R55,=PTR2         GET VAR ADDR
2469 046373 377
2470 046374 321 022 200         CMMD R55,=NXTMEM       CALC VAR ?
2470 046377 372 013             JNC  NOTRAC            JIF YES
2471 046401 321 014 200         CMMD R55,=BOVAR        PGM VAR ?
2472 046404 372 005             JNC  PGMVAR            JIF YES
2473 046406 321 006 200         CMMD R55,=FWCURR       FN PAR ?
2474 046411 372 001             JNC  NOTRAC
2475 046413             PGMVAR  BSS  0
2476 046413 234                 ICE                    FLAG PGM VAR
2477 046414             NOTRAC  BSS  0
2478 046414 236                 RTN
2479 046415
2480 046415             TRACE0  BSS  0
2481 046415 230                 BIN
2482 046416 146 012 342         POBD R46,-R12          get header
2483 046421 036 242             STB R46,R36            save it
```

```
>>>>>          VARIABLE FETCH STORE                                          <<<<<<<<
046423 175 012 343           POMD R75,-R12        get abs adr of name
046426 070 243               STM R75,R70          save it
046430 146 206               LRB R46              trace?
046432 047 242               STB R46,R47          save it
046434 763 012               JEV TRACE1           jif no
046436 204                   LLB R46              ARRAY?
046437 204                   LLB R46
046440 220                   TSB R46
046441 765 005               JPS TRACE1           JIF NO
046443 120 012 342           POBD R20,-R12        DIM FLAG
046446 174 343               POMD R74,-R12        GET ROW/COL
046450 236          TRACE1   RTN
046451
046451              STORV    BSS  0
046451              STORR    BSS  0               store REAL
046451 160 273 317           STMI R60,=PTR2-+
046454 377
046455 236                   RTN
046456              STORI    BSS  0               store INTEGER
046456 165 273 317           STMI R65,=PTR2-+
046461 377
046462 236                   RTN
046463              STORS    BSS  0               store SHORT
046463 164 273 317           STMI R64,=PTR2-+
046466 377
046467 236                   RTN
046470              *
046470              *
046470              *  ROUTINE STOST
046470              *
046470              *  THIS ROUTINE WILL STORE THE STRING
046470              *  VALUE IN THE SPECIFIED STRING
046470              *  VARIABLE.
046470              *
046470              *  INPUTS
046470              *
046470              *  R12      = STACK PTR
046470              *
046470              **  STACK IN  **
046470              *            ROW # (2 BYTES ONLY IF TRACE ARRAY)
046470              *            COL # (2 BYTES ONLY IF TRACE ARRAY)
046470              *            DIM FLG(1 BYTE ONLY IF TRACE ARRAY)
046470              *                   (ODD FOR 1-D, EVEN FOR 2-D)
046470              *            ABS ADDR OF NAME(3 BYTES)
046470              *            HEADER(1 BYTE)
046470              *            MAX LEN OF SINK(2 BYTES)
046470              *            1ST CHAR ADDR OF SINK(3 BYTES)
046470              *            SUB STRING LEN (2 BYTES)
046470              *            SINK ADDR + OFFSET(3 BYTES)
046470              *            SOURCE LEN (2 BYTES)
046470              *            SOURCE ADDR(3 BYTES)
046470              *      R12---->
046470              *
```

```
                    VARIABLE FETCH STORE
                    **  STACK OUT  **
                    *                STR LEN
                    *                STRING PTR(2 BYTES)
                    *        R12---->
                    *
                    *
                    ****  MULTI STORE STRING ATTRIBUTE TABLE  **********
046470 043                   OCT 43
                    ************************************************************
046471              STOSTM   BSS  0                  LET IT FALL THROUGH
                    ****  STORE STRING ATTRIBUTE TABLE  ********************
046471 031                   OCT 31
                    ************************************************************
046472              STOST    BSS  0
046472 230                   BIN
046473 165 012 343            POMD R65,-R12          SOURCE POINTER
046476 124 343                POMD R24,-R12          SOURCE LENGTH
046500 316 257 115            JSB  =STOSTA
046503 155 261 310            LDMD R55,=PTR1         SAVE BASIC PC
046506 377
046507 006 345                PUMD R55,+R6
046511 165 263 310            STMD R65,=PTR1         SOURCE POINTER
046514 377
046515 175 263 314            STMD R75,=PTR2         SINK POINTER
046520 377
046521 124 221                TSM  R24               SOURCE LEN = 0 ?
046523 367 015                JZR  TBFILL            JIF YES
046525 000 243                STM  R24,R0            MAKE A COPY
046527              S/TLOP   BSS  0
046527 137 270 311            LDBI R37,=PTR1-        POP ONE SOURCE BYTE
046532 377
046533 272 315 377            STBI R37,=PTR2-        STORE IN SINK
046536 100 213                DCM  R0                DEC SOURCE COUNT
046540 366 365                JNZ  S/TLOP            LOOP IF MORE
046542              TBFILL   BSS  0
046542 126 221                TSM  R26               ANY TAIL BLANK FILL ?
046544 367 013                JZR  STRTN0            JIF NO
046546 137 250 040            LDB  R37,=BLANK        GET BLANK
046551              S/TLP1   BSS  0
046551 137 272 315            STBI R37,=PTR2-        BALNK FILL TAIL
046554 377
046555 126 213                DCM  R26               DEC COUNT
046557 366 370                JNZ  S/TLP1            LOOP IF MORE
046561 155 006 343 STRTN0     POMD R55,-R6           RESTORE BASIC PC
046564 263 310 377            STMD R55,=PTR1
046567              STRTN1   BSS 0
046567 124 006 345            PUMD R24,+R6           SAVE SOURCE LEN
046572 165 345                PUMD R65,+R6           SAVE SOURCE POINTER
046574 316 015 115            JSB  =TRACE0           GET TRACE INFORMATION
046577 147 220                TSB  R47               TRACE ?
046601 363 035                JEV  STREXX            JIF NOT
046603 316 367 114            JSB  =CALFNP           CAL VAR OR FN PAR ?
046606 371 030                JEZ  STREXX            JIF YES.  DON'T TRACE
```

```
>>>>>>      VARIABLE FETCH STORE                                    <<<<<<<<
2582 046610 146 006 344          PUBD R46,+R6        SAVE ARRAY FLG
2583 046613 316 377 377          JSB =TRCFP          TRACE LINE + NAME
2584 046616 146 250 044          LDB R46,=DOLLAR
2585 046621 012 344              PUBD R46,+R12       PUSH DOLLAR SIGN
2586 046623 006 342              POBD R46,-R6        STRING ARRAY ?
2587 046625 765 003              JPS  STRTN2         JIF NO
2588 046627 316 377 377          JSB  =TRARAY        PUSH SUBSCRIPT FOR STRING ARR
2589 046632 316 377 377 STRTN2   JSB =UNSTAK        DUMP EXP STACK
2590 046635 316 377 377          JSB =TRCLP          END TRACE PROCESSING
2591 046640             STREXX   BSS  0
2592 046640 173 006 343          POMD R73,-R6        POP SOURCE LEN/POINTER
2593 046643 170 270 313          LDBI R70,=PTR1-+    NEXT TOKEN
2593 046646 377
2594 046647 310 025              CMB  R70,=25        MULTI STORE ?
2595 046651 766 003              JNZ  STREXY         JIF NO
2596 046653 173 012 345          PUMD R73,+R12       PUSH SOURCE BACK
2597 046656             STREXY   BSS  0
2598 046656 236                  RTN
2599 046657
2600 046657             STOSTA   BSS  0
2601 046657 175 012 343          POMD R75,-R12       SINK POINTER
2602 046662 126 343              POMD R26,-R12       SINK LENGTH
2603 046664 145 343              POMD R45,-R12       BASE POINTER
2604 046666 120 343              POMD R20,-R12       MAX LENGTH
2605 046670 145 263 314          STMD R45,=PTR2      PT TO BASE
2605 046673 377
2606 046674 122 271 314          LDMI R22,=PTR2      GET ACT LENGTH
2606 046677 377
2607 046700 311 377 377          CMM  R22,=377,377   UNDEF ?
2608 046703 766 001              JNZ  STOS0          JIF NO
2609 046705 223                  CLM  R22
2610 046706             STOS0    BSS  0
2611 046706 155 045 241          LDM  R55,R45        COPY BASE
2612 046711 075 305              SBM  R55,R75        OFFSET OF SINK ADDR
2613 046713 130 055 241          LDM  R30,R55        USE ONLY 2 BYTES
2614 046716 032 243              STM  R30,R32        SAVE A COPY
2615 046720 126 024 301          CMM  R26,R24        SINK LEN > SOURCE LEN ?
2616 046723 373 015              JCY  STOS2          JIF YES
2617 046725 175 045 301          CMM  R75,R45        SINK=BASE ?
2618 046730 766 005              JNZ  STOS1          JIF NO
2619 046732 126 020 301          CMM  R26,R20        SINK LEN = MAX ?
2620 046735 367 013              JZR  STOERX         JIF YES
2621 046737 126 024 243 STOS1    STM  R26,R24        TRUNCATE
2622 046742 130 026 303 STOS2    ADM  R30,R26        NEW ACT
2623 046745 120 030 301          CMM  R20,R30        MAX > NEW ACT ?
2624 046750 373 011              JCY  STOS3          JIF YES
2625 046752             STOERX   BSS  0
2626 046752 316 377 377          JSB  =ERROR
2627 046755 070                  OCT  56D
2628 046756 102 006 343          POMD R2,-R6         TRASH ONE RETURN
2629 046761 760 204              JMP  STRTN1
2630 046763             STOS3    BSS  0
2631 046763 134 022 241          LDM  R34,R22        COPY OLD ACT LEN
```

```
>>>>>>         VARIABLE FETCH STORE                                          <<<<<<<<
2632 046766 032 305                SBM  R34,R32        OLD ACT > OFFSET ?
2633 046770 373 017                JCY  STOS3+         JIF YES
2634 046772 175 263 314            STMD R75,=PTR2      SINK ADDR
2634 046775 377
2635 046776 137 250 040            LDB  R37,=BLANK     GET A BLANK
2636 047001 137 272 316 BLKMID     STBI R37,=PTR2+     BLANK ONE POSITION
2636 047004 377
2637 047005 134 211                ICM  R34            INC COUNT
2638 047007 366 370                JNZ  BLKMID         LOOP IF MORE
2639 047011             STOS3+     BSS  0
2640 047011 175 045 301            CMM  R75,R45        SINK = BASE?
2641 047014 366 031                JNZ  NOPU0          JIF NOT
2642 047016 126 020 301            CMM  R26,R20        SINK LEN = MAX ?
2643 047021 366 024                JNZ  NOPU0          JIF NOT
2644 047023 124 221                TSM  R24            SOURCE LEN = 0 ?
2645 047025 366 011                JNZ  NOPU1          JIF NO
2646 047027 145 263 314            STMD R45,=PTR2      NEW ACT = 0
2646 047032 377
2647 047033 124 273 314            STMI R24,=PTR2
2647 047036 377
2648 047037 236                    RTN
2649 047040             *          GTO  STRTN1
2650 047040             NOPU1      BSS  0
2651 047040 124 026 243            STM  R24,R26        SINK = SOURCE
2652 047043 030 243                STM  R24,R30        NEW ACT LEN
2653 047045 360 005                JMP  NOPU2
2654 047047             NOPU0      BSS  0
2655 047047 122 030 301            CMM  R22,R30        OLD ACT > NEW ?
2656 047052 373 010                JCY  ZRTEST         JIF YES
2657 047054 145 263 314 NOPU2      STMD R45,=PTR2      SET NEW ACT LEN
2657 047057 377
2658 047060 130 273 314            STMI R30,=PTR2
2658 047063 377
2659 047064 126 024 305 ZRTEST     SBM  R26,R24        SINK LEN - SOURCE LEN
2660 047067 236                    RTN
2661 047070
```

```
  >>>>>        INITIALIZE PROGRAM                                                  <<<<<<<<
     047070                   INIPGM  BSS  0
2664 047070 230                       BIN
2665 047071 145 261 006               LDMD R45,=FWCURR
2665 047074 200
2666 047075 315 007 000               SBM  R45,=P.LEN     advance to P.LEN
2667 047100 000                       OCT  0
2668 047101 263 310 377               STMD R45,=PTR1      set pointer
2669 047104 271 311 377               LDMI R45,=PTR1-     get program length
2670 047107 366 011                   JNZ  INPRTN         jif non zero
2671 047111 261 006 200               LDMD R45,=FWCURR
2672 047114 325 014 200               SBMD R45,=BOVAR     calc length
2673 047117 273 312 377               STMI R45,=PTR1+     update length
2674 047122                   INPRTN  BSS  0
2675 047122 236               GNXTL   RTN
2676 047123
2677 047123                           FIN
```

SYMBOL	VALUE	TYPE	COUNT	
ADDCOM	043397	LCL	1	
A[illegible]NI	177777	EXT	1	
A[illegible]01	040037	LCL	0	<--- NOT REFERENCED??
ALLOC	040033	ENT	1	
ARYSPC	041235	LCL	3	
BASE0	041731	LCL	0	<--- NOT REFERENCED??
BCDINT	041323	ENT	1	
BIG1	045107	LCL	2	
BIG1-	045106	LCL	1	
BIG2	045047	LCL	1	
BIGNUM	045211	LCL	0	<--- NOT REFERENCED??
BININT	177777	EXT	2	
BLANK	000040	G EQU	2	
BLKMID	047001	LCL	1	
BOP	177777	EXT	2	
BOVAR	100014	G DAD	5	
CALCV	045574	LCL	2	
CALFNP	046367	LCL	2	
CALVRB	100030	G DAD	3	
COMER+	043240	LCL	2	
COMERR	043234	LCL	2	
C[illegible]FLG	101664	G DAD	2	
COMM	044450	ENT	2	
COMRTN	044477	LCL	1	
CONIN1	045137	LCL	1	
CONIN2	045157	LCL	1	
CONINT	045116	ENT	2	
CONOUT	045217	LCL	0	<--- NOT REFERENCED??
CONTOK	100047	G DAD	1	
CRDF0	044253	LCL	1	
CRDF1?	044406	LCL	0	<--- NOT REFERENCED??
CRDF?	044205	ENT	3	
CSTAR	100174	G DAD	4	
CURFUN	100164	G DAD	4	
CUROP1	044552	ENT	2	
CUROPS	044562	ENT	1	
CUROPT	044535	ENT	2	
DCLCOM	101663	G DAD	2	
DEFDEF	044201	LCL	1	
DEFEN1	040364	LCL	1	
D[illegible]END	040275	LCL	2	
DE[illegible]FN	043779	LCL	1	
DEFFN1	044002	LCL	1	
DEFFN2	044106	LCL	1	
DEFFN3	044112	LCL	1	
DEFFN4	044143	LCL	1	
DEFFNE	044101	LCL	1	
DFCOMN	044323	LCL	2	
DFERR	040365	LCL	4	
DFPAR1	104172	G DAD	10	
DFPAR2	104175	G DAD	2	
DIMAL	044510	ENT	2	
DIMCOM	044524	LCL	3	
DIMDIM	104160	G DAD	5	

SYMBOL	VALUE	TYPE	COUNT	
DIMER+	045667	LCL	2	
DIMER1	045661	LCL	1	
DIMERR	045657	LCL	2	
DIMFLG	102015	G DAD	7	
DOLLAR	000044	G EQU	1	
DUPDEF	044175	LCL	1	
ELSZ	042213	ENT	5	
ELSZR	042236	LCL	1	
ENOVDN	177777	EXT	1	
EOVAR	100017	G DAD	8	
EROM	040000	LCL	1	
ERROR	177777	EXT	7	
ERROR+	177777	EXT	12	
ERRSTP	103403	G DAD	3	
EXTRJM	177777	EXT	1	
FESTVA	046011	LCL	1	
FETA1	045667	LCL	1	
FETAV	045505	ENT	1	
FETAVA	045505	ENT	1	
FETIND	045272	ENT	1	
FETSV	045264	ENT	1	
FETSV1	045330	LCL	0	<--- NOT REFERENCED??
FETSVA	045305	ENT	3	
FETSVX	045267	ENT	1	
FEX3A	045447	LCL	1	
FEX3B	045463	LCL	1	
FEXER	045277	LCL	0	<--- NOT REFERENCED??
FEXIT	045365	LCL	1	
FEXIT2	045373	LCL	1	
FLARTN	045466	LCL	1	
FINT	045475	LCL	1	
FLERR	043703	LCL	1	
FLLOOP	043607	LCL	1	
FLOOP	043631	LCL	0	<--- NOT REFERENCED??
FNASN	040371	LCL	1	
FND2EX	043515	LCL	1	
FND2LP	043473	LCL	1	
FND2RT	043516	LCL	1	
FNDIT	040472	LCL	2	
FNDL2	043462	ENT	2	
FNDLAB	043604	LCL	1	
FNDLIN	177777	EXT	0	<--- NOT REFERENCED??
FNDLN	043444	LCL	1	
FNUM	045374	ENT	3	
FNUNDF	044444	LCL	1	
FOUND	043676	LCL	1	
FREAL	045479	LCL	1	
FRM1	044715	LCL	1	
FRMNA#	044673	LCL	0	<--- NOT REFERENCED??
FRMNAM	044673	ENT	4	
FUNCA1	044435	LCL	1	
FUNCAL	044407	LCL	1	
FWCURR	100006	G DAD	19	
FWPRGM	100003	G DAD	3	

```
ARNA 03/14/81 ================================================================
SYMBOL  VALUE   TYPE   COUNT     Symbol Table        9/11/1981 11:30 AM  PG 63
================================================================================
FWUSER  100000  G DAD     1
GNXT    177777    EXT     3
GINTDS  177401  G DAD     1
GINTEN  177400  G DAD     1
GNXTL   047122    LCL     0    <--- NOT REFERENCED??
GOSTOS  044405    LCL     1
GSBSCR  041307    LCL     4
GSBSR   041332    LCL     1
ILLCOM  044444    LCL     2
ILLDIM  041602    LCL     3
INFR3   177777    EXT     1
INIEND  042444    ENT     4
INIINT  042305    LCL     0    <--- NOT REFERENCED??
INILOP  042266    LCL     1
ININUM  042441    LCL     1
INIPGM  047079    ENT     1
INIREL  042303    LCL     0    <--- NOT REFERENCED??
INISHT  042307    LCL     0    <--- NOT REFERENCED??
INIST   042502    LCL     2
INIST+  042315    LCL     1
INISTR  042311    LCL     2
INITBL  042303    LCL     1
INIV    042237    LCL     2
INIVAR  042342    ENT     1
INIVLP  042362    ENT     2
INPRTN  047122    LCL     1
INT     044514    ENT     2
INTG?   042204    LCL     2
INTMUL  177777    EXT     4
INTORL  177777    EXT     1
JATAB   040210    ENT     2
JMPADR  104167  G DAD     2
LABEL   040466    LCL     1
LABEL?  043647    LCL     1
LABLAL  043537    LCL     1
LABTK?  177777    EXT     1
LAVAIL  100025  G DAD     6
LINEA1  043461    LCL     1
LINEAL  043426    LCL     4
LPTN    043577    LCL     1
L2      043600    LCL     1
MATCH?  043711    ENT     4
MCHRTN  043761    LCL     2
MDONE   043372    LCL     1
MINLOP  043734    LCL     1
MOVE    043345    LCL     2
MOVFL   041277    LCL     3
MOVNM   042760    LCL     1
NARD1   041400    LCL     5
NARDIM  041364    LCL     1
NEXTD   177777    EXT     1
NEXTL   043660    LCL     1
NGTST   045217    LCL     1
NMDONE  042775    LCL     1
```

```
NOFIND  040645     LCL    1
NOPU0   047047     LCL    2
NOPU1   047040     LCL    1
NOPU2   047054     LCL    1
NORM1   042702     LCL    1
NOROOM  043045     LCL    2
NOTARY  042463     LCL    1
NOTFN   041064     LCL    1
NOTFNC  042421     LCL    1
NOTIN   041424     LCL    1
NOTRAC  046414     LCL    2
NOTRL   041111     LCL    1
NOTRMT  042400     LCL    0   <--- NOT REFERENCED??
NTDEF   040613     LCL    1
NTRMT   045602     LCL    0   <--- NOT REFERENCED??
NTSTRF  044430     LCL    1
NXTBAD  040111     LCL    3
NXTCOM  100011   G DAD    3
NXTCON  045224     LCL    1
NXTGUD  040053     LCL    1
NXTLIN  177777     EXT    1
NXTSEM  100022   G DAD    6
NXTONE  040045     ENT    4
NXTVAL  045167     LCL    1
OLDPCR  101657   G DAD    2
ONEB    177777     EXT    0   <--- NOT REFERENCED??
ONED    045530     LCL    1
OPT1    044627     LCL    1
OPT2    044633     LCL    2
OPTAB   105346   G DAD    2
OPTBAS  100175   G DAD    8
OPTCHK  044640     LCL    1
OPTERR  044644     LCL    2
OPTEX   044637     LCL    2
OPTIO.  044571     ENT    1
OPTION  044572     LCL    1
OVFLOD  046251     LCL    1
P.LEN   000007   G EQU    1
PBAC    046335     LCL    0   <--- NOT REFERENCED??
PCR     101115   G DAD    3
PRINT   104214   G DAD    3
PGMVAR  046413     LCL    1
PTR1    177710   G DAD   44
PTR1+   177712   G DAD    1
PTR1-   177711   G DAD   23
PTR1-+  177713   G DAD    4
PTR2    177714   G DAD   43
PTR2+   177716   G DAD    3
PTR2-   177715   G DAD   60
PTR2-+  177717   G DAD    7
REAL    044510     ENT    1
RELJMP  040270     ENT    3
RNDIT   045054     LCL    1
RNDSHF  045035     LCL    1
```

RHMOD	040671	LCL	1	
RMOD1	040674	LCL	4	
ROCL-	040023	ENT	2	
ROMCLA	040014	ENT	2	
ROMCLR	040032	LCL	1	
ROMINI	177777	EXT	1	
ROOMOK	042704	LCL	1	
RSTAR	100171	G LAD	4	
RTOIN	045001	ENT	2	
RTOIN1	045004	LCL	1	
RTORTN	045115	LCL	2	
S/DLOP	045515	LCL	0	<--- NOT REFERENCED??
S/DON	045540	LCL	0	<--- NOT REFERENCED??
S/TLOP	046527	LCL	1	
S/TLP1	046551	LCL	1	
SARRAY	041115	LCL	1	
SAVIT	041333	LCL	1	
SAVIT1	041350	LCL	1	
SAVLAV	104164	G LAD	9	
SAVPT2	104154	G LAD	2	
SETOPT	044540	ENT	2	
S..	041407	LCL	0	<--- NOT REFERENCED??
SHORT	044521	ENT	2	
SKIP1	041006	LCL	1	
SKIPL	177777	EXT	1	
SKP2LN	043522	LCL	1	
SKP2NA	043522	ENT	1	
SKP2ST	042332	LCL	1	
SKPANM	177777	EXT	1	
SKPBY2	045765	LCL	2	
SKPCON	177777	EXT	2	
SKPDEF	177777	EXT	1	
SKPLN	177777	EXT	1	
SKPNM+	177777	EXT	12	
SKPNXT	040141	LCL	1	
SKPRS	177777	EXT	1	
SKPVR	041020	ENT	2	
STNUA+	041702	LCL	1	
STNUA1	041551	LCL	0	<--- NOT REFERENCED??
STNUA2	041705	LCL	1	
STNUA3	041606	LCL	1	
STNUA4	041621	LCL	1	
STNUA5	041676	LCL	1	
STNUAR	041543	LCL	1	
STOERX	046752	LCL	1	
STOR( )	041451	LCL	1	
STOR0	041467	LCL	1	
STOR1	042550	ENT	6	
STOR1W	043245	LCL	1	
STOR1X	042562	LCL	2	
STOR1Y	042656	LCL	1	
STOR1Z	043053	LCL	1	
STORI	046456	LCL	0	<--- NOT REFERENCED??
STORR	046451	LCL	0	<--- NOT REFERENCED??

STORS	046463	LCL	0	<--- NOT REFERENCED??
STRTN	041508	LCL	1	
STV	046451	LCL	1	
STOS0	046706	LCL	1	
STOS1	046737	LCL	1	
STOS2	046742	LCL	1	
STOS3	046763	LCL	1	
STOS3+	047011	LCL	1	
STOST	046472	ENT	1	
STOSTA	046657	ENT	2	
STOSTM	046471	ENT	1	
STOSV	046057	ENT	1	
STOSVJ	046116	LCL	1	
STOSVM	046056	ENT	1	
STR1X2	042617	LCL	1	
STRARY	041147	LCL	1	
STREXX	046640	LCL	2	
STREXY	046656	LCL	1	
STRFUN	044346	LCL	1	
STRNG	041134	LCL	1	
STRTBL	041225	LCL	1	
STRTN0	046561	LCL	1	
STRTN1	046567	LCL	1	
STRTN2	046632	LCL	1	
STS3C-	046265	LCL	1	
STSNUM	041501	LCL	1	
STSST1	042126	LCL	1	
STSSTR	042100	LCL	1	
STSTA1	041752	LCL	1	
STSTA2	042005	LCL	2	
STSTA3	042074	LCL	1	
STSTAR	041543	LCL	1	
STSV1	046274	LCL	1	
STSV1R	046116	LCL	1	
STSV3	046120	LCL	1	
STSV3A	046173	LCL	1	
STSV3B	046210	LCL	1	
STSV3C	046226	LCL	1	
STSV3D	046265	LCL	2	
STSV3L	046150	LCL	1	
STSV3X	046143	LCL	1	
STSV3Y	046156	LCL	1	
STSV3Z	046160	LCL	1	
SUBOK?	045533	LCL	1	
SVAL	040506	LCL	1	
SVAL1	040521	LCL	1	
SVRTN	046366	LCL	2	
SVXIT	046340	LCL	1	
TABD	045242	LCL	1	
TBFILL	046542	LCL	1	
TOTEL1	042541	LCL	2	
TOTELM	042521	LCL	0	<--- NOT REFERENCED??
TRAC-	046277	LCL	0	<--- NOT REFERENCED??
TRACE0	046415	LCL	2	

SYMBOL	VALUE	TYPE	COUNT	
TRACE1	046450	LCL	2	
T[illegible]RAY	177777	EXT	1	
TR[illegible]L2	100057	G UAD	1	
TRCFP	177777	EXT	1	
TRCLP	177777	EXT	1	
TRCVBL	177777	EXT	1	
TSTBIG	045063	LCL	4	
TSTDIM	045776	LCL	2	
TSTDUP	040637	LCL	1	
TSTINX	041437	LCL	1	
TSTRM	044650	LCL	2	
TSTRM1	044654	LCL	0	<--- NOT REFERENCED??
TSTRM2	044650	LCL	1	
TSTRMX	044672	LCL	2	
TWOX	177777	EXT	1	
TWOX0	177777	EXT	1	
UNSTAK	177777	EXT	1	
UPDPTR	043373	LCL	3	
VALLOP	045175	LCL	1	
VALOC	040523	LCL	1	
V[illegible]OC1	040550	LCL	1	
VA[illegible]C2	040553	LCL	2	
VALRTN	040466	LCL	2	
VCTOR	045725	LCL	1	
VCTOR1	045755	LCL	1	
VCTOR2	045765	LCL	2	
VLOOP	040707	LCL	1	
VRTN	041155	LCL	1	
VRTN+	041154	LCL	4	
VSR1	040752	LCL	1	
VSR1.5	040772	LCL	1	
VSR2	040776	LCL	2	
VSRCH	040704	ENT	5	
VSRTN	041013	LCL	3	
VSTOR	041164	LCL	1	
XALL	040174	LCL	1	
XALL0	040135	LCL	1	
XALL1	040144	LCL	1	
XALL1A	040174	LCL	1	
XALL2	040175	LCL	1	
X[illegible]N	106550	G UAD	6	
YESFN	041047	LCL	1	
YLPEN	106560	G UAD	3	
ZERO1-	044767	ENT	2	
ZERO10	044756	ENT	1	
ZERO11	044760	ENT	2	
ZEROL2	044742	LCL	0	<--- NOT REFERENCED??
ZEROL3	044746	LCL	1	
ZEROX	044755	LCL	2	
ZRO	000000	G UAD	2	
ZROM-	044731	ENT	1	
ZROM1-	044733	LCL	1	
ZROMEM	044716	ENT	1	
ZRTEST	047064	LCL	1	

CALLO HAD 0 ERRORS 0 WARNINGS 378 LABELS LAST ERROR AT 0