```
000006 (*$U+  COMPILE UP TO COLUMN 72 ONLY*)                                    COMP      2
000006                                                                          COMP      3
000006                                                                          COMP      4
000006 (*************************************************************          COMP      5
000006  *                                                           *          COMP      6
000006  *                                                           *          COMP      7
000006  *            COMPILER FOR PASCAL 6000 - 3.4                 *          COMP      8
000006  *            ******************************                 *          COMP      9
000006  *                                                           *          COMP     10
000006  *                                                           *          COMP     11
000006  *              RELEASE 1  JUNE 1974                         *          COMP     12
000006  *             UPDATE1-10 1/7/74- 1/8/75                     *          UPDAT10   1
000006  *                                                           *          COMP     13
000006  *                                                           *          COMP     14
000006  *          CDC SCIENTIFIC CHAR SET VERSION                  *          COMP     15
000006  *       (00B AND 63B ARE TREATED IDENTICALLY)               *          COMP     16
000006  *                                                           *          COMP     17
000006  *     AUTHOR:  URS AMMANN                                   *          COMP     18
000006  *              INSTITUT FUER INFORMATIK                     *          COMP     19
000006  *              EIDG. TECHNISCHE HOCHSCHULE                  *          COMP     20
000006  *       CH-8006 ZUERICH                                     *          COMP     21
000006  *                                                           *          COMP     22
000006  *************************************************************)         COMP     23
000006                                                                          COMP     24
000006                                                                          COMP     25
000006                                                                          COMP     26
000006                                                                          COMP     27
000006 (*$T-  COMPILE WITHOUT RUNTIME TESTS*)                                   COMP     28
000006 (*$P-  COMPILE WITHOUT PMD INFORMATION*)                                 COMP     29
000006 (*$X5  PASS UP TO FIVE PARAMETERS IN X-REGISTERS*)                       COMP     30
000006                                                                          COMP     31
000006                                                                          COMP     32
000006 PROGRAM PASCAL( INPUT,OUTPUT*,LGO,XREFFIL ) ;                            OVERLAY   1
000464                                                                          COMP     34
000464                                                                          COMP     35
000464 LABEL 13;   (*EXIT IF EOF ENCOUNTERED*)                                  COMP     36
000464                                                                          COMP     37
000464                                                                          COMP     38
000464 CONST DISPLIMIT = 20; MAXLEVEL = 10; MAXADDR = 377777B;                  COMP     39
000464    WORDSIZE = 60;                                                        COMP     40
000464    EFETSZ = 10; CHEFETSZ = 22;                                           COMP     41
000464    CHEFET = 13; BINEFET = 1; (*RELATIVE ADDRESSES OF EFETES*)            COMP     42
000464    RESWORDS = 35;                                                        COMP     43
000464    NRSTDPROC = 23; NRSTDFUNC = 18; NREXFUNC = 6;                         COMP     44
000464    TWOTO17 = 400000B;                                                    COMP     45
000464    CODEMAX = 800; RCODEMAX = 54 (*=CODEMAX DIV 15 + 1*);                 COMP     46
000464    EXTRFMAX = 300; EXTIDMAX = 60;                                        COMP     47
000464    ASSERR = 46B; INXERR = 47B; DIVERR = 50B; STOVLERR = 51B;             COMP     48
000464    HALTERR = 52B; OVLERR = 53B;                                          COMP     49
000464    ALFALENG = 10;                                                        COMP     50
000464                                                                          COMP     51
000464                                                                          COMP     52
000464                                                                          COMP     53
000464 TYPE                          (*DESCRIBING:*)                            COMP     54
000464                               (*************)                           COMP     55
000464                                                                          COMP     56
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000464                                                                  COMP   57
000464                              (*BASIC SYMBOLS*)                   COMP   58
000464                              (***************)                   COMP   59
000464                                                                  COMP   60
000464  SYMBOL = (IDENT,INTCONST,REALCONST,CHARCONST,STRINGCONST,NOTSY,MULOP,  COMP   61
000464       ADDOP,RELOP,LPARENT,RPARENT,LBRACK,RBRACK,COMMA,SEMICOLON,  COMP   62
000464       PERIOD,ARROW,COLON,BECOMES,LABELSY,CONSTSY,TYPESY,VARSY,  COMP   63
000464       FUNCTIONSY,PROCEDURESY,SETSY,PACKEDSY,ARRAYSY,RECORDSY,FILESY,  COMP   64
000464       BEGINSY,IFSY,CASESY,REPEATSY,WHILESY,FORSY,WITHSY,  COMP   65
000464       GOTOSY,ENDSY,ELSESY,UNTILSY,OFSY,DOSY,TOSY,DOWNTOSY,  COMP   66
000464       THENSY,PROGRAMSY,SEGMENTEDSY,OTHERSY);  COMP   67
000464  OPERATOR = (MUL,RDIV,ANDOP,IDIV,IMOD,PLUS,MINUS,OROP,LTOP,LEOP,GEOP,  COMP   68
000464        GTOP,NEOP,EQOP,INOP,NOOP);  COMP   69
000464  SETOFSYS = SET OF SYMBOL;                                       COMP   70
000464                                                                  COMP   71
000464                              (*CONSTANTS*)                       COMP   72
000464                              (***********)                       COMP   73
000464                                                                  COMP   74
000464  CODERANGE = 0..CODEMAX; POSRANGE = 1..4;                        COMP   75
000464  SHRTINT = -377777B..377777B;                                    COMP   76
000464  CSTCLASS = (INT,BOOL,REEL,PSET,STRG);                           COMP   77
000464  CSP = ↑ CSTHEADREC; LOCOFREF = ↑LOCREC;                         COMP   78
000464  CTAILP = ↑ CSTTAILREC;                                          COMP   79
000464  CSTHEADREC = PACKED RECORD NXTCSP: CSP;                         COMP   80
000464                CSTP: CTAILP;                                     COMP   81
000464                CREF: LOCOFREF                                    COMP   82
000464              END;                                                COMP   83
000464  CSTTAILREC = RECORD NXTCSP: CTAILP; CSVAL: INTEGER END;         COMP   84
000464                                                                  COMP   85
000464  LOCREC = PACKED RECORD NXTREF: LOCOFREF;                        COMP   86
000464             LOC: CODERANGE;                                      COMP   87
000464             POS: POSRANGE                                        COMP   88
000464           END;                                                   COMP   89
000464                                                                  COMP   90
000464  VALU = RECORD CASE CKIND: CSTCLASS OF (*CKIND NEVER SET NOR TESTED*)  COMP   91
000464           INT: (IVAL: INTEGER);                                  COMP   92
000464           BOOL: (BVAL: BOOLEAN);                                 COMP   93
000464           REEL: (RVAL: REAL);                                    COMP   94
000464           PSET: (PVAL: SET OF 0..58); (*IMPL. DEPENDANT RANGE*)  COMP   95
000464           STRG: (VALP: CTAILP)                                   COMP   96
000464         END;                                                     COMP   97
000464                                                                  COMP   98
000464                              (*DATA STRUCTURES*)                 COMP   99
000464                              (*****************)                 COMP  100
000464  LEVRANGE = 0..MAXLEVEL; ADDRRANGE = 0..MAXADDR;                 COMP  101
000464  BITRANGE = 0..59 (*=WORDSIZE-1*);                               COMP  102
000464  STRUCTFORM = (SCALAR,SUBRANGE,POINTER,POWER,ARRAYS,RECORDS,FILES,  COMP  103
000464         TAGFIELD,VARIANT);                                       COMP  104
000464  DECLKIND = (STANDARD,DECLARED);                                 COMP  105
000464  WBSIZE = PACKED RECORD WORDS: ADDRRANGE;                        COMP  106
000464           BITS: BITRANGE                                         COMP  107
000464         END;                                                     COMP  108
000464  STP = ↑ STRUCTREC; CTP = ↑ IDENTREC;                            COMP  109
000464                                                                  COMP  110
000464  STRUCTREC = PACKED RECORD                                       COMP  111
000464         FTYPE: BOOLEAN;                                          COMP  112
```

MOORE BUSINESS FORMS, INC.
FORM 1413
PRINTED IN U.S.A.

```
000464            SIZE: WBSIZE;                                                    COMP   113
000464            CASE FORM: STRUCTFORM OF                                         COMP   114
000464             SCALAR:    (CASE SCALKIND: DECLKIND OF                        COMP   115
000464                          DECLARED: (FCONST: CTP));                        COMP   116
000464             SUBRANGE:  (RANGETYPE: STP; MIN,MAX: VALU);                   COMP   117
000464             POINTER:   (ELTYPE: STP);                                     COMP   118
000464             POWER:     (PCKDSET: BOOLEAN; ELSET: STP);                    COMP   119
000464             ARRAYS:    (AELTYPE,INXTYPE: STP;                             COMP   120
000464                         CASE PCKDARR: BOOLEAN OF                          COMP   121
000464                          TRUE: (CASE PARTWORDELS: BOOLEAN OF              COMP   122
000464                                  TRUE: (ELSPERWORD: 2..60)));             COMP   123
000464             RECORDS:   (PCKDREC: BOOLEAN; FIELDS,FSTFLD: CTP;             COMP   124
000464                         RECVAR: STP);                                     COMP   125
000464             FILES:     (PCKDFIL,TEXTFILE,SEGFILE: BOOLEAN;                COMP   126
000464                         FILTYPE: STP);                                    COMP   127
000464             TAGFIELD:  (TAGFIELDP: CTP; FSTVAR: STP);                     COMP   128
000464             VARIANT:   (FSTVARFLD: CTP; NXTVAR,SUBVAR: STP;               COMP   129
000464                         VARVAL: VALU)                                     COMP   130
000464            END;                                                           COMP   131
000464                                                                           COMP   132
000464    EXTIDRANGE = 1..EXTIDMAX;                                              COMP   133
000464    EXTIDREC = PACKED RECORD NAME: ALFA; LINK: 0..EXTRFMAX                 COMP   134
000464         END;                                                              COMP   135
000464                                                                           COMP   136
000464                                (*NAMES*)                                  COMP   137
000464                                (*******)                                  COMP   138
000464                                                                           COMP   139
000464    IDCLASS = (TYPES,KONST,VARS,FIELD,PROC,FUNC);                          COMP   140
000464    SETOFIDS = SET OF IDCLASS;                                             COMP   141
000464    IDKIND = (ACTUAL,FORMAL);                                              COMP   142
000464    ACCESSKIND = (DRCT,INDRCT,INXD);                                       COMP   143
000464    DRCTINDRCT = DRCT..INDRCT;                                             COMP   144
000464                                                                           COMP   145
000464    IDENTREC = PACKED RECORD                                               COMP   146
000464           NAME: ALFA; LLINK: CTP; RLINK: CTP;                             COMP   147
000464           IDTYPE: STP; NEXT: CTP;                                         COMP   148
000464           CASE KLASS: IDCLASS OF                                          COMP   149
000464            KONST: (VALUES: VALU);                                         COMP   150
000464            VARS:  (VKIND: DRCTINDRCT; VLEV: LEVRANGE;                     COMP   151
000464                    VADDR: ADDRRANGE);                                     COMP   152
000464            FIELD: (FLDADDR: ADDRRANGE;                                    COMP   153
000464                    CASE PCKDFLD: BOOLEAN OF                               COMP   154
000464                     TRUE: (BITADDR: BITRANGE));                           COMP   155
000464            PROC,                                                          COMP   156
000464            FUNC:  (CASE PFDECKIND: DECLKIND OF                            COMP   157
000464                    STANDARD: (KEY: 1..25); (*SAME RANGE AS LKEY*)         COMP   158
000464                    DECLARED: (PFLEV: LEVRANGE; PFADDR: ADDRRANGE;         COMP   159
000464                               CASE PFKIND: IDKIND OF                      COMP   160
000464                                ACTUAL: (PFDECL: (DECL,FORWDECL,           COMP   161
000464                                                  EXTDECL,FTNDECL);        COMP   162
000464                                                  PFCNT: 0..7777B)))       COMP   163
000464           END;                                                            COMP   164
000464                                                                           COMP   165
000464    EXTFILEP = ↑ FILEREC;                                                  COMP   166
000464    FILEREC = PACKED RECORD                                                COMP   167
000464                FILENAME: ALFA;                                            COMP   168
```

MOORE BUSINESS FORMS, INC. FORM 113 PRINTED IN U.S.A. 22'

```
000464                 NXTP: EXTFILEP;                                                  COMP     169
000464                 DECLARED,READONLY: BOOLEAN;                                      COMP     170
000464                 SYSLOC: 2..63B                                                   COMP     171
000464               END;                                                               COMP     172
000464                                                                                  COMP     173
000464                                                                                  COMP     174
000464   DISPRANGE = 0..DISPLIMIT;                                                      COMP     175
000464   WHERE = (BLCK,REC);                                                            COMP     176
000464                                                                                  COMP     177
000464                                                                                  COMP     178
000464                                 (*LABELS*)                                       COMP     179
000464                                 (*********)                                      COMP     180
000464   LBP = ^LABREC;                                                                 COMP     181
000464   LABREC = PACKED RECORD                                                         COMP     182
000464              LABVAL: INTEGER; NEXTLAB: LBP;                                      COMP     183
000464              LABLEV: LEVRANGE; LCNT: 0..36;                                      COMP     184
000464              CASE DEFINED: BOOLEAN OF                                            COMP     185
000464               TRUE:  (LABADDR: ADDRRANGE);                                       COMP     186
000464               FALSE: (FSTOCC: LOCOFREF)                                          COMP     187
000464            END;                                                                  COMP     188
000464                                                                                  COMP     189
000464                                                                                  COMP     190
000464 (*-------------------------------------------------------------------*)         COMP     191
000464                                                                                  COMP     192
000464                                                                                  COMP     193
000464 VAR                                                                              COMP     194
000464                   (*RETURNED BY SOURCE PROGRAM SCANNER                           COMP     195
000464                    INSYMBOL:                                                     COMP     196
000464                    **********)                                                   COMP     197
000464                                                                                  COMP     198
000464   SY: SYMBOL;                     (*LAST SYMBOL*)                              COMP     199
000465   OP: OPERATOR;                   (*CLASSIFICATION OF LAST SYMBOL*)            COMP     200
000466   IVAL: INTEGER;                  (*VALUE OF LAST INTEGER CONSTANT*)           COMP     201
000467   RVAL: REAL;                     (*VALUE OF LAST REAL CONSTANT*)              COMP     202
000470   CONSTP: CTAILP;                 (*POINTER TO LAST STRING*)                   COMP     203
000471   LGTH: INTEGER;                  (*LENGTH OF LAST STRING CONSTANT*)           COMP     204
000472   ID: ALFA;                       (*LAST IDENT (POSSIBLY TRUNCATED)*)          COMP     205
000473   KK: 1..ALFALENG;                (*NR OF CHARS IN LAST IDENTIFIER*)           COMP     206
000474   CH: CHAR;                       (*LAST CHARACTER*)                           COMP     207
000475   EOL: BOOLEAN;                   (*END OF LINE CONDITION*)                    COMP     208
000476                                                                                COMP     209
000476                                                                                COMP     210
000476                   (*COUNTERS:*)                                                COMP     211
000476                   (**********)                                                 COMP     212
000476                                                                                COMP     213
000476   CHCNT,MAXCHCNT: INTEGER;        (*CHARACTER COUNTER*)                        COMP     214
000500   LC,IC: INTEGER;                 (*DATA LOCATION AND INSTR CNTER*)            COMP     215
000502   FILECNT,                        (*NUMBER OF LOC FILES IN NESTED USE*)        COMP     216
000502   LABCNT: 0..36;                  (*NUMBER OF EXTERNAL LABELS*)                COMP     217
000504   EXTFILS: 0..50;                 (*NUMBER OF EXTERNAL FILES*)                 COMP     218
000505   BUFFAC: INTEGER;                (*FACTOR TO COMPUTE FILE BUFFER SIZE*)       COMP     219
000506   XPARMAX: INTEGER;               (*NR OF PARAMS TO BE PASSED IN X-RG*)        COMP     220
000507   PCNT: INTEGER;                  (*NUMBER OF PROCEDURES/FUNCTIONS*)           COMP     221
000510   CTIME : INTEGER ;               (*CPU TIME USED*)                            CTIME      1
000511   B6DPL: ADDRRANGE;                                                            COMP     222
000512                                                                                CROSSREF   2
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A.

```
000512                                                                  CROSSREF    3
000512                  (*CROSS REFERENCE VARIABLES:*)                  CROSSREF    4
000512                  (****************************)                  CROSSREF    5
000512  (*$B1*)                                                         CROSSREF    6
000512  XREFFIL : SEGMENTED FILE OF PACKED RECORD                       CROSSREF    7
000512   LNO : 0..99 ;  PNO : 0..999 ;                                  CROSSREF    8
000512   CASE ERRORREF : BOOLEAN OF                                     CROSSREF    9
000512   FALSE:( AVAL : ALFA ) ;                                        CROSSREF   10
000512   TRUE:( ENO : 0..400 ) ;                                        CROSSREF   11
000512  END ;                                                           CROSSREF   12
000726                                                                  LISTCTRL    1
000726                                                                  LISTCTRL    2
000726                   (*LIST CONTROL VARIABLES:*)                    LISTCTRL    3
000726                   (*************************)                    LISTCTRL    4
000726                                                                  LISTCTRL    5
000726  LINEBUFFER : ARRAY[ 1..130 ] OF CHAR ;  (*HOLDS CURRENT LINE*)  LISTCTRL    6
001130  LINEADDR : INTEGER ;             (*ADDRESS AT START OF LINE*)   LISTCTRL    7
001131  LINECOUNT : INTEGER ;            (*COUNT OF LINES ON CURRENT PAGE*)  LISTCTRL    8
001132  PAGECOUNT : INTEGER ;            (*NUMBER OF CURRENT PAGE OF LISTING*)  LISTCTRL    9
001133  PAGEEJECT : BOOLEAN ;            (*EJECT FLAG*)                 LISTCTRL   10
001134  LDATE,LTIME : ALFA ;             (*DATE AND TIME AT START OF COMPILE*)  LISTCTRL   11
001136  LISTNAME : ALFA ;                (*NAME OF PROCEDURE BEING COMPILED*)  LISTCTRL   12
001137  LSTMAP : PACKED ARRAY[ 1..26 ] OF CHAR ;  (*MAPS COMP OP TO DIGIT*)  LISTCTRL   13
001142  LSTSTR : PACKED ARRAY[ 1..26 ] OF CHAR ;  (*OPTION STRING FOR PRINT*)  LISTCTRL   14
001145  EOP : BOOLEAN ;                  (*END OF PROGRAM FLAG*)        LISTCTRL   15
001146                                                                  COMP      223
001146                                                                  COMP      224
001146                  (*SWITCHES:*)                                   COMP      225
001146                  (***********)                                   COMP      226
001146                                                                  COMP      227
001146  RUNLINKAGE,                      (*USE RUN-FORTRAN CALLING SEQUENCE*)  RUNMODS     1
001146  DP,                              (*DECLARATION PART*)           COMP      228
001146  PRTERR: BOOLEAN;                 (*TO ALLOW FORWARD REFERENCES  COMP      229
001151                                   BY SUPPRESSING ERROR MESSAGE*)  COMP      230
001151  DEBUG,EXTON,LISTON,PMD: BOOLEAN;                                COMP      231
001155                                                                  COMP      232
001155                                                                  COMP      233
001155                  (*POINTERS:*)                                   COMP      234
001155                  (***********)                                   COMP      235
001155  INTPTR,REALPTR,CHARPTR,ALFAPTR,STEXTPTR,                        UPDATE9     2
001155  BOOLPTR,NILPTR,TEXTPTR: STP;     (*POINTERS TO ENTRIES OF STD IDS*)  COMP      237
001165  UTYPPTR,UCSTPTR,UVARPTR,                                        COMP      238
001165  UFLDPTR,UPRCPTR,UFCTPTR,         (*POINTERS TO ENTRIES FOR UNDECL IDS*)  COMP      239
001165  INPUTPTR,OUTPUTPTR,              (*ENTRIES FOR INPUT AND OUTPUT*)  COMP      240
001165  FWPTR: CTP;                      (*HEAD OF CHAIN OF FORW  TYPE IDS*)  COMP      241
001176  FSTLABP: LBP;                    (*HEAD OF LABEL CHAIN*)        COMP      242
001177  FEXFILP: EXTFILEP;               (*HEAD OF LIST OF EXTERNAL FILES*)  COMP      243
001200  FSTCSP: CSP;                     (*HEAD OF CONSTANT CHAIN*)     COMP      244
001201                                                                  COMP      245
001201                                                                  COMP      246
001201                  (*BOOKKEEPING OF DECLARATION LEVELS:*)          COMP      247
001201                  (************************************)          COMP      248
001201                                                                  COMP      249
001201  LEVEL: LEVRANGE;                 (*CURRENT STATIC LEVEL*)       COMP      250
001202  DISX,                            (*LEVEL OF LAST ID SRCHD BY SEARCHID*)  COMP      251
001202  TOP: DISPRANGE;                  (*TOP OF DISPLAY*)             COMP      252
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
001204
001204  DISPLAY:                          (*WHERE:   MEANS:*)
001204   ARRAY [DISPRANGE] OF
001204    PACKED RECORD                  (*=BLCK:   ID IS VARIABLE ID*)
001204     FNAME: CTP;                   (*=REC:    ID IS FIELD ID IN RECORD*)
001204     CASE OCCUR: WHERE OF
001204      REC: (WACC: DRCTINDRCT;
001204         LEV: LEVRANGE; CWDSPL: ADDRRANGE;
001204         CASE PKD: BOOLEAN OF
001204           TRUE: (BACC: DRCTINDRCT; BDSPL: SHRTINT))
001204     END;
001231
001231
001231                  (*ERROR MESSAGES:*)
001231                  (**********)
001231
001231  ERRINX: 0..10;                    (*NR OF ERRORS IN CURR SOURCE LINE*)
001232  ERRORS: BOOLEAN;
001233  ERRLIST:
001233   ARRAY [1..10] OF
001233    PACKED RECORD POS: 1..1000000;
001233           NMR: 1..400
001233       END;
001245
001245                  (*CODEFILE AND TABLES FOR EXT. REFERENCES*)
001245                  (**********)
001245
001245  (*$B1  WILL NOT ALLOW LGO TAPES*)
001245  LGO: SEGMENTED FILE OF INTEGER;
001460  PROGNAME: ALFA;
001461  EXT,EXT1,EXTIDX: 0..EXTIDMAX;
001464  EXTID: ARRAY[1..EXTIDMAX] OF EXTIDREC;   (*NAMES OF EXTERNALS*)
001654  EXTRX: 0..EXTRFMAX;
001655  EXTRFS: ARRAY[1..EXTRFMAX] OF
001655       PACKED RECORD LOC: 0..77777777778;
001655              LINK: 0..EXTRFMAX
001655       END;
002331  CADDR,                            (*ADDRESS FOR NEXT TEXTTABLE*)
002331  CODEADDR: ADDRRANGE;              (*ADDRESS OF
002333                                     CURRENT CODE SEGMENT*)
002333
002333
002333
002333
002333
002333
002333                  (*STRUCTURED CONSTANTS:*)
002333                  (**********)
002333
002333  LETTERS: SET OF ΞAΞ..ΞZΞ; DIGITS: SET OF ΞOΞ..Ξ9Ξ;
002335  CONSTBEGSYS,SIMPTYPEBEGSYS,TYPEBEGSYS,BLOCKBEGSYS,SELECTSYS,FACBEGSYS,
002335  STATBEGSYS,TYPEDELS: SETOFSYS;
002345  RW: ARRAY [1..RESWORDS] OF ALFA;
002410  LRW: ARRAY [0..ALFALENG] OF 0..RESWORDS;
002423  RSY: ARRAY [1..RESWORDS] OF SYMBOL;
002466  SSY: ARRAY [Ξ+Ξ..Ξ;Ξ] OF SYMBOL;
```

```
COMP     253
COMP     254
COMP     255
COMP     256
COMP     257
COMP     258
COMP     259
COMP     260
COMP     261
COMP     262
COMP     263
COMP     264
COMP     265
COMP     266
COMP     267
COMP     268
COMP     269
COMP     270
COMP     271
COMP     272
COMP     273
COMP     274
COMP     275
COMP     276
COMP     277
COMP     278
COMP     279
COMP     280
COMP     281
COMP     282
COMP     283
RUNMODS    2
COMP     285
COMP     286
COMP     287
COMP     288
COMP     289
COMP     290
COMP     291
COMP     292
COMP     293
COMP     294
COMP     295
COMP     296
COMP     297
COMP     298
COMP     299
COMP     300
COMP     301
COMP     302
COMP     303
COMP     304
COMP     305
COMP     306
COMP     307
COMP     308
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A.

```
002521   ROP: ARRAY [1..RESWORDS] OF OPERATOR;                                  COMP     309
002564   SOP: ARRAY [=+=..=;=] OF OPERATOR;                                     COMP     310
002617   NA:  ARRAY [1..51] OF ALFA;                                            COMP     311
002702   PASCF,PASCL,PNAME: ALFA;                                               COMP     312
002705                                                                          COMP     313
002705                                                                          COMP     314
002705                                                                          COMP     315
002705                                                                          COMP     316
002705 (*-------------------------------------------------------------------*)  COMP     317
002705                                                                          COMP     318
002705                                                                          COMP     319
000003 PROCEDURE WRITEERRORS;                                                   COMP     320
000003  VAR LASTPOS,FREEPOS,CURRPOS,CURRNMR,F,K: INTEGER;                       COMP     321
000011 BEGIN WRITE(= ******=:7);                                                COMP     322
000012  LASTPOS := 0; FREEPOS := 1;                                             COMP     323
000014  FOR K := 1 TO ERRINX DO                                                 COMP     324
000015   BEGIN                                                                  COMP     325
000017    WITH ERRLIST[K] DO                                                    COMP     326
000020     BEGIN CURRPOS := POS; CURRNMR := NMR END;                            COMP     327
000023    IF CURRPOS = LASTPOS THEN WRITE(=,=)                                  COMP     328
000026    ELSE                                                                  COMP     329
000027     BEGIN                                                                COMP     330
000027      WHILE FREEPOS < CURRPOS DO                                          COMP     331
000031       BEGIN WRITE(= =); FREEPOS := FREEPOS + 1 END;                      COMP     332
000035      WRITE(=^=);                                                         COMP     333
000037       LASTPOS := CURRPOS                                                 COMP     334
000037     END;                                                                 COMP     335
000041    IF CURRNMR < 10 THEN F := 1                                           COMP     336
000043    ELSE IF CURRNMR < 100 THEN F := 2                                     COMP     337
000046     ELSE F := 3;                                                         COMP     338
000050    WRITE(CURRNMR:F);                                                     COMP     339
000054    FREEPOS := FREEPOS + F + 1                                            COMP     340
000056   END;                                                                   COMP     341
000061  WRITELN; ERRINX := 0                                                    COMP     342
000062 END (*WRITEERRORS*) ;                                                    COMP     343
000070                                                                          COMP     344
000003 PROCEDURE ENDOFLINE;                                                     COMP     345
000003  CONST  LINESPERPAGE = 56 ;                                              LISTCTRL  16
000003  VAR I : INTEGER ;                                                       LISTCTRL  17
000004 BEGIN                                                                    LISTCTRL  18
000004  IF ( LISTON ) OR ( ERRINX>0 ) THEN                                      LISTCTRL  19
000010   BEGIN                                                                  LISTCTRL  20
000010    IF ( PAGEEJECT ) OR ( LINECOUNT >= LINESPERPAGE ) THEN                LISTCTRL  21
000013     BEGIN PAGEEJECT := FALSE; LINECOUNT := 0; PAGECOUNT:=PAGECOUNT+1;    LISTCTRL  22
000015      WRITE( =1= ) ;                                                      LISTCTRL  23
000017      WRITE( LISTNAME ) ;  (* PROCEDURE NAME *)                           LISTCTRL  24
000024      WRITE( = =,LSTSTR ) ;  (*COMPILES OPTIONS STRING*)                  LISTCTRL  25
000033      WRITE( =PASCAL 6000 - 3.4   = ) ;  (*PROCESSOR NAME*)               LISTCTRL  26
000040      WRITE( LDATE,LTIME ) ;  (*LISTING DATE AND TIME*)                   LISTCTRL  27
000052      WRITE( =   PAGE=,PAGECOUNT : 9 ) ;                                  LISTCTRL  28
000063      WRITELN ;  WRITELN ;  (*END HEADING, SKIP LINE*)                    LISTCTRL  29
000065     END ;                                                                LISTCTRL  30
000065    WRITE( = =,LINEADDR:6 OCT,= = ) ;                                     LISTCTRL  31
000075    FOR I := 1 TO CHCNT DO WRITE( LINEBUFFER[ I ] ) ;                     LISTCTRL  32
000105    WRITELN ;  LINECOUNT := LINECOUNT+1 ;                                 LISTCTRL  33
000110    IF ERRINX>0 THEN                                                      LISTCTRL  34
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000111       BEGIN  WRITEERRORS ;  LINECOUNT := LINECOUNT+1 ; END ;                LISTCTRL   35
000114     END ;                                                                   LISTCTRL   36
000114    IF DP THEN LINEADDR := LC ELSE LINEADDR := IC ;                          LISTCTRL   37
000121    IF ( EOF( INPUT ) ) AND ( NOT EOP ) THEN                                 LISTCTRL   38
000124     BEGIN WRITELN( '****** EOF ENCOUNTERED' ) ; ERRORS := TRUE ;            LISTCTRL   39
000133      GOTO 13                                                               COMP      355
000134     END;                                                                    COMP      356
000134    CHCNT := 0                                                              COMP      357
000134  END  (*ENDOFLINE*) ;                                                      COMP      358
000147                                                                            COMP      359
000003  PROCEDURE ERROR(FERRNR: INTEGER);                                         COMP      360
000004  BEGIN ERRORS := TRUE;                                                     COMP      361
000010   IF ERRINX >= 9 THEN                                                      COMP      362
000012    BEGIN ERRLIST[10].NMR := 255; ERRINX := 10 END                          COMP      363
000015   ELSE                                                                     COMP      364
000016    BEGIN ERRINX := ERRINX + 1;                                             COMP      365
000017     ERRLIST[ERRINX].NMR := FERRNR                                          COMP      366
000017    END;                                                                    COMP      367
000022   WITH XREFFIL^ DO                                                         CROSSREF   13
000024    BEGIN  LNO := LINECOUNT ;  PNO := PAGECOUNT ;                           CROSSREF   14
000030     ERRORREF := TRUE ;  END := ERRLIST[ ERRINX ].NMR ;                     CROSSREF   15
000036    END ;                                                                   CROSSREF   16
000036   PUT( XREFFIL ) ;                                                         CROSSREF   17
000037   ERRLIST[ERRINX].POS := CHCNT                                             COMP      368
000040  END (*ERROR*) ;                                                           COMP      369
000047                                                                            COMP      370
000003  PROCEDURE ORDALFA(FNAME: ALFA; VAR FI: INTEGER);                          COMP      371
000005     VAR  SWAP : RECORD CASE INTEGER OF                                     ORDALFA     1
000005               1:( A : ALFA ) ;                                             ORDALFA     2
000005               2:( K : INTEGER ) ;                                          ORDALFA     3
000005          END ;                                                             ORDALFA     4
000006  BEGIN                                                                     COMP      373
000006      SWAP.A := FNAME ;  FI := SWAP.K                                       ORDALFA     5
000010  END (*ORDALFA*) ;                                                         COMP      378
000014                                                                            COMP      379
000003  PROCEDURE INSYMBOL;                                                       COMP      380
000003   (*READ NEXT BASIC SYMBOL OF SOURCE PROGRAM AND RETURN ITS DESCRIPTION    COMP      381
000003   IN THE GLOBAL VARIABLES SY, OP, ID, IVAL, RVAL, SVAL AND LGTH*)          COMP      382
000003   LABEL 1,2;                                                               COMP      383
000003   CONST DIGMAX = 20; MAX8 = 7777777777777777778;                           COMP      384
000003     HEXDIGMAX = 15 ;                                                       HEXCON      1
000003      MAX10 = 31463146314631463138;                                         COMP      385
000003   VAR I,K,SCALE,EXP: INTEGER;                                              COMP      386
000007     R,FAC: REAL; SIGN: BOOLEAN;                                            COMP      387
000012     DIGIT: ARRAY [1..DIGMAX] OF 0..9;                                      COMP      388
000036     APO,STRINGEND: BOOLEAN; NXTP,TAILP: CTAILP;                            COMP      389
000042                                                                            COMP      390
000003   PROCEDURE NEXTCH;                                                        COMP      391
000003   BEGIN                                                                    COMP      392
000003    REPEAT EOL := EOLN(INPUT);                                              COMP      393
000010     READ( CH ) ;  CHCNT := CHCNT+1 ;                                       LISTCTRL   40
000014     LINEBUFFER[ CHCNT ] := CH ;                                            LISTCTRL   41
000015    UNTIL EOL OR (CHCNT <= MAXCHCNT)                                        COMP      396
000016   END (*NEXTCH*) ;                                                        COMP      397
000023                                                                            COMP      398
000023                                                                            COMP      399
```

MOORE BUSINESS FORMS, INC. FORM 121 PRINTED IN U.S.A.

```
000003   PROCEDURE OPTIONS;                                                    COMP       400
000003    VAR ENDOPTIONS: BOOLEAN; CH1: CHAR;                                  COMP       401
000005    BEGIN ENDOPTIONS := FALSE;                                           COMP       402
000007    REPEAT NEXTCH;                                                       COMP       403
000011      IF CH IN [ΞBΞ,ΞEΞ,ΞLΞ,ΞPΞ,ΞRΞ,ΞTΞ,ΞUΞ,ΞXΞ] THEN                    RUNMODS      3
000014      BEGIN CH1 := CH; NEXTCH;                                           COMP       405
000016       CASE CH1 OF                                                       COMP       406
000020        ΞBΞ: IF CH IN [Ξ1Ξ..Ξ9Ξ] THEN BUFFAC := ORD(CH) - ORD(Ξ0Ξ)      COMP       407
000023             ELSE ENDOPTIONS := TRUE;                                    COMP       408
000027        ΞEΞ: IF CH IN [Ξ+Ξ,Ξ-Ξ] THEN EXTON := CH = Ξ+Ξ                   COMP       409
000032             ELSE ENDOPTIONS := TRUE;                                    COMP       410
000037        ΞLΞ: IF CH IN [ Ξ+Ξ,Ξ-Ξ,Ξ*Ξ ] THEN                           LISTCTRL    42
000042              BEGIN  PAGEEJECT := CH=Ξ*Ξ ;                               LISTCTRL    43
000044               LISTON := ( PAGEEJECT ) OR ( CH = Ξ+Ξ ) ;  END            LISTCTRL    44
000050             ELSE ENDOPTIONS := TRUE;                                    COMP       415
000052        ΞPΞ: IF CH IN [Ξ+Ξ,Ξ-Ξ] THEN PMD := CH = Ξ+Ξ                     COMP       416
000055             ELSE ENDOPTIONS := TRUE;                                    COMP       417
000062        ΞRΞ: IF CH IN [Ξ+Ξ,Ξ-Ξ] THEN RUNLINKAGE := CH = Ξ+Ξ              RUNMODS      4
000065              ELSE ENDOPTIONS := TRUE;                                   RUNMODS      5
000072        ΞTΞ: IF CH IN [Ξ+Ξ,Ξ-Ξ] THEN DEBUG := CH = Ξ+Ξ                   COMP       418
000075             ELSE ENDOPTIONS := TRUE;                                    COMP       419
000102        ΞUΞ: IF CH IN [Ξ+Ξ,Ξ-Ξ] THEN                                    COMP       420
000105              IF CH = Ξ+Ξ THEN MAXCHCNT := 72 ELSE MAXCHCNT := 80        LISTCTRL    45
000110             ELSE ENDOPTIONS := TRUE;                                    COMP       422
000114        ΞXΞ: IF CH IN [Ξ0Ξ..Ξ6Ξ] THEN XPARMAX := ORD(CH) - ORD(Ξ0Ξ)      COMP       423
000117             ELSE ENDOPTIONS := TRUE;                                    COMP       424
000123       END;                                                              COMP       425
000152        IF ( NOT ENDOPTIONS ) AND ( CH1 <> ΞLΞ ) THEN                    LISTCTRL    46
000156         LSTSTR[ 5+3*( ORD( LSTMAP[ ORD( CH1 ) ] )-ORD( Ξ0Ξ ) ) ] := CH ;  LISTCTRL    47
000176        IF NOT ENDOPTIONS THEN                                           COMP       426
000200         BEGIN NEXTCH; ENDOPTIONS := CH <> Ξ,Ξ END                       COMP       427
000204       END                                                               COMP       428
000204      ELSE ENDOPTIONS := TRUE                                            COMP       429
000205     UNTIL ENDOPTIONS;                                                   COMP       430
000207    END (*OPTIONS*);                                                     COMP       431
000220                                                                         COMP       432
000220   BEGIN (*INSYMBOL*)                                                    COMP       433
000220   1:                                                                    COMP       434
000006    (*LOOP UNTIL EOL:*)                                                  COMP       435
000006    WHILE (CH = Ξ Ξ)AND NOT EOL DO NEXTCH;                               COMP       436
000014    WHILE EOL DO                                                         COMP       437
000015     BEGIN ENDOFLINE; NEXTCH;                                            COMP       438
000020      WHILE (CH = Ξ Ξ)AND NOT EOL DO NEXTCH                              COMP       439
000024     END;                                                                COMP       440
000027    CASE CH OF                                                           COMP       441
000030     ΞAΞ,ΞBΞ,ΞCΞ,ΞDΞ,ΞEΞ,ΞFΞ,ΞGΞ,ΞHΞ,ΞIΞ,                               COMP       442
000030     ΞJΞ,ΞKΞ,ΞLΞ,ΞMΞ,ΞNΞ,ΞOΞ,ΞPΞ,ΞQΞ,ΞRΞ,                               COMP       443
000030     ΞSΞ,ΞTΞ,ΞUΞ,ΞVΞ,ΞWΞ,ΞXΞ,ΞYΞ,ΞZΞ:                                   COMP       444
000030      BEGIN K := 0;                                                      COMP       445
000031       REPEAT                                                           COMP       446
000031        IF K < ALFALENG THEN                                            COMP       447
000033         BEGIN K := K + 1; ID[K] := CH END;                             COMP       448
000040       NEXTCH                                                           COMP       449
000040      UNTIL NOT (CH IN LETTERS+DIGITS);                                 COMP       450
000045      IF K >= KK THEN KK := K                                           COMP       451
000047      ELSE                                                              COMP       452
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A.

```
000050        REPEAT ID[KK] := Ξ Ξ; KK := KK - 1;                                  COMP       453
000056        UNTIL KK = K;                                                        COMP       454
000060       FOR I := LRW[K-1] + 1 TO LRW[K] DO                                    COMP       455
000064        IF RW[I] = ID THEN                                                   COMP       456
000070         BEGIN SY := RSY[I]; OP := ROP[I]; GOTO 2 END;                       COMP       457
000075        IF LISTON THEN                                                       CROSSREF    18
000076         BEGIN                                                               CROSSREF    19
000076          WITH XREFFIL↑ DO                                                   CROSSREF    20
000100           BEGIN  AVAL := ID ;  LNO := LINECOUNT ;  PNO := PAGECOUNT ;       CROSSREF    21
000105            ERRORREF := FALSE ;                                              CROSSREF    22
000107           END ;                                                             CROSSREF    23
000107          PUT( XREFFIL ) ;                                                   CROSSREF    24
000110         END ;                                                               CROSSREF    25
000110        SY := IDENT; OP := NOOP;                                             COMP       458
000112  2:     END;                                                                COMP       459
000113    Ξ0Ξ,Ξ1Ξ,Ξ2Ξ,Ξ3Ξ,Ξ4Ξ,Ξ5Ξ,Ξ6Ξ,Ξ7Ξ,Ξ8Ξ,Ξ9Ξ:                                          COMP       460
000113     BEGIN SY := INTCONST; OP := NOOP;                                       COMP       461
000115      I := 0;                                                                COMP       462
000116      REPEAT I := I + 1;                                                     COMP       463
000120       IF I <= DIGMAX THEN DIGIT[I] := ORD(CH) - ORD(Ξ0Ξ)                    COMP       464
000122       ELSE                                                                  COMP       465
000123        BEGIN ERROR(203); I := 1 END;                                        COMP       466
000126       NEXTCH                                                                COMP       467
000126      UNTIL NOT (CH IN DIGITS);                                              COMP       468
000133      IVAL := 0;                                                             COMP       469
000134      IF CH = Ξ8Ξ THEN                                                       COMP       470
000135       BEGIN                                                                 COMP       471
000135        FOR K := 1 TO I DO                                                   COMP       472
000137         IF IVAL <= MAX8 THEN                                                COMP       473
000143          BEGIN IF DIGIT[K] IN [8,9] THEN ERROR(204);                        COMP       474
000147           IVAL := 8*IVAL + DIGIT[K]                                         COMP       475
000151          END                                                                COMP       476
000152         ELSE BEGIN ERROR(203); IVAL := 0 END;                               COMP       477
000161        NEXTCH                                                               COMP       478
000161       END                                                                   COMP       479
000163      ELSE                                                                   COMP       480
000164       BEGIN                                                                 COMP       481
000164        FOR K := 1 TO I DO                                                   COMP       482
000166         IF IVAL <= MAX10 THEN                                               COMP       483
000172          IVAL := 10*IVAL + DIGIT[K]                                         COMP       484
000173        ELSE BEGIN ERROR(203); IVAL := 0 END;                                COMP       485
000203       SCALE := 0;                                                           COMP       486
000204       IF CH = Ξ.Ξ THEN                                                      COMP       487
000206        BEGIN NEXTCH;                                                        COMP       488
000207         IF CH = Ξ.Ξ THEN CH := Ξ:Ξ                                          COMP       489
000211         ELSE                                                                COMP       490
000212          BEGIN RVAL := IVAL; SY := REALCONST;                               COMP       491
000214           IF NOT (CH IN DIGITS) THEN ERROR(201)                             COMP       492
000217           ELSE                                                              COMP       493
000221            REPEAT RVAL := 10.0*RVAL + (ORD(CH) - ORD(Ξ0Ξ));                 COMP       494
000225             SCALE := SCALE - 1; NEXTCH                                      COMP       495
000227            UNTIL NOT (CH IN DIGITS)                                         COMP       496
000232          END                                                                COMP       497
000233        END;                                                                 COMP       498
000233       IF CH = ΞEΞ THEN                                                      COMP       499
000235        BEGIN                                                                COMP       500
```

MOORE BUSINESS FORMS, INC. F
FORM 1413
PRINTED IN U.S.A.
22

```
000235          IF SCALE = 0 THEN                                                   COMP       501
000237           BEGIN RVAL :=  IVAL; SY :=  REALCONST END;                         COMP       502
000241          SIGN :=  FALSE; NEXTCH;                                             COMP       503
000243          IF CH = '+' THEN NEXTCH                                             COMP       504
000245          ELSE                                                                COMP       505
000247           IF CH = '-' THEN                                                   COMP       506
000251            BEGIN SIGN :=  TRUE; NEXTCH END;                                  COMP       507
000253          EXP :=  0;                                                          COMP       508
000254          IF NOT (CH IN DIGITS) THEN ERROR(201)                               COMP       509
000257          ELSE                                                                COMP       510
000261           REPEAT EXP :=  10*EXP + (ORD(CH) - ORD('0'));                      COMP       511
000265            NEXTCH                                                            COMP       512
000265           UNTIL NOT (CH IN DIGITS);                                          COMP       513
000271          IF SIGN THEN SCALE :=  SCALE - EXP                                  COMP       514
000272           ELSE SCALE :=  SCALE + EXP                                         COMP       515
000274         END;                                                                 COMP       516
000276        IF SCALE <> 0 THEN                                                    COMP       517
000300         BEGIN R :=  1.0;   (*NOTE POSSIBLE OVERFLOW+UNDERFLOW*)              COMP       518
000301          SIGN :=  SCALE < 0; SCALE :=  ABS( SCALE ); FAC :=  10.0;          UPDATE3      2
000305          REPEAT IF ODD(SCALE) THEN R :=  R*FAC;                            COMP       522
000311           FAC :=  SQR(FAC); SCALE :=  SCALE DIV 2                          COMP       523
000312          UNTIL SCALE = 0;   (*NOW R = 10↑SCALE*)                           COMP       524
000315          IF SIGN THEN RVAL :=  RVAL/R ELSE RVAL :=  RVAL*R                 UPDATE3      3
000320         END;                                                               COMP       526
000322       END                                                                   COMP       527
000322     END;                                                                    COMP       528
000323     '$'                                                                     HEXCON       2
000323      BEGIN  SY :=  INTCONST ;  OP :=  NOOP ;  I :=  0 ;  NEXTCH ;           HEXCON       3
000327       WHILE ( CH IN ( DIGITS+[ 'A'..'F' ] ) ) DO                            HEXCON       4
000332        BEGIN  I :=  I+1 ;                                                   HEXCON       5
000334         IF I <= HEXDIGMAX THEN                                             HEXCON       6
000335          IF ( CH IN DIGITS ) THEN DIGIT[ I ] :=  ORD( CH )-ORD( '0' )      HEXCON       7
000337                               ELSE DIGIT[ I ] :=  ORD( CH )+9               HEXCON       8
000340         ELSE                                                               HEXCON       9
000343          BEGIN  ERROR( 203 ) ;  I :=  1 ;  END ;                           HEXCON      10
000346         NEXTCH ;                                                           HEXCON      11
000350        END ;                                                               HEXCON      12
000351       IF I = 0 THEN ERROR( 203 ) ;                                         HEXCON      13
000354       IVAL :=  0 ;                                                         HEXCON      14
000355       FOR K :=  1 TO I DO                                                  HEXCON      15
000357        IF IVAL <= MAX8 THEN IVAL :=  IVAL*16+DIGIT[ K ]                    HEXCON      16
000363        ELSE                                                                HEXCON      17
000365         BEGIN  ERROR( 203 ) ;  IVAL :=  0 ;  END ;                         HEXCON      18
000373      END ;                                                                 HEXCON      19
000374     ''''                                                                    COMP       529
000374         :                                                                   COMP       530
000374      BEGIN OP :=  NOOP; APO :=  FALSE; STRINGEND :=  FALSE;                 COMP       531
000377       LGTH :=  0; I :=  0; IVAL :=  0; CONSTP :=  NIL;                      COMP       532
000402       NEXTCH;                                                              COMP       533
000403       REPEAT                                                               COMP       534
000403        IF EOL THEN                                                         COMP       535
000404         BEGIN ERROR(202); STRINGEND :=  TRUE END                           COMP       536
000407        ELSE                                                                COMP       537
000410         IF (CH <> '''')OR APO THEN                                         COMP       538
000414          BEGIN                                                             COMP       539
000414           IF I = ALFALENG THEN                                             COMP       540
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A.

```
000416          BEGIN NEW(TAILP);                                                  COMP      541
000417           WITH TAILP↑ DO                                                    COMP      542
000421            BEGIN NXTCSP ≠= CONSTP; CSVAL ≠= IVAL END;                       COMP      543
000423           CONSTP ≠= TAILP; I ≠= 0; IVAL ≠= 0                                COMP      544
000425          END;                                                               COMP      545
000425         I ≠= I + 1; LGTH ≠= LGTH + 1; APO ≠= FALSE;                         COMP      546
000430         IVAL ≠= 64*IVAL + ORD(CH);                                          COMP      547
000432         NEXTCH                                                              COMP      548
000432        END                                                                  COMP      549
000434       ELSE                                                                  COMP      550
000435         BEGIN APO ≠= TRUE;                                                  COMP      551
000436         NEXTCH; STRINGEND ≠=                                                COMP      552
000437                                 CH <> ΞΞΞΞ                                  COMP      553
000437          END                                                                COMP      554
000442      UNTIL STRINGEND;                                                      COMP      555
000443      IF LGTH = 1 THEN                                                      COMP      556
000445       SY ≠= CHARCONST                                                      COMP      557
000445      ELSE                                                                  COMP      558
000446       BEGIN FOR I ≠= I + 1 TO ALFALENG DO IVAL ≠= 64*IVAL;                 COMP      559
000455        NEW(TAILP);                                                         COMP      560
000457        WITH TAILP↑ DO                                                      COMP      561
000461         BEGIN NXTCSP ≠= CONSTP; CSVAL ≠= IVAL END;                        COMP      562
000463        (*REVERSE POINTERS≠*)                                               COMP      563
000463        CONSTP ≠= NIL;                                                      COMP      564
000464        WHILE TAILP <> NIL DO                                               COMP      565
000466         WITH TAILP↑ DO                                                     COMP      566
000467          BEGIN NXTP ≠= NXTCSP; NXTCSP ≠= CONSTP;                           COMP      567
000471           CONSTP ≠= TAILP; TAILP ≠= NXTP                                   COMP      568
000472          END;                                                              COMP      569
000473           SY ≠= STRINGCONST                                                COMP      570
000473          END                                                               COMP      571
000474       END;                                                                 COMP      572
000475      COL,PER                                                               PERCENTB    1
000475           ≠                                                                COMP      574
000475       BEGIN OP ≠= NOOP; NEXTCH;                                            COMP      575
000500        IF CH = Ξ=Ξ THEN                                                    COMP      576
000502         BEGIN SY ≠= BECOMES; NEXTCH END                                    COMP      577
000505        ELSE SY ≠= COLON                                                    COMP      578
000506       END;                                                                 COMP      579
000510      Ξ.Ξ≠                                                                  COMP      580
000510       BEGIN OP ≠= NOOP; NEXTCH;                                           COMP      581
000513        IF CH = Ξ.Ξ THEN                                                   COMP      582
000515         BEGIN SY ≠= COLON; NEXTCH END                                     COMP      583
000520        ELSE SY ≠= PERIOD                                                   COMP      584
000521       END;                                                                 COMP      585
000523      Ξ↑Ξ≠                                                                  COMP      586
000523       BEGIN NEXTCH;                                                       COMP      587
000525        IF CH = Ξ$Ξ THEN OPTIONS;                                          COMP      588
000530        IF EOL THEN ENDOFLINE;                                             COMP      589
000532        WHILE CH <> Ξ↓Ξ DO                                                 COMP      590
000534         BEGIN NEXTCH;                                                     COMP      591
000535          IF EOL THEN ENDOFLINE                                            COMP      592
000536         END;                                                               COMP      593
000540        NEXTCH; GOTO 1                                                      COMP      594
000543       END;                                                                 COMP      595
000543      Ξ(Ξ≠                                                                  COMP      596
```

MOORE BUSINESS FORMS, INC., F
FORM 141
PRINTED IN U.S.A.
22

```
000543      BEGIN NEXTCH;                                                    COMP     597
000545       IF CH = Ξ*Ξ THEN                                                COMP     598
000547        BEGIN NEXTCH;                                                  COMP     599
000550         IF CH = Ξ$Ξ THEN OPTIONS;                                     COMP     600
000553         REPEAT                                                        COMP     601
000553          (*LOOP UNTIL CH = Ξ*Ξ#*)                                     COMP     602
000553          IF EOL THEN ENDOFLINE;                                       COMP     603
000555          WHILE CH <> Ξ*Ξ DO                                           COMP     604
000557           BEGIN NEXTCH;                                               COMP     605
000560            IF EOL THEN ENDOFLINE                                      COMP     606
000561           END;                                                        COMP     607
000563          NEXTCH                                                       COMP     608
000563         UNTIL CH = Ξ)Ξ;                                               COMP     609
000567         NEXTCH; GOTO 1                                                COMP     610
000571        END;                                                           COMP     611
000571       SY # LPARENT; OP # NOOP                                         COMP     612
000572      END;                                                             COMP     613
000574     Ξ<Ξ#                                                             COMP     614
000574      BEGIN NEXTCH; SY # RELOP;                                       COMP     615
000577       IF CH = Ξ=Ξ THEN                                               COMP     616
000601        BEGIN OP # LEOP; NEXTCH END                                   COMP     617
000603       ELSE                                                           COMP     618
000604        IF CH = Ξ>Ξ THEN                                              COMP     619
000606         BEGIN OP # NEOP; NEXTCH END                                  COMP     620
000610        ELSE OP # LTOP                                                COMP     621
000611      END;                                                             COMP     622
000613     Ξ>Ξ#                                                             COMP     623
000613      BEGIN NEXTCH; SY # RELOP;                                       COMP     624
000616       IF CH = Ξ=Ξ THEN                                               COMP     625
000620        BEGIN OP # GEOP; NEXTCH END                                   COMP     626
000622       ELSE OP # GTOP                                                 COMP     627
000623      END;                                                             COMP     628
000625     Ξ¬Ξ,Ξ*Ξ,Ξ/Ξ,Ξ^Ξ,Ξ+Ξ,Ξ-Ξ,                                       COMP     629
000625     Ξ∨Ξ,Ξ≤Ξ,Ξ≥Ξ,Ξ≠Ξ,Ξ=Ξ,Ξ)Ξ,                                       COMP     630
000625     Ξ(Ξ,Ξ]Ξ,Ξ,Ξ,Ξ;Ξ,Ξ↑Ξ,Ξ↓Ξ                                       HEXCON    20
000625                                 #                                     COMP     632
000625      BEGIN SY # SSY[CH]; OP # SOP[CH];                               COMP     633
000630       NEXTCH                                                         COMP     634
000630      END;                                                            COMP     635
000632    END (*CASE*)                                                      COMP     636
000732  END (*INSYMBOL*) ;                                                  COMP     637
000742                                                                      COMP     638
000003  PROCEDURE ENTERID(FCP# CTP);                                        COMP     639
000004   (*ENTER ID POINTED AT BY FCP INTO THE NAME-TABLE,                   COMP     640
000004    WHICH ON EACH DECLARATION LEVEL IS ORGANISED AS                   COMP     641
000004    AN UNBALANCED BINARY TREE*)                                       COMP     642
000004   VAR NAM# ALFA; LCP, LCP1 # CTP; LLEFT# BOOLEAN;                    COMP     643
000010  BEGIN NAM # FCP↑.NAME;                                              COMP     644
000010   LCP # DISPLAY[TOP].FNAME;                                          COMP     645
000012   IF LCP = NIL THEN                                                  COMP     646
000014    DISPLAY[TOP].FNAME # FCP                                          COMP     647
000014   ELSE                                                               COMP     648
000020    BEGIN                                                             COMP     649
000020     REPEAT LCP1 # LCP;                                               COMP     650
000021      IF LCP↑.NAME = NAM THEN   (*NAME CONFLICT, FOLLOW RIGHT LINK*)  COMP     651
000023       BEGIN ERROR(101); LCP # LCP↑.RLINK; LLEFT # FALSE END          COMP     652
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000027       ELSE                                                                COMP  653
000027        IF LCP↑.NAME < NAM THEN                                            COMP  654
000031         BEGIN LCP := LCP↑.RLINK; LLEFT := FALSE END                       COMP  655
000033        ELSE BEGIN LCP := LCP↑.LLINK; LLEFT := TRUE END                    COMP  656
000036      UNTIL LCP = NIL;                                                     COMP  657
000040      IF LLEFT THEN LCP1↑.LLINK := FCP ELSE LCP1↑.RLINK := FCP             COMP  658
000045     END;                                                                  COMP  659
000047    FCP↑.LLINK := NIL; FCP↑.RLINK := NIL                                   COMP  660
000052  END (*ENTERID*) ;                                                        COMP  661
000060                                                                           COMP  662
000003  PROCEDURE SEARCHSECTION(FCP: CTP; VAR FCP1: CTP);                        COMP  663
000005   (*TO FIND RECORD FIELDS AND FORWARD DECLARED PROCEDURE IDES              COMP  664
000005     --> PROCEDURE PROCEDUREDECLARATION                                    COMP  665
000005     --> PROCEDURE SELECTOR*)                                              COMP  666
000005   LABEL 1;                                                                COMP  667
000005  BEGIN                                                                    COMP  668
000005   WHILE FCP <> NIL DO                                                     COMP  669
000011    IF FCP↑.NAME = ID THEN GOTO 1                                          COMP  670
000013    ELSE IF FCP↑.NAME < ID THEN FCP := FCP↑.RLINK                          COMP  671
000016     ELSE FCP := FCP↑.LLINK;                                               COMP  672
000023 1:  FCP1 := FCP                                                           COMP  673
000024  END (*SEARCHSECTION*) ;                                                  COMP  674
000030                                                                           COMP  675
000003  PROCEDURE SEARCHID(FIDCLS: SETOFIDS; VAR FCP: CTP);                      COMP  676
000005   LABEL 1;                                                                COMP  677
000005   VAR LCP: CTP;                                                           COMP  678
000006  BEGIN                                                                    COMP  679
000006   FOR DISX := TOP DOWNTO 0 DO                                             COMP  680
000010    BEGIN LCP := DISPLAY[DISX].FNAME;                                      COMP  681
000014     WHILE LCP <> NIL DO                                                   COMP  682
000016      IF LCP↑.NAME = ID THEN                                               COMP  683
000020       IF LCP↑.KLASS IN FIDCLS THEN GOTO 1                                 COMP  684
000024       ELSE                                                                COMP  685
000024        BEGIN IF PRTERR THEN ERROR(103);                                   COMP  686
000027         LCP := LCP↑.RLINK                                                 COMP  687
000030        END                                                                COMP  688
000031      ELSE                                                                 COMP  689
000032       IF LCP↑.NAME < ID THEN                                              COMP  690
000034        LCP := LCP↑.RLINK                                                  COMP  691
000034       ELSE LCP := LCP↑.LLINK                                              COMP  692
000036    END;                                                                   COMP  693
000043   (*SEARCH NOT SUCCSESSFUL; SUPPRESS ERROR MESSAGE IN CASE                COMP  694
000043    OF FORWARD REFERENCED TYPE ID IN POINTER TYPE DEFINITION               COMP  695
000043    OR VARIANTS WITHOUT TAGFIELDS                                          COMP  696
000043    --> PROCEDURE FIELDLIST                                                COMP  697
000043    --> PROCEDURE SIMPLETYPE*)                                             COMP  698
000043   IF PRTERR THEN                                                          COMP  699
000044    BEGIN ERROR(104);                                                      COMP  700
000046     (*TO AVOID RETURNING NIL, REFERENCE AN ENTRY                          COMP  701
000046      FOR AN UNDECLARED ID OF APPROPRIATE CLASS                            COMP  702
000046      --> PROCEDURE ENTERUNDECL*)                                          COMP  703
000046     IF TYPES IN FIDCLS THEN LCP := UTYPPTR                                COMP  704
000050     ELSE                                                                  COMP  705
000052      IF VARS IN FIDCLS THEN LCP := UVARPTR                                COMP  706
000054      ELSE                                                                 COMP  707
000056       IF FIELD IN FIDCLS THEN LCP := UFLDPTR                              COMP  708
```

MOORE BUSINESS FORMS, INC., FORM 1413 PRINTED IN U.S.A.

```
000060        ELSE                                                              COMP     709
000062         IF KONST IN FIDCLS THEN LCP := UCSTPTR                           COMP     710
000064         ELSE                                                             COMP     711
000066          IF PROC IN FIDCLS THEN LCP := UPRCPTR                           COMP     712
000070          ELSE LCP := UFCTPTR;                                            COMP     713
000074     END;                                                                 COMP     714
000074 1:   FCP := LCP                                                          COMP     715
000075   END (*SEARCHID*) ;                                                     COMP     716
000101                                                                          COMP     717
000101                                                                          COMP     718
000003   PROCEDURE GETBOUNDS(FSP: STP; VAR FMIN,FMAX: INTEGER);                 COMP     719
000006    (*GET INTERNAL BOUNDS OF SUBRANGE OR SCALAR TYPE*)                    COMP     720
000006    (*ASSUME (FSP <> INTPTR) AND (FSP <> REALPTR)*)                       COMP     721
000006    BEGIN                                                                 COMP     722
000006     IF FSP <> NIL THEN                                                   COMP     723
000011      WITH FSP^ DO                                                        COMP     724
000012       IF FORM = SUBRANGE THEN                                            COMP     725
000014        BEGIN FMIN := MIN.IVAL; FMAX := MAX.IVAL END                      COMP     726
000016       ELSE                                                               COMP     727
000016        BEGIN FMIN := 0; FMAX := 0;                                       COMP     728
000017         IF FORM = SCALAR THEN                                            COMP     729
000021          BEGIN                                                           COMP     730
000021           IF SCALKIND = STANDARD THEN                                    COMP     731
000023            BEGIN IF FSP = CHARPTR THEN FMAX := 63                        COMP     732
000025            END                                                           COMP     733
000026           ELSE                                                           COMP     734
000027            IF FSP^.FCONST <> NIL THEN                                    COMP     735
000032              FMAX := FSP^.FCONST^.VALUES.IVAL                            COMP     736
000033          END                                                             COMP     737
000034        END                                                               COMP     738
000034    END (*GETBOUNDS*);                                                    COMP     739
000040                                                                          COMP     740
000003   PROCEDURE SKIP(FSYS: SETOFSYS);                                        COMP     741
000004    (*SKIP INPUT STRING UNTIL RELEVANT SYMBOL FOUND*)                     COMP     742
000004   BEGIN WHILE NOT (SY IN FSYS) DO INSYMBOL                               COMP     743
000012   END (*SKIP*) ;                                                         COMP     744
000020                                                                          COMP     745
000020                                                                          COMP     746
000003   PROCEDURE BLOCK(FSYS: SETOFSYS; FSY: SYMBOL; FPROCP: CTP);             COMP     747
000006    VAR LSY: SYMBOL; FLABP: LBP; LFSTCSP: CSP;                            COMP     748
000011        TRAPSET : BOOLEAN ;  TRADD : ADDRRANGE ;  TRAPLAB : INTEGER ;     UPDAT10    2
000014                                                                          COMP     749
000014                                                                          COMP     750
000003    FUNCTION NROFBITS(FVAL: INTEGER) : INTEGER;                           COMP     751
000004     (*COMPUTE NR OF BITS NECESSARY TO REPRESENT 0..FVAL*)                 COMP     752
000004     VAR B: INTEGER;                                                      COMP     753
000005    BEGIN B := 0;                                                         COMP     754
000007     REPEAT FVAL := FVAL DIV 2; B := B + 1                                COMP     755
000011     UNTIL FVAL = 0;                                                      COMP     756
000013     NROFBITS := B                                                        COMP     757
000013    END (*NROFBITS*);                                                     COMP     758
000017                                                                          COMP     759
000003    FUNCTION FULLWORDS(FSIZE: WBSIZE) : INTEGER;                          COMP     760
000004    BEGIN                                                                 COMP     761
000004     WITH FSIZE DO                                                        COMP     762
000007      IF BITS <> 0 THEN                                                   COMP     763
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000011      FULLWORDS := WORDS + 1                                          COMP    764
000011     ELSE FULLWORDS := WORDS                                          COMP    765
000013   END (*FULLWORDS*) ;                                                COMP    766
000021                                                                      COMP    767
000003   FUNCTION COMPTYPES(FSP1,FSP2: STP; FPCKD: BOOLEAN) : BOOLEAN;      COMP    768
000006    (*DECIDE WHETHER STRUCT POINTED AT BY FSP1 AND FSP2 ARE COMPATIBLE;COMP    769
000006     COMPREHENDING STRUCTURES ARE FPCKD (FALSE AT INITIAL CALL)*)     COMP    770
000006    VAR NXT1,NXT2: CTP; COMP: BOOLEAN;                                COMP    771
000011                                                                      COMP    772
000003    FUNCTION COMPLISTS(FCP1,FCP2: CTP; FSP1,FSP2: STP; FPCKD: BOOLEAN)COMP    773
000010     : BOOLEAN;                                                       COMP    774
000010     (*DECIDE WHETHER FIELDLISTS ARE COMPATIBLE*)                     COMP    775
000010     (*FCP1, FCP2: HEADS OF FIELDLISTS; FSP1, FSP2: POINTERS TO HEAD OF COMP    776
000010      VARIANT CHAIN*)                                                 COMP    777
000010     VAR COMP: BOOLEAN; NXT1, NXT2: STP;                              COMP    778
000013    BEGIN COMP := TRUE;                                               COMP    779
000011     WHILE COMP AND (FCP1 <> NIL)AND (FCP2 <> NIL) DO                 COMP    780
000016      BEGIN COMP := COMP AND COMPTYPES(FCP1↑.IDTYPE,FCP2↑.IDTYPE,FPCKD);COMP    781
000023       FCP1 := FCP1↑.NEXT; FCP2 := FCP2↑.NEXT                         COMP    782
000026      END;                                                            COMP    783
000030     COMP := COMP AND (FCP1 = FCP2);                                  COMP    784
000032     IF (FSP1 <> NIL)AND (FSP2 <> NIL) THEN                           COMP    785
000035      BEGIN                                                           COMP    786
000035       IF (FSP1↑.TAGFIELDP <> NIL)AND (FSP2↑.TAGFIELDP <> NIL) THEN   COMP    787
000042        COMP := COMP AND COMPTYPES(FSP1↑.TAGFIELDP↑.IDTYPE,           COMP    788
000044            FSP2↑.TAGFIELDP↑.IDTYPE,FPCKD);                           COMP    789
000050       NXT1 := FSP1↑.FSTVAR; NXT2 := FSP2↑.FSTVAR;                    COMP    790
000054       WHILE COMP AND (NXT1 <> NIL)AND (NXT2 <> NIL) DO                COMP    791
000051        BEGIN COMP := COMP AND COMPLISTS(NXT1↑.FSTVARFLD,NXT2↑.FSTVARFLD,COMP    792
000063              NXT1↑.SUBVAR,NXT2↑.SUBVAR,FPCKD);                        COMP    793
000070         NXT1 := NXT1↑.NXTVAR; NXT2 := NXT2↑.NXTVAR                    COMP    794
000072        END;                                                          COMP    795
000074       COMPLISTS := COMP AND (NXT1 = NXT2)                            COMP    796
000075      END                                                             COMP    797
000076     ELSE COMPLISTS := COMP AND (FSP1 = FSP2)                         COMP    798
000100    END (*COMPLISTS*) ;                                               COMP    799
000106                                                                      COMP    800
000003    FUNCTION EQUALBOUNDS(FSP1,FSP2: STP) : BOOLEAN;                   COMP    801
000005     VAR LMIN1,LMIN2,LMAX1,LMAX2: INTEGER;                            COMP    802
000011    BEGIN GETBOUNDS(FSP1,LMIN1,LMAX1);                                COMP    803
000011     GETBOUNDS(FSP2,LMIN2,LMAX2);                                    COMP    804
000014     EQUALBOUNDS := (LMIN1 = LMIN2)AND (LMAX1 = LMAX2)               COMP    805
000016    END (*EQUALBOUNDS*) ;                                             COMP    806
000025                                                                      COMP    807
000025   BEGIN (*COMPTYPES*)                                                COMP    808
000025    IF FSP1 = FSP2 THEN COMPTYPES := TRUE                             COMP    809
000011    ELSE                                                              COMP    810
000012     IF (FSP1 <> NIL)AND (FSP2 <> NIL) THEN                           COMP    811
000016      IF FSP1↑.FORM = FSP2↑.FORM THEN                                 COMP    812
000021       CASE FSP1↑.FORM OF                                             COMP    813
000022        SCALAR:                                                       COMP    814
000022         IF (FSP1↑.SCALKIND = STANDARD)OR(FSP2↑.SCALKIND = STANDARD) THEN COMP    815
000027          COMPTYPES := FALSE                                          COMP    816
000027         ELSE                                                         COMP    817
000030          BEGIN NXT1 := FSP1↑.FCONST; NXT2 := FSP2↑.FCONST;           COMP    818
000033           COMP := TRUE;                                              COMP    819
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000033          WHILE COMP AND (NXT1 <> NIL)AND (NXT2 <> NIL) DO                    COMP  820
000040           BEGIN COMP := COMP AND (NXT1↑.NAME = NXT2↑.NAME);                  COMP  821
000042            NXT1 := NXT1↑.NEXT; NXT2 := NXT2↑.NEXT                            COMP  822
000043           END;                                                               COMP  823
000045          COMPTYPES := COMP AND (NXT1 = NXT2)                                 COMP  824
000046         END;                                                                 COMP  825
000050        SUBRANGE:                                                            COMP  826
000050         BEGIN COMP := COMPTYPES(FSP1↑.RANGETYPE,FSP2↑.RANGETYPE,FPCKD);     COMP  827
000056          IF COMP AND FPCKD AND (FSP1↑.RANGETYPE <> REALPTR) THEN            COMP  828
000063           COMP := COMP AND EQUALBOUNDS(FSP1,FSP2);                          COMP  829
000070          COMPTYPES := COMP                                                  COMP  830
000070         END;                                                                COMP  831
000072        POINTER:                                                             COMP  832
000072         COMPTYPES := COMPTYPES(FSP1↑.ELTYPE,FSP2↑.ELTYPE,FALSE);            COMP  833
000101        POWER:                                                               COMP  834
000101         COMPTYPES := (FSP1↑.PCKDSET = FSP2↑.PCKDSET)                       COMP  835
000103               AND COMPTYPES(FSP1↑.ELSET,FSP2↑.ELSET,FSP1↑.PCKDSET);        COMP  836
000115        ARRAYS:                                                              COMP  837
000115         BEGIN COMP := (FSP1↑.PCKDARR = FSP2↑.PCKDARR)                      COMP  838
000120               AND COMPTYPES(FSP1↑.AELTYPE,FSP2↑.AELTYPE,FSP1↑.PCKDARR)     COMP  839
000125               AND COMPTYPES(FSP1↑.INXTYPE,FSP2↑.INXTYPE,FALSE);             COMP  840
000142          COMPTYPES := COMP AND EQUALBOUNDS(FSP1↑.INXTYPE,                   COMP  841
000145                       FSP2↑.INXTYPE);                                       COMP  842
000151         END;                                                                COMP  843
000152        RECORDS:                                                             COMP  844
000152         COMPTYPES := (FSP1↑.PCKDREC = FSP2↑.PCKDREC)                       COMP  845
000154               AND COMPLISTS(FSP1↑.FSTFLD,FSP2↑.FSTFLD,                      COMP  846
000160                     FSP1↑.RECVAR,FSP2↑.RECVAR,FSP1↑.PCKDREC);               COMP  847
000170          FILES:                                                            COMP  848
000170           COMPTYPES := (FSP1↑.PCKDFIL = FSP2↑.PCKDFIL)                     COMP  849
000172                AND COMPTYPES(FSP1↑.FILTYPE,FSP2↑.FILTYPE,FSP1↑.PCKDFIL)    COMP  850
000200        END (*CASE*)                                                        COMP  851
000213       ELSE (*FSP1↑.FORM <> FSP2↑.FORM*)                                   COMP  852
000214        IF FSP1↑.FORM = SUBRANGE THEN                                      COMP  853
000216         BEGIN COMP := COMPTYPES(FSP1↑.RANGETYPE,FSP2,FPCKD);              COMP  854
000223          IF COMP AND FPCKD AND (FSP2 <> REALPTR) THEN                      COMP  855
000227           COMP := COMP AND EQUALBOUNDS(FSP1,FSP2);                         COMP  856
000234          COMPTYPES := COMP                                                 COMP  857
000234         END                                                                COMP  858
000236        ELSE                                                               COMP  859
000236         IF FSP2↑.FORM = SUBRANGE THEN                                     COMP  860
000240          BEGIN COMP := COMPTYPES(FSP1,FSP2↑.RANGETYPE,FPCKD);             COMP  861
000245           IF COMP AND FPCKD AND (FSP1 <> REALPTR) THEN                     COMP  862
000251            COMP := COMP AND EQUALBOUNDS(FSP1,FSP2);                        COMP  863
000257           COMPTYPES := COMP                                                COMP  864
000257          END                                                               COMP  865
000261         ELSE COMPTYPES := FALSE                                           COMP  866
000261       ELSE COMPTYPES := TRUE                                              COMP  867
000263    END (*COMPTYPES*) ;                                                    COMP  868
000271                                                                           COMP  869
000003    FUNCTION STRING(FSP: STP) : BOOLEAN;                                   COMP  870
000004    BEGIN STRING := FALSE;                                                 COMP  871
000007     IF FSP <> NIL THEN                                                    COMP  872
000011      WITH FSP↑ DO                                                         COMP  873
000012       IF FORM = ARRAYS THEN                                               COMP  874
000014        IF COMPTYPES(AELTYPE,CHARPTR,FALSE) THEN STRING := PCKDARR         COMP  875
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000022  END (*STRING*) ;                                                          COMP  876
000031                                                                            COMP  877
000003  PROCEDURE STRINGTYPE(VAR FSP: STP);                                       COMP  878
000004   (*ENTER TYPE OF STRINGCONST (PACKED ARRAY [1..LGTH] OF CHAR) INTO        COMP  879
000004    STRUCTURE TABLE*)                                                       COMP  880
000004   VAR LSP,LSP1: STP;                                                       COMP  881
000006  BEGIN NEW(LSP,SUBRANGE);                                                  COMP  882
000010   WITH LSP↑ DO                                                            COMP  883
000011    BEGIN FORM := SUBRANGE; RANGETYPE := INTPTR;                           COMP  884
000015     MIN.IVAL := 1; MAX.IVAL := LGTH ; FTYPE := FALSE;                     COMP  885
000022     WITH SIZE DO                                                          COMP  886
000023      BEGIN WORDS := 0; BITS := NROFBITS(LGTH) END                         COMP  887
000034    END;                                                                   COMP  888
000034   NEW(LSP1,ARRAYS,TRUE,TRUE);                                             COMP  889
000036   WITH LSP1↑ DO                                                           COMP  890
000040    BEGIN FORM := ARRAYS;                                                  COMP  891
000042     AELTYPE := CHARPTR; INXTYPE := LSP;                                   COMP  892
000047     PCKDARR := TRUE; PARTWORDELS := TRUE;                                 COMP  893
000053     ELSPERWORD := ALFALENG; FTYPE := FALSE;                               COMP  894
000057     WITH SIZE DO                                                          COMP  895
000060      BEGIN WORDS := LGTH DIV ALFALENG;                                    COMP  896
000065       BITS := LGTH MOD ALFALENG*6                                        COMP  897
000070      END                                                                   COMP  898
000073    END;                                                                   COMP  899
000073   FSP := LSP1                                                             COMP  900
000073  END (*STRINGTYPE*) ;                                                     COMP  901
000100                                                                            COMP  902
000003  PROCEDURE CONVERTCST(FVALU: VALU; FTYPTR: STP; VAR FCST: INTEGER);       COMP  903
000010   (*CONVERT CONSTANT FVALU WITH TYPE FTYPTR (<>STRING) TO INTERNAL        COMP  904
000010    REPRESENTATION FCST*)                                                  COMP  905
000010  BEGIN                                                                     COMP  906
000010   IF FTYPTR <> NIL THEN                                                   COMP  907
000015    IF FTYPTR↑.FORM = POWER THEN FCST := ORD(FVALU.PVAL)                   COMP  908
000020    ELSE                                                                   COMP  909
000021     IF COMPTYPES(FTYPTR,INTPTR,FALSE) THEN FCST := FVALU.IVAL             COMP  910
000030     ELSE                                                                  COMP  911
000031      IF COMPTYPES(FTYPTR,REALPTR,FALSE) THEN FCST := ORD(FVALU.RVAL)      COMP  912
000040      ELSE FCST := ORD(FVALU.BVAL);                                        COMP  913
000043  END (*CONVERTCST*) ;                                                      COMP  914
000047                                                                            COMP  915
000003  PROCEDURE CONSTANT(FSYS: SETOFSYS; VAR FSP: STP; VAR FVALU: VALU);       COMP  916
000006   VAR LSP: STP; LCP: CTP; SIGN: (NONE,POS,NEG); LCSP: CSP;                 COMP  917
000012  BEGIN LSP := NIL; FVALU.IVAL := 0;                                        COMP  918
000011   IF NOT (SY IN CONSTBEGSYS) THEN                                         COMP  919
000014    BEGIN ERROR(50); SKIP(FSYS+CONSTBEGSYS) END;                           COMP  920
000020   IF SY IN CONSTBEGSYS THEN                                               COMP  921
000023    BEGIN                                                                  COMP  922
000023     IF SY = CHARCONST THEN                                                COMP  923
000024      BEGIN LSP := CHARPTR; FVALU.IVAL := IVAL; INSYMBOL END               COMP  924
000030     ELSE                                                                  COMP  925
000031      IF SY = STRINGCONST THEN                                             COMP  926
000033       BEGIN STRINGTYPE(LSP);                                              COMP  927
000035        FVALU.VALP := CONSTP;                                              COMP  928
000037        INSYMBOL                                                           COMP  929
000037       END                                                                 COMP  930
000040     ELSE                                                                  COMP  931
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000041      BEGIN                                                                COMP    932
000041       SIGN := NONE;                                                       COMP    933
000042       IF (SY = ADDOP)AND (OP IN [PLUS,MINUS]) THEN                        COMP    934
000046        BEGIN IF OP = PLUS THEN SIGN := POS ELSE SIGN := NEG;              COMP    935
000052         INSYMBOL                                                          COMP    936
000052        END;                                                               COMP    937
000053       IF SY = IDENT THEN                                                  COMP    938
000055        BEGIN SEARCHID([KONST],LCP);                                       COMP    939
000057         WITH LCP^ DO                                                      COMP    940
000061          BEGIN LSP := IDTYPE; FVALU := VALUES END;                        COMP    941
000065         IF SIGN <> NONE THEN                                              COMP    942
000066          IF LSP = INTPTR THEN                                             COMP    943
000070           BEGIN IF SIGN = NEG THEN FVALU.IVAL := -FVALU.IVAL END          COMP    944
000072          ELSE                                                             COMP    945
000073           IF LSP = REALPTR THEN                                           COMP    946
000074            BEGIN                                                          COMP    947
000074             IF SIGN = NEG THEN FVALU.RVAL := -FVALU.RVAL                  COMP    948
000076            END                                                            COMP    949
000077         ELSE ERROR(105);                                                  COMP    950
000102         INSYMBOL;                                                         COMP    951
000103        END                                                                COMP    952
000103       ELSE                                                                COMP    953
000104        IF SY = INTCONST THEN                                              COMP    954
000105         BEGIN IF SIGN = NEG THEN IVAL := -IVAL;                           COMP    955
000110          LSP := INTPTR; FVALU.IVAL := IVAL; INSYMBOL                      COMP    956
000113         END                                                               COMP    957
000114        ELSE                                                               COMP    958
000115         IF SY = REALCONST THEN                                            COMP    959
000116          BEGIN IF SIGN = NEG THEN RVAL := -RVAL;                          COMP    960
000121           LSP := REALPTR; FVALU.RVAL := RVAL; INSYMBOL                    COMP    961
000124          END                                                              COMP    962
000125         ELSE                                                              COMP    963
000126          BEGIN ERROR(106); SKIP(FSYS) END                                 COMP    964
000132       END;                                                                COMP    965
000132      IF NOT (SY IN FSYS) THEN                                             COMP    966
000135       BEGIN ERROR(6); SKIP(FSYS) END                                      COMP    967
000140      END;                                                                 COMP    968
000140    FSP := LSP                                                             COMP    969
000141  END (*CONSTANT*) ;                                                       COMP    970
000146                                                                           COMP    971
000146                                                                           COMP    972
000003  PROCEDURE TYP(FSYS: SETOFSYS; VAR FSP: STP);                             COMP    973
000005   VAR LSP,LSP1,LSP2: STP; OLDTOP: DISPRANGE; LCP: CTP;                    COMP    974
000012     LSIZE,DISPL: WBSIZE; LMIN,LMAX,LRL: INTEGER;                          COMP    975
000017     T,T1: INTEGER; PACKFLAG,SEGFLAG,LFILTYP,EXITLOOP: BOOLEAN;            COMP    976
000025                                                                           COMP    977
000003    PROCEDURE SIMPLETYPE(FSYS: SETOFSYS; VAR FSP: STP);                    COMP    978
000005     VAR LSP,LSP1: STP; LCP,LCP1: CTP; TTOP: DISPRANGE;                    COMP    979
000012      LVAL: INTEGER; LVALU: VALU;                                          COMP    980
000015                                                                           COMP    981
000003     PROCEDURE SUBRANGES(FSP: STP; FVALU: VALU);                           COMP    982
000007      (*PROCESS SUBRANGE TYPE*)                                            COMP    983
000007      VAR LOW,HIGH: INTEGER;                                               COMP    984
000011     BEGIN NEW(LSP,SUBRANGE);                                              COMP    985
000015      WITH LSP^ DO                                                         COMP    986
000017       BEGIN RANGETYPE := FSP; FORM := SUBRANGE;                           COMP    987
```

MOORE BUSINESS FORMS, INC. F
FORM 113
PRINTED IN U.S.A.

```
000023          MIN := FVALU; FTYPE := FALSE                                    COMP     988
000026        END;                                                              COMP     989
000031      IF SY = COLON THEN INSYMBOL ELSE ERROR(5);                          COMP     990
000037      CONSTANT(FSYS,LSP1,LVALU);                                          COMP     991
000044      WITH LSP↑ DO                                                        COMP     992
000046       BEGIN MAX := LVALU;                                                COMP     993
000051        WITH SIZE DO                                                      COMP     994
000052         BEGIN WORDS := 1; BITS := 0 END;                                 COMP     995
000056        IF NOT COMPTYPES(FSP,LSP1,FALSE) THEN ERROR(107)                  COMP     996
000065        ELSE                                                              COMP     997
000067         WITH SIZE DO                                                     COMP     998
000070          IF FSP = REALPTR THEN                                           COMP     999
000072           BEGIN IF MIN.RVAL >MAX.RVAL THEN ERROR(102)                    COMP    1000
000075           END                                                            COMP    1001
000076          ELSE                                                            COMP    1002
000077           IF STRING(FSP) THEN                                            COMP    1003
000103            BEGIN ERROR(148); RANGETYPE := NIL                            COMP    1004
000105            END                                                           COMP    1005
000107           ELSE                                                           COMP    1006
000107            BEGIN LOW := MIN.IVAL; HIGH := MAX.IVAL;                      COMP    1007
000112             IF LOW > HIGH THEN ERROR(102);                               COMP    1008
000114             WORDS := 0;                                                  COMP    1009
000117             IF ABS(LOW) < ABS(HIGH) THEN                                 COMP    1010
000122              BITS := NROFBITS(ABS(HIGH))                                 COMP    1011
000124             ELSE BITS := NROFBITS(ABS(LOW));                             COMP    1012
000140             IF LOW < 0 THEN BITS := BITS + 1                             COMP    1013
000142               END                                                        COMP    1014
000145         END                                                              COMP    1015
000145      END (*SUBRANGES*);                                                  COMP    1016
000151                                                                          COMP    1017
000151    BEGIN (*SIMPLETYPE*)                                                  COMP    1018
000151     IF NOT (SY IN SIMPTYPEBEGSYS) THEN                                   COMP    1019
000012      BEGIN ERROR(1); SKIP(FSYS+SIMPTYPEBEGSYS) END;                      COMP    1020
000016     IF SY IN SIMPTYPEBEGSYS THEN                                         COMP    1021
000021      BEGIN                                                               COMP    1022
000021       IF SY = LPARENT THEN                                               COMP    1023
000022        BEGIN TTOP := TOP;   (*DECL. CONSTS LOCAL TO INNERMOST BLOCK*)    COMP    1024
000024         WHILE DISPLAY[TOP].OCCUR <> BLCK DO TOP := TOP - 1;              COMP    1025
000031         NEW(LSP,SCALAR,DECLARED);                                        COMP    1026
000033         WITH LSP↑ DO                                                     COMP    1027
000035          BEGIN FORM := SCALAR; SCALKIND := DECLARED; FTYPE := FALSE;     COMP    1028
000043           FCONST := NIL                                                  COMP    1029
000043          END;                                                            COMP    1030
000045         LCP1 := NIL; LVAL := -1;                                         COMP    1031
000047         REPEAT INSYMBOL;                                                 COMP    1032
000050          IF SY = IDENT THEN                                              COMP    1033
000052           BEGIN NEW(LCP,KONST); LVAL := LVAL + 1;                        COMP    1034
000055            WITH LCP↑ DO                                                  COMP    1035
000056             BEGIN NAME := ID; IDTYPE := LSP; NEXT := LCP1;               COMP    1036
000064              VALUES.IVAL := LVAL; KLASS := KONST                         COMP    1037
000065             END;                                                         COMP    1038
000067            ENTERID(LCP);                                                 COMP    1039
000070            LCP1 := LCP; INSYMBOL                                         COMP    1040
000071           END                                                            COMP    1041
000072          ELSE ERROR(2);                                                  COMP    1042
000075          IF NOT (SY IN FSYS+[COMMA,RPARENT]) THEN                        COMP    1043
```

MOORE BUSINESS FORMS, INC. F
FORM 1413
PRINTED IN U.S.A.

```
000100            BEGIN ERROR(6); SKIP(FSYS+[COMMA,RPARENT]) END                    COMP  1044
000105           UNTIL SY <> COMMA;                                                 COMP  1045
000107           WITH LSP↑, SIZE DO                                                 COMP  1046
000111            BEGIN FCONST := LCP1;                                             COMP  1047
000114             WORDS := 0; BITS := NROFBITS(LVAL)                               COMP  1048
000120            END;                                                              COMP  1049
000124           TOP := TTOP;                                                       COMP  1050
000125           IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                        COMP  1051
000132          END                                                                 COMP  1052
000133         ELSE                                                                 COMP  1053
000134          BEGIN                                                               COMP  1054
000134           IF SY = IDENT THEN                                                 COMP  1055
000135            BEGIN SEARCHID([TYPES,KONST],LCP);                                COMP  1056
000137             INSYMBOL;                                                       COMP  1057
000140             WITH LCP↑ DO                                                    COMP  1058
000142              IF KLASS = KONST THEN SUBRANGES(IDTYPE,VALUES)                 COMP  1059
000145              ELSE                                                           COMP  1060
000150               LSP := IDTYPE                                                 COMP  1061
000150            END (*SY = IDENT*)                                               COMP  1062
000152           ELSE                                                              COMP  1063
000153            BEGIN CONSTANT(FSYS+[COLON],LSP1,LVALU);                         COMP  1064
000160             SUBRANGES(LSP1,LVALU)                                           COMP  1065
000162            END;                                                             COMP  1066
000163          END;                                                               COMP  1067
000163         FSP := LSP;                                                         COMP  1068
000165         IF NOT (SY IN FSYS) THEN                                            COMP  1069
000167          BEGIN ERROR(6); SKIP(FSYS) END                                     COMP  1070
000172        END                                                                  COMP  1071
000172         ELSE FSP := NIL                                                     COMP  1072
000174      END (*SIMPLETYPE*) ;                                                   COMP  1073
000202                                                                             COMP  1074
000003      PROCEDURE FIELDLIST(FSYS: SETOFSYS; VAR FRECVAR: STP;                  COMP  1075
000005                VAR FFSTFLD: CTP; VAR FTYP: BOOLEAN);                        COMP  1076
000007       VAR LASTFLD, LCP,LCP1,NXT,NXT1: CTP; LSP,LSP1,LSP2,LSP3,LSP4: STP;    COMP  1077
000021        MINSIZE,MAXSIZE: WBSIZE; LVALU: VALU;                                COMP  1078
000025        TAGFLAG,LFILTYP,EXITLOOP: BOOLEAN;                                   COMP  1079
000030                                                                             COMP  1080
000003       PROCEDURE FIELDADDRESS(FCP: CTP; FSIZE: WBSIZE);                      COMP  1081
000005        (*COMPUTE ADDRESS OF FCP↑ ACCORDING TO ITS SIZE *)                   COMP  1082
000005                                                                             COMP  1083
000003        PROCEDURE ADJUST;                                                    COMP  1084
000003         (*ADJUST LASTFLD*)                                                  COMP  1085
000003        BEGIN                                                                COMP  1086
000003         IF NOT TAGFLAG THEN                                                 COMP  1087
000010          WITH LASTFLD↑ DO                                                   COMP  1088
000012           IF IDTYPE <> NIL THEN                                             COMP  1089
000014            IF IDTYPE↑.FORM <= POWER THEN                                    COMP  1090
000017             IF BITADDR = 0 THEN PCKDFLD := FALSE                            COMP  1091
000021             ELSE                                                            COMP  1092
000023              BITADDR := WORDSIZE - IDTYPE↑.SIZE.BITS;                       COMP  1093
000027         WITH DISPL DO                                                       COMP  1094
000027          BEGIN WORDS := WORDS + 1; BITS := 0 END                            COMP  1095
000036        END (*ADJUST*);                                                      COMP  1096
000042                                                                             COMP  1097
000042      BEGIN (*FIELDADDRESS*)                                                 COMP  1098
000042       WITH DISPL, FCP↑ DO                                                   COMP  1099
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A.

```
000010        IF PACKFLAG AND (FSIZE.WORDS = 0) THEN                          COMP    1100
000014         BEGIN IF BITS + FSIZE.BITS > WORDSIZE THEN ADJUST;             COMP    1101
000021          FLDADDR := WORDS; PCKDFLD := TRUE;                            COMP    1102
000031          BITADDR := BITS;                                              COMP    1103
000033          IF BITS + FSIZE.BITS = WORDSIZE THEN                          COMP    1104
000036           BEGIN WORDS := WORDS + 1; BITS := 0 END                      COMP    1105
000043          ELSE BITS := BITS + FSIZE.BITS                                COMP    1106
000043         END                                                            COMP    1107
000046        ELSE                                                            COMP    1108
000047         BEGIN IF BITS <> 0 THEN ADJUST;                                COMP    1109
000053          FLDADDR := WORDS; PCKDFLD := FALSE;                           COMP    1110
000063          WORDS := WORDS + FULLWORDS(FSIZE)                             COMP    1111
000063         END;                                                           COMP    1112
000071      END (*FIELDADDRESS*) ;                                            COMP    1113
000075                                                                        COMP    1114
000075     BEGIN (*FIELDLIST*) NXT1 := NIL; LSP := NIL;                       COMP    1115
000012      TAGFLAG := TRUE; FTYP := FALSE;                                   COMP    1116
000014      IF NOT (SY IN FSYS+[IDENT,CASESY]) THEN                           COMP    1117
000016       BEGIN ERROR(19); SKIP(FSYS+[IDENT,CASESY]) END;                  COMP    1118
000023      WHILE SY = IDENT DO                                               COMP    1119
000025       BEGIN NXT := NXT1;                                               COMP    1120
000026        (*LOOP UNTIL SY <> COMMA*)                                      COMP    1121
000026        REPEAT                                                          COMP    1122
000026         IF SY = IDENT THEN                                             COMP    1123
000030          BEGIN NEW(LCP,FIELD);                                         COMP    1124
000031           WITH LCP^ DO                                                 COMP    1125
000033            BEGIN NAME := ID; IDTYPE := NIL; NEXT := NXT;               COMP    1126
000041             KLASS := FIELD                                             COMP    1127
000041            END;                                                        COMP    1128
000043          NXT := LCP;                                                   COMP    1129
000043          ENTERID(LCP);                                                 COMP    1130
000045          INSYMBOL                                                      COMP    1131
000045         END                                                            COMP    1132
000046        ELSE ERROR(2);                                                  COMP    1133
000051        IF NOT (SY IN [COMMA,COLON]) THEN                               COMP    1134
000054         BEGIN ERROR(6); SKIP(FSYS+[COMMA,COLON,SEMICOLON,CASESY])      COMP    1135
000057         END;                                                           COMP    1136
000060        EXITLOOP := SY <> COMMA;                                        COMP    1137
000063        IF NOT EXITLOOP THEN INSYMBOL                                   COMP    1138
000064       UNTIL EXITLOOP;                                                  COMP    1139
000066       IF SY = COLON THEN INSYMBOL ELSE ERROR(5);                       COMP    1140
000074       TYP(FSYS+[CASESY,SEMICOLON],LSP);                                COMP    1141
000100       WHILE NXT <> NXT1 DO                                             COMP    1142
000102        WITH NXT^ DO                                                    COMP    1143
000103         BEGIN IDTYPE := LSP;                                           COMP    1144
000105          IF LSP <> NIL THEN                                            COMP    1145
000106           BEGIN FTYP := FTYP OR LSP^.FTYPE;                            COMP    1146
000110            FIELDADDRESS(NXT,LSP^.SIZE)                                 COMP    1147
000112           END                                                          COMP    1148
000113          ELSE                                                          COMP    1149
000114           BEGIN FLDADDR := DISPL.WORDS; PCKDFLD := FALSE END;          COMP    1150
000123          TAGFLAG := FALSE; LASTFLD := NXT; NXT := NEXT                 COMP    1151
000126         END;                                                           COMP    1152
000130       NXT1 := LCP;                                                     COMP    1153
000131       IF SY = SEMICOLON THEN                                           COMP    1154
000133        BEGIN INSYMBOL;                                                 COMP    1155
```

MOORE BUSINESS FORMS, INC., F
FORM 1413
PRINTED IN U.S.A

```
000134          IF NOT (SY IN FSYS+[IDENT,CASESY]) THEN                          COMP    1156
000137            BEGIN ERROR(19); SKIP(FSYS+[IDENT,CASESY]) END                 COMP    1157
000144          END                                                              COMP    1158
000144        END (*WHILE*);                                                     COMP    1159
000145      NXT := NIL;                                                          COMP    1160
000146      WHILE NXT1 <> NIL DO                                                 COMP    1161
000150        WITH NXT1↑ DO                                                      COMP    1162
000151          BEGIN LCP := NEXT; NEXT := NXT; NXT := NXT1; NXT1 := LCP END;    COMP    1163
000156      FFSTFLD := NXT;                                                      COMP    1164
000160      IF SY = CASESY THEN                                                  COMP    1165
000162        BEGIN NEW(LSP,TAGFIELD);                                           COMP    1166
000163          WITH LSP↑ DO                                                     COMP    1167
000165            BEGIN TAGFIELDP := NIL; FSTVAR := NIL; FORM := TAGFIELD;       COMP    1168
000173              FTYPE := FALSE                                               COMP    1169
000173            END;                                                           COMP    1170
000175          FRECVAR := LSP;                                                  COMP    1171
000176          INSYMBOL;                                                        COMP    1172
000177          IF SY = IDENT THEN                                               COMP    1173
000201            BEGIN PRTERR := FALSE; SEARCHID([TYPES],LCP1); PRTERR := TRUE; COMP    1174
000205              NEW(LCP,FIELD);                                              COMP    1175
000207              WITH LCP↑ DO                                                 COMP    1176
000211                BEGIN IDTYPE := NIL; KLASS := FIELD; NEXT := NIL END;      COMP    1177
000217              IF LCP1 = NIL THEN   (*EXPLICITE TAGFIELD*)                  COMP    1178
000220                BEGIN LCP↑.NAME := ID; ENTERID(LCP);                       COMP    1179
000222                  INSYMBOL;                                                COMP    1180
000223                  IF SY = COLON THEN INSYMBOL ELSE ERROR(5);               COMP    1181
000231                  IF SY = IDENT THEN SEARCHID([TYPES],LCP1)                COMP    1182
000234                ELSE                                                       COMP    1183
000236                  BEGIN ERROR(2); SKIP(FSYS+[OFSY,LPARENT]);               COMP    1184
000243                    LCP1 := NIL                                            COMP    1185
000243                  END                                                      COMP    1186
000244              END                                                          COMP    1187
000244            ELSE LCP↑.NAME := Ξ           Ξ;                             COMP    1188
000246            IF LCP1 <> NIL THEN                                            COMP    1189
000250              BEGIN LSP1 := LCP1↑.IDTYPE;                                  COMP    1190
000251                IF LSP1 <> NIL THEN                                        COMP    1191
000252                  BEGIN                                                    COMP    1192
000252                    IF LSP1↑.FORM <= SUBRANGE THEN                         COMP    1193
000254                      BEGIN                                                COMP    1194
000254                        IF COMPTYPES(REALPTR,LSP1,FALSE) THEN ERROR(109)   COMP    1195
000250                        ELSE                                               COMP    1196
000252                          BEGIN LSP↑.TAGFIELDP := LCP;                     COMP    1197
000265                            WITH LCP↑ DO                                   COMP    1198
000256                              BEGIN IDTYPE := LSP1;                        COMP    1199
000270                                IF NAME <> Ξ           Ξ THEN            COMP    1200
000271                                  FIELDADDRESS(LCP,LSP1↑.SIZE)             COMP    1201
000272                              END                                          COMP    1202
000273                          END                                              COMP    1203
000273                      END                                                  COMP    1204
000273                    ELSE ERROR(110)                                        COMP    1205
000275                  END;                                                     COMP    1206
000276                INSYMBOL                                                   COMP    1207
000276              END                                                          COMP    1208
000277          END                                                              COMP    1209
000277        ELSE                                                             COMP    1210
000300          BEGIN ERROR(2); SKIP(FSYS+[OFSY,LPARENT]) END;                 COMP    1211
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000305        LSP↑.SIZE := DISPL;                                                    COMP  1212
000311        IF SY = OFSY THEN INSYMBOL ELSE ERROR(8);                              COMP  1213
000317        LSP1 := NIL; MINSIZE := DISPL; MAXSIZE := DISPL;                       COMP  1214
000322        (*LOOP UNTIL SY <> SEMICOLON:*)                                        COMP  1215
000322        REPEAT                                                                 COMP  1216
000322         IF NOT (SY IN FSYS+[SEMICOLON]) THEN                                  COMP  1217
000325         BEGIN LSP2 := NIL;                                                    COMP  1218
000326         (*LOOP UNTIL SY <> SEMICOLON:*)                                       COMP  1219
000326         REPEAT CONSTANT(FSYS+[COMMA,COLON,LPARENT],LSP3,LVALU);               COMP  1220
000333          IF LSP↑.TAGFIELDP <> NIL THEN                                        COMP  1221
000336           IF NOT COMPTYPES(LSP↑.TAGFIELDP↑.IDTYPE,LSP3,FALSE) THEN            COMP  1222
000345            ERROR(111);                                                        COMP  1223
000346          NEW(LSP3,VARIANT);                                                   COMP  1224
000350          WITH LSP3↑ DO                                                        COMP  1225
000352           BEGIN NXTVAR := LSP1; SUBVAR := LSP2; VARVAL := LVALU;              COMP  1226
000350            FORM := VARIANT; FTYPE := FALSE                                    COMP  1227
000353           END;                                                                COMP  1228
000365          LSP1 := LSP3; LSP2 := LSP3;                                          COMP  1229
000367          EXITLOOP := SY <> COMMA;                                             COMP  1230
000372          IF NOT EXITLOOP THEN INSYMBOL                                        COMP  1231
000373         UNTIL EXITLOOP;                                                       COMP  1232
000375         IF SY = COLON THEN INSYMBOL ELSE ERROR(5);                            COMP  1233
000403         IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                          COMP  1234
000410         FIELDLIST(FSYS+[RPARENT,SEMICOLON],LSP2,LCP,LFILTYP);                 COMP  1235
000415         IF LFILTYP THEN ERROR(108);                                           COMP  1236
000420         FTYP := FTYP OR LFILTYP;                                              COMP  1237
000422          IF (DISPL.WORDS > MAXSIZE.WORDS)OR                                   COMP  1238
000426           (DISPL.WORDS = MAXSIZE.WORDS)AND (DISPL.BITS > MAXSIZE.BITS)       COMP  1239
000431           THEN MAXSIZE := DISPL;                                             COMP  1240
000434         WHILE LSP3 <> NIL DO                                                 COMP  1241
000436          WITH LSP3↑ DO                                                       COMP  1242
000437           BEGIN LSP4 := SUBVAR; SUBVAR := LSP2;                              COMP  1243
000442            SIZE := DISPL; FSTVARFLD := LCP;                                  COMP  1244
000447            LSP3 := LSP4                                                      COMP  1245
000447           END;                                                               COMP  1246
000450         IF SY = RPARENT THEN                                                 COMP  1247
000452          BEGIN INSYMBOL;                                                     COMP  1248
000453           IF NOT (SY IN FSYS+[SEMICOLON]) THEN                               COMP  1249
000456            BEGIN ERROR(6); SKIP(FSYS+[SEMICOLON]) END                        COMP  1250
000463          END                                                                 COMP  1251
000463         ELSE ERROR(4);                                                       COMP  1252
000466         END (*NOT (SY IN ...*) ;                                             COMP  1253
000466         EXITLOOP := SY <> SEMICOLON;                                         COMP  1254
000471         IF NOT EXITLOOP THEN                                                 COMP  1255
000472          BEGIN DISPL := MINSIZE; INSYMBOL END                                COMP  1256
000475        UNTIL EXITLOOP;                                                       COMP  1257
000476        DISPL := MAXSIZE;                                                     COMP  1258
000500        LSP↑.FSTVAR := LSP1;                                                  COMP  1259
000503       END                                                                    COMP  1260
000503      ELSE                                                                    COMP  1261
000503       FRECVAR := NIL                                                        COMP  1262
000504     END (*FIELDLIST*) ;                                                     COMP  1263
000517                                                                             COMP  1264
000517    BEGIN (*TYP*) LSP := NIL;                                                COMP  1265
000010     IF NOT (SY IN TYPEBEGSYS) THEN                                          COMP  1266
000013       BEGIN ERROR(10); SKIP(FSYS+TYPEBEGSYS) END;                           COMP  1267
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A.

```
000017    IF SY IN TYPEBEGSYS THEN                                                  COMP     1268
000022     BEGIN                                                                    COMP     1269
000022      IF SY IN SIMPTYPEBEGSYS THEN SIMPLETYPE(FSYS,LSP)                       COMP     1270
000025      ELSE                                                                    COMP     1271
000027   (*↑*)                                                                      COMP     1272
000027       IF SY = ARROW THEN                                                     COMP     1273
000031        BEGIN NEW(LSP,POINTER);                                               COMP     1274
000032         WITH LSP↑ DO                                                         COMP     1275
000034          BEGIN ELTYPE := NIL; FORM := POINTER; FTYPE := FALSE;               COMP     1276
000042           WITH SIZE DO                                                       COMP     1277
000043            BEGIN WORDS := 0; BITS := NROFBITS(MAXADDR) END                   COMP     1278
000054          END;                                                                COMP     1279
000054         INSYMBOL;                                                            COMP     1280
000055         IF SY = IDENT THEN                                                   COMP     1281
000057          BEGIN PRTERR := FALSE;   (*NO ERROR IF SEARCH NOT SUCCESSFUL*)      COMP     1282
000060           SEARCHID([TYPES],LCP); PRTERR := TRUE;                             COMP     1283
000063           IF LCP = NIL THEN   (*FORWARD REFERENCED TYPE ID*)                 COMP     1284
000065            BEGIN NEW(LCP,TYPES);                                             COMP     1285
000066             WITH LCP↑ DO                                                     COMP     1286
000070              BEGIN NAME := ID; IDTYPE := LSP; KLASS := TYPES;                COMP     1287
000076               NEXT := FWPTR                                                  COMP     1288
000076              END;                                                            COMP     1289
000100             FWPTR := LCP                                                     COMP     1290
000100            END                                                               COMP     1291
000100           ELSE                                                              COMP     1292
000101            BEGIN                                                            COMP     1293
000101             IF LCP↑.IDTYPE <> NIL THEN                                      COMP     1294
000103              IF LCP↑.IDTYPE↑.FTYPE THEN ERROR(108)                          COMP     1295
000105              ELSE LSP↑.ELTYPE := LCP↑.IDTYPE                                COMP     1296
000110            END;                                                             COMP     1297
000112           INSYMBOL;                                                         COMP     1298
000113          END                                                                COMP     1299
000113         ELSE ERROR(2);                                                      COMP     1300
000116        END                                                                  COMP     1301
000116       ELSE                                                                  COMP     1302
000117        BEGIN                                                                COMP     1303
000117         IF SY = PACKEDSY THEN                                               COMP     1304
000121          BEGIN PACKFLAG := TRUE; INSYMBOL END                               COMP     1305
000123         ELSE PACKFLAG := FALSE;                                             COMP     1306
000125         IF SY = SEGMENTEDSY THEN                                            COMP     1307
000127          BEGIN SEGFLAG := TRUE; INSYMBOL END                                COMP     1308
000131         ELSE SEGFLAG := FALSE;                                              COMP     1309
000133         IF NOT (SY IN TYPEDELS) THEN                                        COMP     1310
000136          BEGIN ERRCR(10); SKIP(FSYS+TYPEDELS) END;                          COMP     1311
000142         IF (SY <> FILESY)AND SEGFLAG THEN ERROR(57);                        COMP     1312
000147   (*ARRAY*)                                                                  COMP     1313
000147         IF SY = ARRAYSY THEN                                                COMP     1314
000151          BEGIN INSYMBOL;                                                    COMP     1315
000152           IF SY = LBRACK THEN INSYMBOL ELSE ERROR(11);                      COMP     1316
000157           LSP1 := NIL;                                                      COMP     1317
000160           (*LOOP UNTIL SY <> COMMA:*)                                       COMP     1318
000160           REPEAT NEW(LSP,ARRAYS);                                           COMP     1319
000162            WITH LSP↑ DO                                                     COMP     1320
000164             BEGIN AELTYPE := LSP1; INXTYPE := NIL;                          COMP     1321
000170              PCKDARR := PACKFLAG; FORM := ARRAYS;                           COMP     1322
000175              FTYPE := FALSE                                                 COMP     1323
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000175              END;                                                    COMP     1324
000177             LSP1 := LSP;                                             COMP     1325
000200             SIMPLETYPE(FSYS+[COMMA,RBRACK,OFSY],LSP2);               COMP     1326
000203             IF LSP2 <> NIL THEN                                      COMP     1327
000205              IF LSP2↑.FORM <= SUBRANGE THEN                          COMP     1328
000207               IF COMPTYPES(LSP2,REALPTR,FALSE)OR (LSP2 =             COMP     1329
000215                INTPTR) THEN ERROR(149)                               COMP     1330
000220               ELSE LSP↑.INXTYPE := LSP2                              COMP     1331
000223              ELSE ERROR(113);                                        COMP     1332
000227             EXITLOOP := SY <> COMMA;                                 COMP     1333
000232             IF NOT EXITLOOP THEN INSYMBOL                            COMP     1334
000233            UNTIL EXITLOOP;                                           COMP     1335
000235            IF SY = RBRACK THEN INSYMBOL ELSE ERROR(12);              COMP     1336
000242            IF SY = OFSY THEN INSYMBOL ELSE ERROR(8);                 COMP     1337
000250            TYP(FSYS,LSP);                                            COMP     1338
000253            (*REVERSE POINTERS, COMPUTE SIZE, SET PARTWORDELS AND     COMP     1339
000253             ELSPERWORD*)                                             COMP     1340
000253            IF LSP <> NIL THEN                                        COMP     1341
000255             BEGIN LSIZE := LSP↑.SIZE;                                COMP     1342
000257              REPEAT                                                  COMP     1343
000257               WITH LSP1↑ DO                                          COMP     1344
000261                BEGIN LSP2 := AELTYPE; AELTYPE := LSP;                COMP     1345
000264                 FTYPE := LSP↑.FTYPE;                                 COMP     1346
000267                 IF INXTYPE <> NIL THEN                               COMP     1347
000271                  BEGIN GETBOUNDS(INXTYPE,LMIN,LMAX);                 COMP     1348
000273                   IF (LSIZE.WORDS > 0)OR NOT PACKFLAG THEN           COMP     1349
000277                    BEGIN LSIZE.WORDS := FULLWORDS(LSIZE)*            COMP     1350
000304                                (LMAX-LMIN+1);                        COMP     1351
000311                     LSIZE.BITS := 0; PARTWORDELS := FALSE            COMP     1352
000313                    END                                               COMP     1353
000315                   ELSE                                               COMP     1354
000316                    BEGIN                                             COMP     1355
000316                     IF LSIZE.BITS > 0 THEN                           COMP     1356
000320                      T := WORDSIZE DIV LSIZE.BITS                    COMP     1357
000320                     ELSE T := 1;                                     COMP     1358
000325                     T1 := ( LMAX-LMIN+1 ) MOD T ;                    UPDATE7     2
000332                     IF (T1 = 0) AND (T*LSIZE.BITS < WORDSIZE) THEN T1:=T; UPDATE7  3
000337                     LSIZE.WORDS := ( LMAX-LMIN+1-T1 ) DIV T ;        UPDATE7     4
000346                     LSIZE.BITS := T1*LSIZE.BITS ;                    UPDATE7     5
000351                     IF T > 1 THEN                                    COMP     1362
000352                      BEGIN PARTWORDELS := TRUE;                      COMP     1363
000354                       ELSPERWORD := T                                COMP     1364
000354                      END                                             COMP     1365
000356                     ELSE PARTWORDELS := FALSE                        COMP     1366
000360                    END                                               COMP     1367
000362                  END;                                                COMP     1368
000362                 SIZE := LSIZE                                        COMP     1369
000363                END (*WITH LSP1↑*) ;                                  COMP     1370
000365               LSP := LSP1; LSP1 := LSP2                              COMP     1371
000366              UNTIL LSP1 = NIL                                        COMP     1372
000367             END (*LSP <> NIL*)                                       COMP     1373
000371            END                                                       COMP     1374
000371           ELSE                                                       COMP     1375
000372    (*RECORD*)                                                        COMP     1376
000372           IF SY = RECORDSY THEN                                      COMP     1377
000374            BEGIN INSYMBOL;                                           COMP     1378
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A 22

```
000375              OLDTOP := TOP;                                                  COMP  1379
000377              IF TOP < DISPLIMIT THEN                                         COMP  1380
000400               BEGIN TOP := TOP + 1;                                          COMP  1381
000401                WITH DISPLAY[TOP] DO                                          COMP  1382
000402                 BEGIN FNAME := NIL; OCCUR := REC END                         COMP  1383
000407               END                                                            COMP  1384
000407              ELSE ERROR(250);                                                COMP  1385
000412              WITH DISPL DO                                                   COMP  1386
000412               BEGIN WORDS := 0; BITS := 0 END;                               COMP  1387
000417              FIELDLIST(FSYS-[SEMICOLON]+[ENDSY],LSP1,LCP,LFILTYP);           COMP  1388
000424              NEW(LSP,RECORDS);                                               COMP  1389
000426              WITH LSP↑ DO                                                    COMP  1390
000430               BEGIN FIELDS := DISPLAY[TOP].FNAME; FTYPE := LFILTYP;          COMP  1391
000436                FSTFLD := LCP; RECVAR := LSP1; SIZE := DISPL;                 COMP  1392
000445                PCKDREC := PACKFLAG; FORM := RECORDS                          COMP  1393
000447               END;                                                           COMP  1394
000451              TOP := OLDTOP;                                                  COMP  1395
000452              IF SY = ENDSY THEN INSYMBOL ELSE ERROR(13)                      COMP  1396
000457             END                                                              COMP  1397
000460            ELSE                                                             COMP  1398
000461  (*SET*)                                                                     COMP  1399
000461          IF SY = SETSY THEN                                                  COMP  1400
000463            BEGIN INSYMBOL;                                                   COMP  1401
000464             IF SY = OFSY THEN INSYMBOL ELSE ERROR(8);                        COMP  1402
000472             SIMPLETYPE(FSYS,LSP1);                                           COMP  1403
000475             IF LSP1 <> NIL THEN                                              COMP  1404
000477              IF LSP1↑.FORM > SUBRANGE THEN                                   COMP  1405
000501               BEGIN ERROR(115); LSP1 := NIL END                              COMP  1406
000504              ELSE                                                            COMP  1407
000505               IF LSP1 = REALPTR THEN ERROR(114)                              COMP  1408
000507               ELSE                                                           COMP  1409
000511                IF LSP1 = INTPTR THEN                                         COMP  1410
000512                 ERROR(169)                                                   COMP  1411
000513                ELSE                                                          COMP  1412
000515                 BEGIN GETBOUNDS(LSP1,LMIN,LMAX);                             COMP  1413
000520                  IF (LMIN < 0)OR (LMAX > 58) THEN ERROR(169);                COMP  1414
000525                  (*IMPLEMENTATION RESTRICTION TO ONE WORD SETS*)              COMP  1415
000525                  NEW(LSP,POWER);                                             COMP  1416
000527                  WITH LSP↑, SIZE DO                                          COMP  1417
000531                   BEGIN ELSET := LSP1; PCKDSET := PACKFLAG;                  COMP  1418
000536                    FORM := POWER; FTYPE := FALSE;                            COMP  1419
000542                    IF LMAX >= 58 THEN                                        COMP  1420
000544                     BEGIN WORDS := 1; BITS := 0 END                          COMP  1421
000550                    ELSE                                                      COMP  1422
000551                     BEGIN WORDS := 0; BITS := LMAX + 1 END                   COMP  1423
000556                   END                                                        COMP  1424
000556                 END                                                          COMP  1425
000556            END                                                               COMP  1426
000556          ELSE                                                                COMP  1427
000557  (*FILE*) IF SY = FILESY THEN                                                COMP  1428
000561            BEGIN INSYMBOL;                                                   COMP  1429
000562             IF SY = OFSY THEN INSYMBOL ELSE ERROR(8);                        COMP  1430
000570             TYP(FSYS,LSP1);                                                  COMP  1431
000573             (*COMPUTE MACHINE AND IMPLEMENTATION DEPENDANT                   COMP  1432
000573             FILE SIZE*)                                                      COMP  1433
000573             IF LSP1 <> NIL THEN                                              COMP  1434
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000575                BEGIN LRL := FULLWORDS(LSP1↑.SIZE);                                  COMP     1435
000602                 IF LRL <= 1 THEN LRL := 1                                           COMP     1436
000603                END                                                                  COMP     1437
000604               ELSE LRL := 1;                                                        COMP     1438
000606               T1 := ((128*BUFFAC + LRL - 1) DIV LRL + 1)*LRL;                       COMP     1439
000614               NEW(LSP,FILES);                                                       COMP     1440
000616               WITH LSP↑ DO                                                          COMP     1441
000620                BEGIN FILTYPE := LSP1; FORM := FILES; FTYPE := TRUE;                 COMP     1442
000627                 SEGFILE := SEGFLAG;                                                 COMP     1443
000631                 TEXTFILE := (*PACKFLAG AND *)COMPTYPES(CHARPTR,LSP1,                COMP     1444
000634                       FALSE);                                                       COMP     1445
000641                 IF TEXTFILE THEN PCKDFIL := TRUE                                    COMP     1446
000643                 ELSE PCKDFIL := PACKFLAG;                                           COMP     1447
000650                 WITH SIZE DO                                                        COMP     1448
000651                  BEGIN                                                              COMP     1449
000651                   IF TEXTFILE THEN WORDS := T1 + CHEFETSZ                           COMP     1450
000653                   ELSE WORDS := T1 + EFETSZ;                                        COMP     1451
000663                   BITS := 0                                                         COMP     1452
000664                  END                                                                COMP     1453
000666                END;                                                                 COMP     1454
000666               IF LSP1 <> NIL THEN                                                   COMP     1455
000670                IF LSP1↑.FTYPE THEN                                                  COMP     1456
000672                 BEGIN ERROR(108); LSP↑.FILTYPE := NIL END;                          COMP     1457
000676              END;                                                                   COMP     1458
000676         END;                                                                        COMP     1459
000676       IF NOT (SY IN FSYS) THEN                                                      COMP     1460
000701         BEGIN ERROR(6); SKIP(FSYS) END                                              COMP     1461
000704       END;                                                                          COMP     1462
000704      FSP := LSP                                                                     COMP     1463
000705    END (*TYP*) ;                                                                    COMP     1464
000714                                                                                     COMP     1465
000003    PROCEDURE LABELDECLARATION;                                                      COMP     1466
000003     LABEL 1;                                                                        COMP     1467
000003     VAR LLP: LBP; EXITLOOP: BOOLEAN;                                                COMP     1468
000005    BEGIN                                                                            COMP     1469
000005     (*LOOP UNTIL SY <> COMMA:*)                                                     COMP     1470
000005     REPEAT                                                                          COMP     1471
000006      IF SY = INTCONST THEN                                                          COMP     1472
000010       BEGIN LLP := FSTLABP;                                                         COMP     1473
000011        WHILE LLP <> FLABP DO                                                        COMP     1474
000014         IF LLP↑.LABVAL = IVAL THEN                                                  COMP     1475
000016          BEGIN ERROR(166); GOTO 1 END                                               COMP     1476
000020         ELSE LLP := LLP↑.NEXTLAB;                                                   COMP     1477
000023        NEW(LLP);                                                                    COMP     1478
000025        WITH LLP↑ DO                                                                 COMP     1479
000027         BEGIN LABVAL := IVAL; DEFINED := FALSE; NEXTLAB := FSTLABP;                 COMP     1480
000035          LABLEV := LEVEL; LCNT := 0; FSTOCC := NIL                                  COMP     1481
000041         END;                                                                        COMP     1482
000043        FSTLABP := LLP;                                                              COMP     1483
000043   18   INSYMBOL ;                                                                   UPDAT10     3
000044        IF OP = MUL THEN                                                             UPDAT10     4
000046         BEGIN  IF TRAPSET THEN ERROR( 178 ) ;                                       UPDAT10     5
000051          TRAPSET := TRUE ;  TRAPLAB := LLP↑.LABVAL ;                                UPDAT10     6
000054          TRADD := LC ;  LC := LC+1 ;                                                UPDAT10     7
000056          INSYMBOL ;                                                                 UPDAT10     8
000057         END                                                                         UPDAT10     9
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A.

```
000057       END                                                          COMP  1485
000057      ELSE ERROR(15);                                               COMP  1486
000062      IF NOT (SY IN FSYS+[COMMA,SEMICOLON]) THEN                    COMP  1487
000066       BEGIN ERROR(6); SKIP(FSYS+[COMMA,SEMICOLON]) END;            COMP  1488
000073      EXITLOOP := SY <> COMMA;                                      COMP  1489
000076      IF NOT EXITLOOP THEN INSYMBOL                                 COMP  1490
000077     UNTIL EXITLOOP;                                                COMP  1491
000101     IF SY = SEMICOLON THEN INSYMBOL ELSE ERROR(14)                 COMP  1492
000105   END (*LABELDECLARATION*) ;                                       COMP  1493
000112                                                                    COMP  1494
000003   PROCEDURE CONSTDECLARATION;                                      COMP  1495
000003    VAR LCP: CTP; LSP: STP; LVALU: VALU;                            COMP  1496
000007   BEGIN                                                            COMP  1497
000007    IF SY <> IDENT THEN                                             COMP  1498
000007     BEGIN ERROR(2); SKIP(FSYS+[IDENT]) END;                        COMP  1499
000014    WHILE SY = IDENT DO                                             COMP  1500
000016     BEGIN NEW(LCP,KONST);                                          COMP  1501
000017      WITH LCP↑ DO                                                  COMP  1502
000021       BEGIN NAME := ID; IDTYPE := NIL; NEXT := NIL;                COMP  1503
000026        KLASS := KONST                                              COMP  1504
000026       END;                                                         COMP  1505
000030      INSYMBOL;                                                     COMP  1506
000031      IF OP = EQOP THEN INSYMBOL ELSE ERROR(16);                    COMP  1507
000037      CONSTANT(FSYS+[SEMICOLON],LSP,LVALU);                         COMP  1508
000044      ENTERID(LCP);                                                 COMP  1509
000046      LCP↑.IDTYPE := LSP; LCP↑.VALUES := LVALU;                     COMP  1510
000053      IF SY = SEMICOLON THEN                                        COMP  1511
000055       BEGIN INSYMBOL;                                              COMP  1512
000056        IF NOT (SY IN FSYS+[IDENT]) THEN                            COMP  1513
000062         BEGIN ERROR(6); SKIP(FSYS+[IDENT]) END                     COMP  1514
000066       END                                                          COMP  1515
000066      ELSE ERROR(14)                                                COMP  1516
000067     END                                                            COMP  1517
000070   END (*CONSTDECLARATION*) ;                                       COMP  1518
000075                                                                    COMP  1519
000003   PROCEDURE TYPEDECLARATION;                                       COMP  1520
000003    VAR LCP,LCP1,LCP2: CTP; LSP: STP;                               COMP  1521
000007   BEGIN                                                            COMP  1522
000007    IF SY <> IDENT THEN                                             COMP  1523
000007     BEGIN ERROR(2); SKIP(FSYS+[IDENT]) END;                        COMP  1524
000014    WHILE SY = IDENT DO                                             COMP  1525
000016     BEGIN NEW(LCP,TYPES);                                          COMP  1526
000017      WITH LCP↑ DO                                                  COMP  1527
000021       BEGIN NAME := ID; IDTYPE := NIL; KLASS := TYPES END;         COMP  1528
000027      INSYMBOL;                                                     COMP  1529
000030      IF OP = EQOP THEN INSYMBOL ELSE ERROR(16);                    COMP  1530
000036      TYP(FSYS+[SEMICOLON],LSP);                                    COMP  1531
000042      ENTERID(LCP);                                                 COMP  1532
000044      LCP↑.IDTYPE := LSP;                                           COMP  1533
000047      (*HAS ANY FORWARD REFERENCE BEEN SATISFIED:*)                  COMP  1534
000047      LCP1 := FWPTR;                                                COMP  1535
000050      WHILE LCP1 <> NIL DO                                          COMP  1536
000052       BEGIN                                                        COMP  1537
000052        IF LCP1↑.NAME = LCP↑.NAME THEN                              COMP  1538
000054         BEGIN                                                      COMP  1539
000054          WITH LCP↑ DO                                              COMP  1540
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A.

```
000055            BEGIN LCP1↑.IDTYPE↑.ELTYPE := IDTYPE;                          COMP     1541
000061             IF IDTYPE <> NIL THEN                                         COMP     1542
000063              IF IDTYPE↑.FTYPE THEN ERROR(108)                             COMP     1543
000065            END;                                                          COMP     1544
000066           IF LCP1 <> FWPTR THEN                                          COMP     1545
000070            LCP2↑.NEXT := LCP1↑.NEXT                                      COMP     1546
000070           ELSE FWPTR := LCP1↑.NEXT;                                      COMP     1547
000076          END;                                                           COMP     1548
000076         LCP2 := LCP1; LCP1 := LCP1↑.NEXT                                COMP     1549
000077        END;                                                            COMP     1550
000101       IF SY = SEMICOLON THEN                                           COMP     1551
000103        BEGIN INSYMBOL;                                                 COMP     1552
000104         IF NOT (SY IN FSYS+[IDENT]) THEN                               COMP     1553
000110          BEGIN ERROR(6); SKIP(FSYS+[IDENT]) END                        COMP     1554
000114        END                                                             COMP     1555
000114       ELSE ERROR(14)                                                   COMP     1556
000116      END;                                                              COMP     1557
000120     IF FWPTR <> NIL THEN                                               COMP     1558
000122      BEGIN  ERROR( 117 ) ;  LINECOUNT := LINECOUNT+1 ;                 SPACERR     1
000126       REPEAT WRITELN(= TYPE-ID =,FWPTR↑.NAME);                         COMP     1560
000141        FWPTR := FWPTR↑.NEXT                                            COMP     1561
000142       UNTIL FWPTR = NIL;                                               COMP     1562
000145      END                                                               COMP     1564
000145    END (*TYPEDECLARATION*) ;                                           COMP     1565
000152                                                                        COMP     1566
000003    PROCEDURE VARDECLARATION;                                           COMP     1567
000003     VAR LCP,NXT: CTP; LSP: STP; EXITLOOP: BOOLEAN;                     COMP     1568
000007    BEGIN NXT := NIL;                                                   COMP     1569
000007     REPEAT                                                             COMP     1570
000007      (*LOOP UNTIL SY <> COMMA:*)                                       COMP     1571
000007      REPEAT                                                            COMP     1572
000007       IF SY = IDENT THEN                                               COMP     1573
000011        BEGIN NEW(LCP,VARS);                                            COMP     1574
000012         WITH LCP↑ DO                                                   COMP     1575
000014          BEGIN NAME := ID; NEXT := NXT; KLASS := VARS;                 COMP     1576
000022           IDTYPE := NIL; VKIND := DRCT; VLEV := LEVEL                  COMP     1577
000026          END;                                                          COMP     1578
000030         ENTERID(LCP);                                                  COMP     1579
000031         NXT := LCP;                                                    COMP     1580
000032         INSYMBOL;                                                      COMP     1581
000033        END                                                             COMP     1582
000033       ELSE ERROR(2);                                                   COMP     1583
000036       IF NOT (SY IN FSYS+[COMMA,COLON]+TYPEDELS) THEN                  COMP     1584
000043        BEGIN ERROR(6); SKIP(FSYS+[COMMA,COLON,SEMICOLON]+TYPEDELS) END; COMP     1585
000051       EXITLOOP := SY <> COMMA;                                         COMP     1586
000054       IF NOT EXITLOOP THEN INSYMBOL                                    COMP     1587
000055      UNTIL EXITLOOP;                                                   COMP     1588
000057      IF SY = COLON THEN INSYMBOL ELSE ERROR(5);                        COMP     1589
000065      TYP(FSYS+[SEMICOLON]+TYPEDELS,LSP);                               COMP     1590
000072      WHILE NXT <> NIL DO                                               COMP     1591
000074       WITH  NXT↑ DO                                                    COMP     1592
000075        BEGIN IDTYPE := LSP; VADDR := LC;                               COMP     1593
000102         IF LSP <> NIL THEN LC := LC + FULLWORDS(LSP↑.SIZE);            COMP     1594
000110         NXT := NEXT                                                    COMP     1595
000110        END;                                                            COMP     1596
000113      IF SY = SEMICOLON THEN                                            COMP     1597
```

MOORE BUSINESS FORMS, INC., FORM 1413 PRINTED IN U.S.A. 22

```
000115      BEGIN INSYMBOL;                                                    COMP     1598
000116       IF NOT (SY IN FSYS+[IDENT]) THEN                                  COMP     1599
000122        BEGIN ERROR(6); SKIP(FSYS+[IDENT]) END                           COMP     1600
000126       END                                                               COMP     1601
000126      ELSE ERROR(14)                                                     COMP     1602
000130     UNTIL (SY <> IDENT)AND NOT (SY IN TYPEDELS);                        COMP     1603
000135     IF FWPTR <> NIL THEN                                               COMP     1604
000137      BEGIN  ERROR( 117 ) ;  LINECOUNT := LINECOUNT+1 ;                 SPACERR     2
000143       REPEAT WRITELN(= TYPE-ID =,FWPTR^.NAME);                          COMP     1606
000156        FWPTR := FWPTR^.NEXT                                             COMP     1607
000157       UNTIL FWPTR = NIL;                                                COMP     1608
000162      END;                                                               COMP     1610
000152    END (*VARDECLARATION*) ;                                             COMP     1611
000171                                                                         COMP     1612
000003    PROCEDURE PROCEDUREDECLARATION(FSY: SYMBOL);                         COMP     1613
000004     VAR OLDLEV: 0..MAXLEVEL; LSY: SYMBOL; LCP,LCP1: CTP; LSP: STP;      COMP     1614
000011       FORW: BOOLEAN; OLDTOP: DISPRANGE;                                 COMP     1615
000013       LLC,LCM: ADDRRANGE; DLOCP: LOCOFREF;                              COMP     1616
000016       LASTLISTNAME : ALFA ;                                             LISTCTRL   48
000017                                                                         COMP     1617
000003     PROCEDURE PARAMETERLIST(FSY: SETOFSYS; VAR FPAR: CTP);              COMP     1618
000005      VAR LCP,LCP1,LCP2,LCP3: CTP; LSP,LSP1,LSP2: STP; LKIND: DRCTINDRCT;COMP     1619
000015        DESCSZ,SZ: INTEGER;                                              COMP     1620
000017        PACKFLAG,EXITLOOP: BOOLEAN;                                      COMP     1621
000021     BEGIN LCP1 := NIL;                                                  COMP     1622
000010      IF NOT (SY IN FSY+[LPARENT]) THEN                                  COMP     1623
000013       BEGIN ERROR(7); SKIP(FSYS+FSY+[LPARENT]) END;                     COMP     1624
000021      IF SY = LPARENT THEN                                               COMP     1625
000023       BEGIN IF FORW THEN ERROR(119);                                    COMP     1626
000026        INSYMBOL;                                                        COMP     1627
000027        IF NOT (SY IN [IDENT,VARSY,PROCEDURESY,FUNCTIONSY]) THEN         COMP     1628
000032         BEGIN ERROR(7); SKIP(FSYS+[IDENT,RPARENT]) END;                 COMP     1629
000037        WHILE SY IN [IDENT,VARSY,PROCEDURESY,FUNCTIONSY] DO              COMP     1630
000042         BEGIN                                                           COMP     1631
000042          IF SY = PROCEDURESY THEN                                       COMP     1632
000043           BEGIN                                                         COMP     1633
000043            REPEAT INSYMBOL;                                             COMP     1634
000044             IF SY = IDENT THEN                                          COMP     1635
000046             BEGIN NEW(LCP,PROC,DECLARED,FORMAL);                        COMP     1636
000047               WITH LCP^ DO                                              COMP     1637
000051                BEGIN NAME := ID; IDTYPE := NIL; NEXT := LCP1;           COMP     1638
000057                 KLASS := PROC; PFDECKIND := DECLARED;                   COMP     1639
000063                 PFLEV := LEVEL; PFADDR := LC; PFKIND := FORMAL;         COMP     1640
000072                END;                                                     COMP     1641
000072               ENTERID(LCP);                                             COMP     1642
000073               LCP1 := LCP; LC := LC + 1;                                COMP     1643
000076               INSYMBOL                                                  COMP     1644
000076              END                                                        COMP     1645
000077             ELSE ERROR(2);                                              COMP     1646
000102             IF NOT (SY IN FSYS+[COMMA,SEMICOLON,RPARENT]) THEN          COMP     1647
000106              BEGIN ERROR(7); SKIP(FSYS+[COMMA,SEMICOLON,RPARENT]) END   COMP     1648
000114            UNTIL SY <> COMMA                                            COMP     1649
000114           END                                                           COMP     1650
000116          ELSE                                                           COMP     1651
000117           BEGIN LCP2 := LCP1; LSP := NIL;                               COMP     1652
000121            IF SY = FUNCTIONSY THEN                                      COMP     1653
```

MOORE BUSINESS FORMS, INC.
FORM 1413
PRINTED IN U.S.A.
22

```
000123          BEGIN                                                          COMP    1654
000123           REPEAT INSYMBOL;                                              COMP    1655
000124            IF SY = IDENT THEN                                           COMP    1656
000126             BEGIN NEW(LCP,FUNC,DECLARED,FORMAL);                        COMP    1657
000127              WITH LCP↑ DO                                               COMP    1658
000131               BEGIN NAME := ID; IDTYPE := NIL; NEXT := LCP1;             COMP    1659
000137                KLASS := FUNC; PFDECKIND := DECLARED;                     COMP    1660
000143                PFLEV := LEVEL; PFADDR := LC; PFKIND := FORMAL            COMP    1661
000150               END;                                                       COMP    1662
000152              ENTERID(LCP);                                               COMP    1663
000153              LCP1 := LCP; LC := LC + 1;                                  COMP    1664
000156              INSYMBOL                                                    COMP    1665
000156             END                                                          COMP    1666
000157            ELSE ERROR(2);                                               COMP    1667
000162            IF NOT (SY IN [COMMA,COLON]+FSYS) THEN                       COMP    1668
000166             BEGIN ERROR(7); SKIP(FSYS+[COMMA,SEMICOLON,RPARENT])        COMP    1669
000173             END                                                          COMP    1670
000174           UNTIL SY <> COMMA;                                             COMP    1671
000176           IF SY = COLON THEN                                             COMP    1672
000200            BEGIN INSYMBOL;                                               COMP    1673
000201             IF SY = IDENT THEN                                           COMP    1674
000203              BEGIN SEARCHID([TYPES],LCP);                                COMP    1675
000205               LSP := LCP↑.IDTYPE;                                        COMP    1676
000207               IF LSP <> NIL THEN                                         COMP    1677
000211                IF NOT (LSP↑.FORM IN [SCALAR,SUBRANGE,POINTER])           COMP    1678
000213                 THEN BEGIN ERROR(120); LSP := NIL END;                   COMP    1679
000216               INSYMBOL                                                   COMP    1680
000216              END                                                         COMP    1681
000217               ELSE ERROR(2);                                             COMP    1682
000222               IF NOT (SY IN FSYS+[SEMICOLON,RPARENT]) THEN               COMP    1683
000226                BEGIN ERROR(7); SKIP(FSYS+[SEMICOLON,RPARENT]) END        COMP    1684
000234             END                                                          COMP    1685
000234            ELSE ERROR(5)                                                 COMP    1686
000236           END                                                            COMP    1687
000237          ELSE                                                            COMP    1688
000240           BEGIN                                                          COMP    1689
000240            IF SY = VARSY THEN                                            COMP    1690
000242             BEGIN LKIND := INDRCT; INSYMBOL END                          COMP    1691
000243            ELSE LKIND := DRCT;                                           COMP    1692
000245            (*LOOP UNTIL SY <> COMMA*)                                    COMP    1693
000245            REPEAT                                                        COMP    1694
000245             IF SY = IDENT THEN                                           COMP    1695
000247              BEGIN NEW(LCP,VARS);                                        COMP    1696
000250               WITH LCP↑ DO                                               COMP    1697
000252                BEGIN NAME := ID; IDTYPE := NIL; KLASS := VARS;           COMP    1698
000260                 VKIND := LKIND; NEXT := LCP1; VLEV := LEVEL;             COMP    1699
000266                 VADDR := LC                                              COMP    1700
000266                END;                                                      COMP    1701
000271               ENTERID(LCP);                                              COMP    1702
000272               LCP1 := LCP; LC := LC + 1;                                 COMP    1703
000275               INSYMBOL;                                                  COMP    1704
000276              END                                                         COMP    1705
000276             ELSE ERROR(2);                                               COMP    1706
000301             IF NOT (SY IN [COMMA,COLON]+FSYS) THEN                       COMP    1707
000305              BEGIN ERROR(7); SKIP(FSYS+[COMMA,SEMICOLON,RPARENT])        COMP    1708
000312              END;                                                        COMP    1709
```

MOORE BUSINESS FORMS, INC. FORM 213 PRINTED IN U.S.A 22

```
000313              EXITLOOP := SY <> COMMA;                                    COMP     1710
000316              IF NOT EXITLOOP THEN INSYMBOL                               COMP     1711
000317             UNTIL EXITLOOP;                                              COMP     1712
000321             IF SY = COLON THEN                                           COMP     1713
000323              BEGIN INSYMBOL;                                             COMP     1714
000324                IF SY = IDENT THEN                                        COMP     1715
000326                 BEGIN SEARCHID([TYPES],LCP);                             COMP     1716
000330                  LSP := LCP↑.IDTYPE;                                     COMP     1717
000332                  INSYMBOL                                                COMP     1718
000332                 END                                                      COMP     1719
000333                ELSE ERROR(2);                                            COMP     1720
000336               IF LSP <> NIL THEN                                         COMP     1721
000340                IF ( LKIND = DRCT ) AND LSP↑.FTYPE THEN                   UPDATE8    2
000343                 ERROR(121);                                              COMP     1723
000345               IF NOT (SY IN FSYS+[SEMICOLON,RPARENT]) THEN               COMP     1724
000351                BEGIN ERROR(7); SKIP(FSYS+[SEMICOLON,RPARENT]) END        COMP     1725
000357              END                                                         COMP     1726
000357             ELSE ERROR(5);                                               COMP     1727
000362            END;                                                          COMP     1728
000362           LCP3 := LCP1;                                                  COMP     1729
000364           WHILE LCP3 <> LCP2 DO                                          COMP     1730
000366            BEGIN LCP3↑.IDTYPE := LSP; LCP3 := LCP3↑.NEXT END             COMP     1731
000371          END;                                                            COMP     1732
000372         IF SY = SEMICOLON THEN                                           COMP     1733
000374          BEGIN INSYMBOL;                                                 COMP     1734
000375            IF NOT (SY IN FSYS+[IDENT,VARSY,PROCEDURESY,FUNCTIONSY]) THEN COMP     1735
000401             BEGIN ERROR(7); SKIP(FSYS+[IDENT,RPARENT]) END               COMP     1736
000407           END                                                            COMP     1737
000407          END (*WHILE*) ;                                                 COMP     1738
000410         IF SY = RPARENT THEN                                             COMP     1739
000412          BEGIN INSYMBOL;                                                 COMP     1740
000413           IF NOT (SY IN FSY+FSYS) THEN                                   COMP     1741
000417            BEGIN ERROR(6); SKIP(FSY+FSYS) END                            COMP     1742
000425          END                                                             COMP     1743
000425         ELSE ERROR(4);                                                   COMP     1744
000430         LCP3 := NIL;                                                     COMP     1745
000431         (*REVERSE POINTERS AND RESERVE LOCAL CELLS FOR COPIES OF         COMP     1746
000431          EITHER MULTIPLE VALUES OR THEIR DESCRIPTORS*)                   COMP     1747
000431         WHILE LCP1 <> NIL DO                                             COMP     1748
000433          WITH LCP1↑ DO                                                   COMP     1749
000434           BEGIN LCP2 := NEXT; NEXT := LCP3;                              COMP     1750
000437            IF KLASS = VARS THEN                                          COMP     1751
000441             IF IDTYPE <> NIL THEN                                        COMP     1752
000442             WITH IDTYPE↑ DO                                              COMP     1753
000443                BEGIN SZ := FULLWORDS(SIZE);                              COMP     1754
000447                 IF (VKIND = DRCT)AND (SZ <> 1) THEN                      COMP     1755
000454                  BEGIN VADDR := LC; LC := LC + SZ                        COMP     1756
000456                  END                                                     COMP     1757
000457                END;                                                      COMP     1758
000457            LCP3 := LCP1; LCP1 := LCP2                                    COMP     1759
000461           END;                                                           COMP     1760
000462         FPAR := LCP3                                                     COMP     1761
000463        END                                                               COMP     1762
000464         ELSE FPAR := NIL                                                 COMP     1763
000465   END (*PARAMETERLIST*) ;                                                COMP     1764
000474                                                                          COMP     1765
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000474   BEGIN (*PROCEDUREDECLARATION*)                                          COMP     1766
000474    LLC := LC; LC := 3; FORW := FALSE; DP := TRUE;                         COMP     1767
000013    LINEADDR := LC ;  LASTLISTNAME := LISTNAME ;  LISTNAME := ID ;         LISTCTRL   49
000016    IF SY = IDENT THEN                                                     COMP     1768
000020     BEGIN SEARCHSECTION(DISPLAY[TOP].FNAME,LCP); (*DECIDE WHETHER FORW*)  COMP     1769
000023      IF LCP <> NIL THEN                                                   COMP     1770
000025      BEGIN                                                                COMP     1771
000025       IF LCP↑.KLASS = PROC THEN                                           COMP     1772
000030        FORW := (LCP↑.PFDECL = FORWDECL)AND (FSY = PROCEDURESY)            COMP     1773
000033            AND (LCP↑.PFKIND = ACTUAL)                                     COMP     1774
000036       ELSE                                                                COMP     1775
000041        IF LCP↑.KLASS = FUNC THEN                                          COMP     1776
000043         FORW := (LCP↑.PFDECL = FORWDECL)AND (FSY = FUNCTIONSY)            COMP     1777
000046             AND (LCP↑.PFKIND = ACTUAL)                                     COMP     1778
000051        ELSE FORW := FALSE;                                                 COMP     1779
000055       IF NOT FORW THEN ERROR(160)                                         COMP     1780
000057      END                                                                  COMP     1781
000060      ELSE FORW := FALSE;                                                  COMP     1782
000062      IF NOT FORW THEN                                                     COMP     1783
000064       BEGIN                                                               COMP     1784
000064        IF FSY = PROCEDURESY THEN                                          COMP     1785
000066         BEGIN NEW(LCP,PROC,DECLARED,ACTUAL); LCP↑.KLASS := PROC END       COMP     1786
000073        ELSE                                                               COMP     1787
000073         BEGIN NEW(LCP,FUNC,DECLARED,ACTUAL); LCP↑.KLASS := FUNC END;      COMP     1788
000101        WITH LCP↑ DO                                                       COMP     1789
000103         BEGIN NAME := ID; IDTYPE := NIL; NEXT := NIL;                     COMP     1790
000110          PFDECKIND := DECLARED; PFKIND := ACTUAL;                         COMP     1791
000114          PFDECL := DECL; PFLEV := LEVEL;                                  COMP     1792
000120          PCNT := PCNT + 1;                                                COMP     1793
000121          IF EXTON THEN PFCNT := 0 ELSE PFCNT := PCNT                      COMP     1794
000125         END;                                                              COMP     1795
000127        ENTERID(LCP)                                                       COMP     1796
000130       END                                                                 COMP     1797
000131      ELSE                                                                 COMP     1798
000132       BEGIN LCP1 := LCP↑.NEXT;                                            COMP     1799
000134        WHILE LCP1 <> NIL DO                                               COMP     1800
000136         BEGIN                                                             COMP     1801
000136          WITH LCP1↑ DO                                                    COMP     1802
000137           IF KLASS = VARS THEN                                            COMP     1803
000141           BEGIN                                                           UPDAT10    10
000141            IF IDTYPE <> NIL THEN                                          COMP     1804
000143             BEGIN LCM := VADDR + FULLWORDS(IDTYPE↑.SIZE);                 COMP     1805
000152              IF LCM > LC THEN LC := LCM                                   COMP     1806
000153             END;                                                          COMP     1807
000153           END                                                             UPDAT10    11
000153           ELSE                                                            UPDAT10    12
000154           BEGIN  LCM := PFADDR+1 ;                                        UPDAT10    13
000156            IF LCM > LC THEN LC := LCM ;                                   UPDAT10    14
000160           END ;                                                           UPDAT10    15
000160          LCP1 := LCP1↑.NEXT                                               COMP     1808
000161         END                                                               COMP     1809
000162        END;                                                               COMP     1810
000163      INSYMBOL                                                             COMP     1811
000163     END                                                                   COMP     1812
000164    ELSE                                                                   COMP     1813
000165     BEGIN ERROR(2); LCP := UFCTPTR END;                                   COMP     1814
```

MOORE BUSINESS FORMS, INC. F
FORM 113
PRINTED IN U.S.A
22

```
000171    OLDLEV := LEVEL; OLDTOP := TOP;                                    COMP  1815
000174    IF LEVEL < MAXLEVEL THEN LEVEL := LEVEL + 1 ELSE ERROR(251);       COMP  1816
000200    IF TOP < DISPLIMIT THEN                                            COMP  1817
000202     BEGIN TOP := TOP + 1;                                             COMP  1818
000203      WITH DISPLAY[TOP] DO                                             COMP  1819
000204       BEGIN                                                           COMP  1820
000204        IF FORW THEN FNAME := LCP↑.NEXT                                COMP  1821
000206        ELSE FNAME := NIL;                                             COMP  1822
000215        OCCUR := BLCK                                                  COMP  1823
000216       END                                                             COMP  1824
000220     END                                                               COMP  1825
000220    ELSE ERROR(250);                                                   COMP  1826
000223    IF FSY = PROCEDURESY THEN                                          COMP  1827
000225     BEGIN PARAMETERLIST([SEMICOLON],LCP1);                            COMP  1828
000230      IF NOT FORW THEN LCP↑.NEXT := LCP1                               COMP  1829
000232     END                                                               COMP  1830
000235    ELSE                                                               COMP  1831
000236     BEGIN PARAMETERLIST([SEMICOLON,COLON],LCP1);                      COMP  1832
000241      IF NOT FORW THEN LCP↑.NEXT := LCP1;                              COMP  1833
000246      IF SY = COLON THEN                                               COMP  1834
000250       BEGIN INSYMBOL;                                                 COMP  1835
000251        IF SY = IDENT THEN                                             COMP  1836
000253         BEGIN IF FORW THEN ERROR(122);                                COMP  1837
000255          SEARCHID([TYPES],LCP1);                                      COMP  1838
000257          LSP := LCP1↑.IDTYPE;                                         COMP  1839
000261          LCP↑.IDTYPE := LSP;                                          COMP  1840
000263          IF LSP <> NIL THEN                                           COMP  1841
000264           IF NOT (LSP↑.FORM IN [SCALAR,SUBRANGE,POINTER]) THEN        COMP  1842
000267            BEGIN ERROR(120); LCP↑.IDTYPE := NIL END;                  COMP  1843
000274          INSYMBOL                                                     COMP  1844
000274         END                                                           COMP  1845
000275        ELSE BEGIN ERROR(2); SKIP(FSYS+[SEMICOLON]) END                COMP  1846
000303       END                                                             COMP  1847
000303      ELSE                                                             COMP  1848
000304       IF NOT FORW THEN ERROR(123)                                     COMP  1849
000306     END;                                                              COMP  1850
000307    IF SY = SEMICOLON THEN INSYMBOL ELSE ERROR(14);                    COMP  1851
000314     IF (SY = IDENT)AND ((ID = ΞFORWARD   Ξ)OR (ID = ΞEXTERN    Ξ)      COMP  1852
000317      OR (ID = ΞFORTRAN   Ξ)) THEN                                     COMP  1853
000325      BEGIN IF FORW THEN ERROR(161);                                   COMP  1854
000327       WITH LCP↑ DO                                                    COMP  1855
000331        BEGIN                                                          COMP  1856
000331         IF ID = ΞFORWARD   Ξ THEN PFDECL := FORWDECL                   COMP  1857
000333         ELSE                                                          COMP  1858
000336          IF ID = ΞEXTERN    Ξ THEN PFDECL := EXTDECL                   COMP  1859
000340          ELSE PFDECL := FTNDECL;                                      COMP  1860
000346         IF PFDECL IN [EXTDECL,FTNDECL] THEN PFCNT := 0                COMP  1861
000352        END;                                                           COMP  1862
000354      INSYMBOL;                                                        COMP  1863
000355      IF SY = SEMICOLON THEN INSYMBOL ELSE ERROR(14);                  COMP  1864
000362      IF NOT (SY IN FSYS) THEN                                         COMP  1865
000365       BEGIN ERROR(6); SKIP(FSYS) END                                  COMP  1866
000372     END                                                               COMP  1867
000372    ELSE                                                               COMP  1868
000373     BEGIN LCP↑.PFDECL := DECL;                                        COMP  1869
000377      NEW(DLOCP);                                                      COMP  1870
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000400       REPEAT BLOCK(FSYS,SEMICOLON,LCP);                                  COMP      1871
000404        IF SY = SEMICOLON THEN                                            COMP      1872
000406         BEGIN INSYMBOL;                                                  COMP      1873
000407          IF NOT (SY IN [BEGINSY,PROCEDURESY,FUNCTIONSY]) THEN            COMP      1874
000412            BEGIN ERROR(6); SKIP(FSYS) END                                COMP      1875
000416         END                                                              COMP      1876
000416        ELSE ERROR(14)                                                    COMP      1877
000417       UNTIL SY IN [BEGINSY,PROCEDURESY,FUNCTIONSY];                      COMP      1878
000423       RELEASE(DLOCP);                                                    COMP      1879
000424      END;                                                                COMP      1880
000424     LEVEL := OLDLEV; TOP := OLDTOP; LC := LLC;                           COMP      1881
000430     LISTNAME := LASTLISTNAME ;                                           LISTCTRL    50
000432    END (*PROCEDUREDECLARATION*) ;                                        COMP      1882
000443                                                                          COMP      1883
000003    PROCEDURE BODY(FSYS: SETOFSYS);                                       COMP      1884
000004     TYPE OPRANGE = 0..63;                                                COMP      1885
000004        RCODERANGE = 0..RCODEMAX; RELRANGE= 0..2;                         COMP      1886
000004        CLOSEP = ↑ CLOSEREC;                                              COMP      1887
000004        CLOSEREC = PACKED RECORD NXTP: CLOSEP;                            COMP      1888
000004                     EFETADDR: ADDRRANGE                                  COMP      1889
000004                  END;                                                    COMP      1890
000004                                                                          COMP      1891
000004                  (*TO DESCRIBE EXPRESSION CURRENTLY COMPILED*)            COMP      1892
000004                  (*************************************)                 COMP      1893
000004                                                                          COMP      1894
000004        ATTRKIND = (CST,VARBL,COND,EXPR);                                 COMP      1895
000004        REGKIND = (NONE,XREG);                                            COMP      1896
000004        REGNR = 0..7;                                                     COMP      1897
000004                                                                          COMP      1898
000004        ATTR = RECORD TYPTR: STP;                                         COMP      1899
000004            CASE KIND: ATTRKIND OF                                        COMP      1900
000004             CST:   (CVAL: VALU);                                         COMP      1901
000004             VARBL: (WORDACC: ACCESSKIND;                                 COMP      1902
000004                     VLEVEL: LEVRANGE; CWDISPL: SHRTINT;                  COMP      1903
000004                     VWDISPL: REGNR;                                      COMP      1904
000004                     CASE PCKD: BOOLEAN OF                                COMP      1905
000004                      TRUE: (CBDISPL: SHRTINT;                            COMP      1906
000004                             BITREG: REGKIND; VBDISPL: REGNR));           COMP      1907
000004             COND:  (CDR: REGNR; CONDCD: (ZR,NZ,PL,NG));                  COMP      1908
000004             EXPR:  (EXPREG: REGNR)                                       COMP      1909
000004            END;                                                          COMP      1910
000004                                                                          COMP      1911
000004                                                                          COMP      1912
000004                (*TO DESCRIBE REGISTER STATUS*)                           COMP      1913
000004                (*****************************)                         COMP      1914
000004                                                                          COMP      1915
000004        ARGSTR = (SIMPADDR,INDADDR,UNSPECADDR);                           COMP      1916
000004        XRGSTR = (AVAIL,SHRTCST,LONGCST,SIMPVAR,INDVAR,OTHER);            COMP      1917
000004       (*REMXRG = SHRTCST..INDVAR; AFTER BOOTSTRAP*)                      COMP      1918
000004        BRGSTR = (FREE,BASADDR,SPECPURP);                                 COMP      1919
000004                                                                          COMP      1920
000004        ARGSTAT =                                                         COMP      1921
000004           PACKED RECORD CASE ACONT: ARGSTR OF                            COMP      1922
000004                  SIMPADDR: (ALEV: LEVRANGE; AADDR: ADDRRANGE);           COMP      1923
000004                  INDADDR:  (AREG: REGNR; ADISPL: ADDRRANGE)              COMP      1924
000004                 END;                                                     COMP      1925
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A.

```
000004                                                                  COMP    1926
000004      XRGSTAT =                                                   COMP    1927
000004        PACKED RECORD                                             COMP    1928
000004         CASE XCONT: XRGSTR OF                                    COMP    1929
000004          SHRTCST,LONGCST,                                        COMP    1930
000004          SIMPVAR,INDVAR,                                         COMP    1931
000004          OTHER:                                                  COMP    1932
000004           (REFNR: SHRTINT; LASTREF: CODERANGE;                   COMP    1933
000004            CASE REMXRCONT: XRGSTR (*LATER REMXRG*) OF (*TAGF NOT USED*) COMP 1934
000004            SHRTCST:                                              COMP    1935
000004              (CSTVAL: SHRTINT);                                  COMP    1936
000004            LONGCST:                                              COMP    1937
000004              (CPTR: CTAILP);                                     COMP    1938
000004            SIMPVAR,                                              COMP    1939
000004            INDVAR:                                               COMP    1940
000004              (SHFTCNT: BITRANGE;                                 COMP    1941
000004              CASE ADDRMODE: ACCESSKIND (*LATER DRCTINDRCT*) OF    COMP    1942
000004               DRCT:                                              COMP    1943
000004                (XLEV: LEVRANGE; XADDR: ADDRRANGE;                COMP    1944
000004                 VPADDR: BOOLEAN);                                COMP    1945
000004               INDRCT:                                            COMP    1946
000004                (XREG: REGNR; XDISPL: ADDRRANGE)))                COMP    1947
000004        END;                                                      COMP    1948
000004                                                                  COMP    1949
000004      BRGSTAT =                                                   COMP    1950
000004        PACKED RECORD CASE BCONT: BRGSTR OF                       COMP    1951
000004              BASADDR: (BLEV: LEVRANGE)                           COMP    1952
000004              END;                                                COMP    1953
000004                                                                  COMP    1954
000004        ARGSTATUS = ARRAY [REGNR] OF ARGSTAT;                     COMP    1955
000004        XRGSTATUS = ARRAY [REGNR] OF XRGSTAT;                     COMP    1956
000004        BRGSTATUS = ARRAY [REGNR] OF BRGSTAT;                     COMP    1957
000004        BASREGS = ARRAY [LEVRANGE] OF REGNR;                      COMP    1958
000004                                                                  COMP    1959
000004                                                                  COMP    1960
000004        CSTKIND = (NOP,PUREP,POSP,NEGP);                          COMP    1961
000004        CSTREC = PACKED RECORD                                    COMP    1962
000004             CASE CKIND : CSTKIND OF                              COMP    1963
000004              PUREP: (EXP: BITRANGE);                             COMP    1964
000004              POSP,                                               COMP    1965
000004              NEGP:  (EXP1,EXP2: BITRANGE)                        COMP    1966
000004             END;                                                 COMP    1967
000004                                                                  COMP    1968
000004                                                                  COMP    1969
000004    VAR ARGS: ARGSTATUS; XRGS: XRGSTATUS; BRGS: BRGSTATUS;        COMP    1970
000044      BRG: BASREGS;                                               COMP    1971
000057      LEVELS: SET OF LEVRANGE;                                    COMP    1972
000060      BONUS: ARRAY [SHRTCST..INDVAR] OF INTEGER;                  COMP    1973
000064                                                                  COMP    1974
000064                                                                  COMP    1975
000064      LCP: CTP; LFSTOCC: LOCOFREF;                                COMP    1976
000066      FCLSP,CLSP: CLOSEP; LP: CTAILP;                             COMP    1977
000071      I: REGNR;                                                   COMP    1978
000072      CODE: ARRAY[CODERANGE] OF INTEGER;                          COMP    1979
001533      CIX,LCIX,LCIX1,LCIX2: CODERANGE; CP,LCPT,LCP1,LCP2: POSRANGE; UPDAT10  16
001543    CBUF : INTEGER ;                                              UPDAT10  17
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
001544      LCMAX,LDISP,LSZ: ADDRRANGE;                                          COMP    1981
001547      RCODE: ARRAY[RCODERANGE] OF INTEGER;                                 COMP    1982
001636      RCIX: RCODERANGE; RCP: 1..15; RBUF: INTEGER;                         COMP    1983
001641      LASTOP: OPRANGE;   LASTI: REGNR;                                     COMP    1984
001643      LOCP: LOCOFREF; LCSP: CSP;                                           COMP    1985
001645      GATTR: ATTR; LCD: INTEGER;                                           COMP    1986
001660      EXFILP: EXTFILEP; LOCFILES: BOOLEAN;                                 COMP    1987
001662                                                                           COMP    1988
001662                                                                           COMP    1989
000003    PROCEDURE NOOP;                                                        COMP    1990
000003    BEGIN LASTOP:= 46B;                                                    COMP    1991
000007     WHILE CP<4 DO BEGIN CBUF:= CBUF*1000000B+460000B;                     COMP    1992
000014           RBUF:= 2*RBUF; CP:= CP+1                                        COMP    1993
000015          END                                                              COMP    1994
000015    END (*NOOP*);                                                          COMP    1995
000022                                                                           COMP    1996
000003    PROCEDURE GEN15(FOP: OPRANGE; FI,FJ,FK: REGNR);                        COMP    1997
000007    BEGIN LASTOP:= FOP; LASTI:= FI;                                        COMP    1998
000012     IF CP<>4 THEN                                                         COMP    1999
000014     BEGIN CP:= CP+1; CBUF:= CBUF*64+FOP; RBUF:= RBUF*2;                   COMP    2000
000017     END ELSE                                                              COMP    2001
000020     BEGIN CODE[CIX]:= CBUF; CBUF:= FOP; CP:= 1;                           COMP    2002
000023      IC:= IC+1; CIX:= CIX+1;                                              COMP    2003
000025      IF RCP=15 THEN                                                       COMP    2004
000027      BEGIN RCODE[RCIX]:= RBUF; RBUF:= 0; RCP:= 1; RCIX:= RCIX+1;          COMP    2005
000032      END ELSE BEGIN RBUF:= 2*RBUF; RCP:= RCP+1 END;                       COMP    2006
000035      IF CIX>CODEMAX THEN                                                  COMP    2007
000040      BEGIN ERROR(253); CIX:= 0; RCIX:= 0;                                 COMP    2008
000044      END;                                                                 COMP    2009
000044     END;                                                                  COMP    2010
000044     CBUF:= ((8*CBUF+FI)*8+FJ)*8+FK;                                       COMP    2011
000050    END (*GEN15*) ;                                                        COMP    2012
000054                                                                           COMP    2013
000003    PROCEDURE GEN30(FOP: OPRANGE; FI,FJ: REGNR; FK: ADDRRANGE;             COMP    2014
000007     FR: RELRANGE);                                                        COMP    2015
000010     LABEL 1;                                                              COMP    2016
000010     VAR I,J,K: REGNR; MAXPR: CODERANGE;                                   COMP    2017
000014    BEGIN                                                                  COMP    2018
000014     IF FR=1 THEN BEGIN FR:= 0; IF FJ=0 THEN FR:= 2 END;                   COMP    2019
000014     IF CP<3 THEN                                                          COMP    2020
000017     BEGIN CBUF:= 64*CBUF+FOP; RBUF:= 4*RBUF+FR; CP:= CP+2;                COMP    2021
000023     END ELSE                                                              COMP    2022
000024     BEGIN                                                                 COMP    2023
000024      IF CP = 3 THEN (*REPLACE NOOP BY BXIXJ INSTRUCTION IF APPROPRIATE*)  COMP    2024
000025       BEGIN                                                               COMP    2025
000025        FOR I := 0 TO 5 DO (*FIND AVAILABLE X-REGISTER*)                   COMP    2026
000026         IF (XRGS[I].XCONT=AVAIL)AND (I<>FJ) THEN                          COMP    2027
000036          BEGIN MAXPR := 0;                                                COMP    2028
000036           FOR J := 6 TO 7 DO                                              COMP    2029
000037            WITH XRGS[J] DO                                                COMP    2030
000043             IF XCONT IN [SIMPVAR,INDVAR] THEN                             COMP    2031
000046              IF REFNR <> 0 THEN                                           COMP    2032
000050               BEGIN MAXPR := CIX; K := J END                              COMP    2033
000052              ELSE                                                         COMP    2034
000053               IF LASTREF > MAXPR THEN                                     COMP    2035
000055                BEGIN MAXPR := LASTREF; K := J END;                        COMP    2036
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A 22

```
000061           IF MAXPR > 0 THEN (*COPY X-K INTO X-I*)                    COMP     2037
000063            BEGIN GEN15(10B,I,K,0); XRGS[I] := XRGS[K];               COMP     2038
000074             WITH XRGS[K] DO                                          COMP     2039
000076              IF REFNR <> 0 THEN                                      COMP     2040
000100               BEGIN XCONT := OTHER;                                  COMP     2041
000103                WITH XRGS[I] DO                                       COMP     2042
000104                 BEGIN REFNR := 0; LASTREF := CIX END                 COMP     2043
000111               END                                                    COMP     2044
000111              ELSE )CONT := AVAIL                                     COMP     2045
000111            END                                                       COMP     2046
000113           ELSE NOOP;                                                 COMP     2047
000116           GOTO 1                                                     COMP     2048
000117          END;                                                        COMP     2049
000121        NOOP;                                                         COMP     2050
000123   1: END (*CP = 3*) ;                                                COMP     2051
000123      CODE[CIX]:= CBUF; CBUF:= FOP; CP:= 2;                           COMP     2052
000127      IC:= IC+1; CIX:= CIX+1;                                         COMP     2053
000131      IF RCP=15 THEN                                                  COMP     2054
000133      BEGIN RCODE[RCIX]:= RBUF; RBUF:= FR; RCP:= 1; RCIX:= RCIX+1;    COMP     2055
000137      END ELSE BEGIN RBUF:= RBUF*4+FR; RCP:= RCP+1 END;               COMP     2056
000142      IF CIX>CODEMAX THEN                                             COMP     2057
000145      BEGIN ERROR(253); CIX:= 0; RCIX:= 0 END;                        COMP     2058
000151     END;                                                             COMP     2059
000151     LASTOP := FOP; LASTI := FI;                                      COMP     2060
000154     CBUF:= (8*CBUF+FI)*8+FJ;                                         COMP     2061
000156     IF FK>=0 THEN                                                    COMP     2062
000157     BEGIN CBUF:= CBUF*1000B*1000B+FK;                                COMP     2063
000151      IF EXT<>0 THEN                                                  COMP     2064
000152      BEGIN IF EXTRX=EXTRFMAX THEN BEGIN ERROR(256); EXTRX:=0 END;    COMP     2065
000156       EXTRX:= EXTRX+1; EXTRFS[EXTRX].LINK:= EXTID[EXT].LINK;         COMP     2066
000175       EXTRFS[EXTRX].LOC:= ((8-CP)*1000B+1)*1000B*1000B+IC-1;         COMP     2067
000205       EXTID[EXT].LINK:= EXTRX; EXT:= 0;                              COMP     2068
000211      END;                                                            COMP     2069
000211     END ELSE CBUF:= CBUF*1000B*1000B+777777B+FK;                     COMP     2070
000215    END (*GEN30*)  ;                                                  COMP     2071
000222                                                                      COMP     2072
000003    PROCEDURE INS(FIC: INTEGER; FLOC: CODERANGE; FPOS: POSRANGE);     COMP     2073
000006    BEGIN                                                             COMP     2074
000006     IF FIC < 0 THEN FIC := 7777778 + FIC;                            UPDATE4    2
000012     IF FLOC = CIX THEN FPOS := 4 - CP + FPOS;                        COMP     2075
000017     CASE FPOS OF                                                     COMP     2076
000021      1: FIC := FIC*1000000B*1000000B*1000000B;                       COMP     2077
000024      2: FIC := FIC*1000000B*1000000B;                                COMP     2078
000027      3: FIC := FIC*1000000B;                                         COMP     2079
000031      4:                                                              COMP     2080
000031     END;                                                             COMP     2081
000036     IF FLOC <> CIX THEN CODE[FLOC] := CODE[FLOC] + FIC               COMP     2082
000041     ELSE CBUF := CBUF + FIC                                          COMP     2083
000043    END (*INS*) ;                                                     COMP     2084
000052                                                                      COMP     2085
000003    PROCEDURE LINKOCC(VAR FPTR: LOCOFREF);                            COMP     2086
000004     VAR LOCP: LOCOFREF;                                              COMP     2087
000005    BEGIN NEW(LOCP);                                                  COMP     2088
000010     WITH LOCP↑ DO                                                    COMP     2089
000011      BEGIN NXTREF := FPTR; FPTR := LOCP;                             COMP     2090
000014       LOC := CIX; POS := CP                                          COMP     2091
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000017      END                                                                 COMP    2092
000022    END (*LINKOCC*) ;                                                     COMP    2093
000025                                                                          COMP    2094
000003    PROCEDURE GEN60(FC:INTEGER);                                          COMP    2095
000004    VAR I:SHRTINT;                                                        COMP    2096
000005    BEGIN NOOP;                                                           COMP    2097
000010     CODE[CIX]:= CBUF; IC:= IC+1; CIX:= CIX+1; CBUF:= FC;                 COMP    2098
000016     IF RCP=15 THEN                                                       COMP    2099
000020     BEGIN RCODE[RCIX]:= RBUF; RBUF:= 0; RCP:= 1; RCIX:= RCIX+1;          COMP    2100
000023     END ELSE BEGIN RBUF:= 16*RBUF; RCP:= RCP+1 END;                      COMP    2101
000025     IF CIX>CODEMAX THEN                                                  COMP    2102
000030     BEGIN ERROR(253); CIX:= 0; RCIX:= 0 END;                             COMP    2103
000034    END (*GEN60*);                                                        COMP    2104
000040                                                                          COMP    2105
000003    PROCEDURE PMDINFO(FCP: CTP);                                          COMP    2106
000004     VAR I: INTEGER;                                                      COMP    2107
000005    BEGIN                                                                 COMP    2108
000005     IF FCP <> NIL THEN                                                   COMP    2109
000010      WITH FCP↑ DO                                                        COMP    2110
000011       BEGIN PMDINFO(LLINK); PMDINFO(RLINK);                              COMP    2111
000016        IF KLASS = VARS THEN                                              COMP    2112
000021         IF IDTYPE <> NIL THEN                                            COMP    2113
000023          IF IDTYPE↑.FORM <= POINTER THEN                                 COMP    2114
000026           BEGIN ORDALFA(NAME,I); GEN60(I);                               COMP    2115
000033            IF IDTYPE↑.FORM = POINTER THEN I := 14                        COMP    2116
000037            ELSE                                                          COMP    2117
000040             IF COMPTYPES(IDTYPE,INTPTR,FALSE) THEN I := 2                COMP    2118
000045             ELSE                                                         COMP    2119
000046              IF COMPTYPES(IDTYPE,REALPTR,FALSE) THEN I := 4              COMP    2120
000053              ELSE                                                        COMP    2121
000054               IF COMPTYPES(IDTYPE,CHARPTR,FALSE) THEN I := 6             COMP    2122
000061               ELSE                                                       COMP    2123
000062                IF COMPTYPES(IDTYPE,BOOLPTR,FALSE) THEN I := 8            COMP    2124
000067                ELSE I := 10;                                             COMP    2125
000071           IF VKIND = INDRCT THEN I := I + 1;                             COMP    2126
000075           GEN60(I*1000000B + VADDR)                                      COMP    2127
000100          END                                                             COMP    2128
000102       END                                                                COMP    2129
000102    END (*PMDINFO*) ;                                                     COMP    2130
000106                                                                          COMP    2131
000003    PROCEDURE SHORTNAME(FNAME: ALFA; VAR F1NAME: ALFA);                   COMP    2132
000005     VAR I: 1..ALFALENG;                                                  COMP    2133
000006    BEGIN I := ALFALENG;                                                  COMP    2134
000010     IF FNAME <> Ξ          Ξ THEN                                        COMP    2135
000012      WHILE (FNAME[I] = Ξ Ξ)OR (I > 7) DO                                 COMP    2136
000021       BEGIN FNAME[I] := CHR(0); I := I - 1 END;                          COMP    2137
000026     F1NAME := FNAME                                                      COMP    2138
000027    END (*SHORTNAME*);                                                    COMP    2139
000034                                                                          COMP    2140
000003    PROCEDURE PFNAME(FI: INTEGER);                                        COMP    2141
000004     (*CREATE INTERNAL NAME FOR PROC/FUNC FROM FI*)                       COMP    2142
000004     VAR K,L: INTEGER;                                                    COMP    2143
000006    BEGIN                                                                 COMP    2144
000006     FOR K := 7 DOWNTO 4 DO                                               COMP    2145
000010      BEGIN L := FI DIV 8;                                                COMP    2146
000014       PNAME[K] := CHR(ORD(Ξ0Ξ) + FI - 8*L);                              COMP    2147
```

MOORE BUSINESS FORMS, INC. F
FORM 1413
PRINTED IN U.S.A
22

```
000021      FI := L                                                          COMP     2148
000021     END                                                               COMP     2149
000022   END (*PFNAME*);                                                     COMP     2150
000027                                                                       COMP     2151
000003   PROCEDURE LGOHEAD(NAME: ALFA);                                      COMP     2152
000004    TYPE STYPE= (WORD,ADRS,NAMS);                                      COMP     2153
000004       B18 = 0..777777B;                                               COMP     2154
000004    VAR PGNAME: INTEGER; I: INTEGER; LLP: LBP; LNAME: ALFA;            COMP     2155
000010     STRUCTURES: RECORD CASE C:STYPE OF (*NEVER SET NOR TESTED*)       COMP     2156
000010           WORD: (CVAL: INTEGER);                                      COMP     2157
000010           NAMS: (CNAM: ALFA);                                         COMP     2158
000010           ADRS: (IDW: PACKED RECORD CN: 0..63;                        COMP     2159
000010                    WC: B18; LR: B18; L: B18                           COMP     2160
000010              END)                                                     COMP     2161
000010         END;                                                          COMP     2162
000012   BEGIN                                                               COMP     2163
000012    WITH STRUCTURES DO                                                 COMP     2164
000007     BEGIN  CVAL := 0 ;  IDW.CN := 77B ;  IDW.WC := 16B ;  (*PRFX*)    XTNDPRFX    1
000015     LGO↑:= CVAL; PUT(LGO); SHORTNAME(NAME,CNAM); IDW.L:= 0;           COMP     2166
000023     PGNAME:= CVAL; LGO↑:= PGNAME; PUT(LGO);                           COMP     2167
000026      LNAME := Ξ          Ξ ;                                          XTNDPRFX    2
000030      FOR I := 2 TO 10 DO LNAME[ I-1 ] := LDATE[ I ] ;  (*DATE*)       XTNDPRFX    3
000043      ORDALFA( LNAME       ,LGO↑ ) ;  PUT( LGO ) ;                     XTNDPRFX    4
000047      FOR I := 2 TO 10 DO LNAME[ I-1 ] := LTIME[ I ] ;  (*TIME*)       XTNDPRFX    5
000062      ORDALFA( LNAME       ,LGO↑ ) ;  PUT( LGO ) ;                     XTNDPRFX    6
000066      ORDALFA( ΞSCOPE  3.4Ξ,LGO↑ ) ;  PUT( LGO ) ;      (*SYSTEM+VERS*)  XTNDPRFX    7
000072      ORDALFA( ΞPASCAL 2.0Ξ,LGO↑ ) ;  PUT( LGO ) ;      (*LANG+VERS*)    XTNDPRFX    8
000076      ORDALFA( ΞUP10 6 6  Ξ,LGO↑ ) ;  PUT( LGO ) ;      (*LANG.MOD*)     XTNDPRFX    9
000102      ORDALFA( Ξ I        Ξ,LGO↑ ) ;  PUT( LGO ) ;      (*HRDWRE DEP*)   XTNDPRFX   10
000106      FOR I := 1 TO 10 DO LNAME[ I ] := LSTSTR[ I ] ;   (*LANGUAGE*)     XTNDPRFX   11
000124      ORDALFA( LNAME       ,LGO↑ ) ;  PUT( LGO ) ;      (*PROCESSOR*)    XTNDPRFX   12
000130      FOR I := 1 TO 10 DO LNAME[ I ] := LSTSTR[ I+10 ] ;(*COMMENTS*)     XTNDPRFX   13
000147      ORDALFA( LNAME       ,LGO↑ ) ;  PUT( LGO ) ;      (*(OPTIONS*)     XTNDPRFX   14
000153      LNAME := Ξ          Ξ ;                           (*STRING)*)      XTNDPRFX   15
000155      FOR I := 1 TO 6 DO LNAME[ I ] := LSTSTR[ I+20 ] ;                  XTNDPRFX   16
000174      ORDALFA( LNAME       ,LGO↑ ) ;  PUT( LGO ) ;                     XTNDPRFX   17
000200      ORDALFA( LISTNAME    ,LGO↑ ) ;  PUT( LGO ) ;      (*USER*)         XTNDPRFX   18
000204      ORDALFA( Ξ           Ξ,LGO↑ ) ;  PUT( LGO ) ;     (*COMMENTS*)     XTNDPRFX   19
000210      ORDALFA( Ξ           Ξ,LGO↑ ) ;  PUT( LGO ) ;                     XTNDPRFX   20
000214      ORDALFA( Ξ           Ξ,LGO↑ ) ;  PUT( LGO ) ;                     XTNDPRFX   21
000220    CVAL := 0; IDW.CN:= 34B;                              (*PIDL*)     COMP     2169
000224    IF LEVEL = 1 THEN IDW.WC := 2 ELSE IDW.WC := 1;                    COMP     2170
000232    LGO↑:= CVAL; PUT(LGO); LGO↑:= PGNAME; PUT(LGO);                    COMP     2171
000240    IF LEVEL = 1 THEN                                                  COMP     2172
000242     BEGIN                                                             COMP     2173
000242      CNAM:=Ξ          Ξ; IDW.L:= 0; LGO↑:= CVAL; PUT(LGO);            COMP     2174
000246     END;                                                              COMP     2175
000246    CVAL := 0; IDW.CN:= 36B;                              (*ENTR*)     COMP     2176
000252    I := 0; (*COUNT NUMBER OF ADDITIONAL ENTRY POINTS*)                COMP     2177
000252    LLP := FSTLABP;                                                    COMP     2178
000253    WHILE LLP <> FLABP DO                                              COMP     2179
000256     WITH LLP↑ DO                                                      COMP     2180
000257      BEGIN IF LCNT <> 0 THEN I := I + 1;                              COMP     2181
000262       LLP := NEXTLAB                                                  COMP     2182
000262      END;                                                             COMP     2183
000265    IF (LEVEL = 1)AND (NAME <> ΞP.MAIN    Ξ) THEN                      COMP     2184
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000272      BEGIN IDW.WC := 2*(I + 2); LGO↑ := CVAL; PUT(LGO);                          COMP     2185
000277       LGO↑ := PGNAME; PUT(LGO); LGO↑ := 1000003B; PUT(LGO);                      COMP     2186
000305       SHORTNAME(=P.MAIN    =,LNAME); ORDALFA(LNAME,LGO↑); PUT(LGO);              COMP     2187
000314       LGO↑ := 1000003B; PUT(LGO)                                                 COMP     2188
000317      END                                                                         COMP     2189
000317     ELSE                                                                         COMP     2190
000320      BEGIN IDW.WC := 2*(I + 1); LGO↑ := CVAL; PUT(LGO);                          COMP     2191
000325       LGO↑ := PGNAME; PUT(LGO);                                                  COMP     2192
000330       IF LEVEL = 1 THEN LGO↑ := 1000003B ELSE LGO↑ := 1000002B;                  COMP     2193
000336       PUT(LGO)                                                                   COMP     2194
000337      END;                                                                        COMP     2195
000337     WHILE FSTLABP <> FLABP DO                                                    COMP     2196
000342      WITH FSTLABP↑ DO                                                            COMP     2197
000343       BEGIN                                                                      COMP     2198
000343        IF LCNT <> 0 THEN                                                         COMP     2199
000345         BEGIN PASCL[7] := CHR(LCNT);                                            COMP     2200
000350          ORDALFA(PASCL,LGO↑); PUT(LGO);                                         COMP     2201
000353          IF DEFINED THEN                                                        COMP     2202
000355           BEGIN CVAL := LABADDR; IDW.LR := 1; LGO↑ := CVAL;                     COMP     2203
000361            PUT(LGO)                                                             COMP     2204
000362           END                                                                    COMP     2205
000362          ELSE                                                                    COMP     2206
000363           BEGIN ERROR(168);                                                      COMP     2207
000365           WRITELN( = LABEL =,LABVAL ) ;  LINECOUNT := LINECOUNT+1 ;              SPACERR     3
000401           END                                                                    COMP     2209
000401         END                                                                      UPDATE9     4
000401        ELSE IF NOT DEFINED AND ( FSTOCC <> NIL ) THEN                            UPDATE9     5
000407              BEGIN  ERROR( 168 ) ;                                              UPDATE9     6
000410               WRITELN ;  WRITELN( = LABEL=,LABVAL ) ;                           UPDATE9     7
000423              END ;                                                              UPDATE9     8
000423        FSTLABP := NEXTLAB                                                       COMP     2211
000423       END;                                                                      COMP     2212
000426    END;                                                                         COMP     2213
000426   END (*LGOHEAD*);                                                             COMP     2214
000444                                                                                COMP     2215
000003   PROCEDURE LGOTEXT;                                                           COMP     2216
000003    TYPE B18 = 0..777777B;                                                      COMP     2217
000003    VAR J,DISP: 0..15;                                                          COMP     2218
000005      I: 1..RCODEMAX;                                                           COMP     2219
000006      L,LCIX: CODERANGE;                                                        COMP     2220
000010      STRUCTURES: RECORD CASE INTG: BOOLEAN OF (*NEVER SET NOR TESTED*)         COMP     2221
000010            TRUE: (CVAL: INTEGER);                                              COMP     2222
000010            FALSE: (IDW: PACKED RECORD CN: 0..63;                               COMP     2223
000010                              WC: B18; LR: B18; L: B18                          COMP     2224
000010                    END)                                                        COMP     2225
000010          END;                                                                  COMP     2226
000012   BEGIN LCIX:= 1;                                                              COMP     2227
000007    WITH STRUCTURES DO                                                          COMP     2228
000007     BEGIN NOOP;                                          (*TEXT*)             COMP     2229
000010     CODE[CIX]:= CBUF; IC:= IC+1;                                               COMP     2230
000014     WHILE RCP < 15 DO                                                          COMP     2231
000017     BEGIN RBUF := RBUF*16; RCP := RCP + 1 END;                                 COMP     2232
000021     RCODE[RCIX]:= RBUF;                                                        COMP     2233
000023     IDW.CN:= 40B; IDW.LR:= 1; DISP:= 15; IDW.WC:= 20B;                         COMP     2234
000033     FOR I:=1 TO RCIX DO                                                        COMP     2235
000034     BEGIN IF I=RCIX THEN                                                       COMP     2236
```

MOORE BUSINESS FORMS, INC. F
FORM 1413
PRINTED IN U.S.A

```
000040      BEGIN J:= CIX MOD 15;                                                COMP  2237
000044       IF J<>0 THEN BEGIN DISP:= J; IDW.WC:= J+1 END                       COMP  2238
000050      END;                                                                 COMP  2239
000050      IDW.L :=  CADDR; LGO↑:= CVAL; PUT(LGO);                              COMP  2240
000054      LGO↑:= RCODE[I]; PUT(LGO);                                           COMP  2241
000060      FOR L:= LCIX TO LCIX+DISP-1 DO                                       COMP  2242
000063      BEGIN LGO↑:= CODE[L]; PUT(LGO) END;                                  COMP  2243
000073      CADDR:= CADDR + DISP; LCIX:= LCIX + DISP;                            COMP  2244
000076     END (*FOR I*);                                                        COMP  2245
000100    END;                                                                   COMP  2246
000100   END (*LGOTEXT*);                                                        COMP  2247
000104                                                                           COMP  2248
000003   PROCEDURE LGOEND;                                                       COMP  2249
000003    TYPE STYP= (WORD,ADRS,HLFS,NAMS,EXTS);                                 COMP  2250
000003       B18 = 0..777777B; B30 = 0..7777777777B;                             COMP  2251
000003       HALFS= PACKED RECORD LH: B30; RH: B30 END;                          COMP  2252
000003    VAR PAR: BOOLEAN;                                                      COMP  2253
000004       STRUCTURES: RECORD CASE C: STYP OF  (*NEVER SET NOR TESTED*)        COMP  2254
000004              WORD: (CVAL: INTEGER);                                       COMP  2255
000004              HLFS: (HS: HALFS);                                           COMP  2256
000004              ADRS: (IDW: PACKED RECORD CN: 0..63;                         COMP  2257
000004                           WC: B18; LR: B18; L: B18                        COMP  2258
000004                    END);                                                  COMP  2259
000004              NAMS: (CNAM: ALFA);                                          COMP  2260
000004              EXTS: (EXTIDS: EXTIDREC)                                     COMP  2261
000004             END;                                                          COMP  2262
000007       BUFF: RECORD CASE INTG: BOOLEAN OF (*NEVER SET NOR TESTED*)         COMP  2263
000007              TRUE:  (BUF0: INTEGER);                                      COMP  2264
000007              FALSE: (BHS: HALFS)                                          COMP  2265
000007            END;                                                           COMP  2266
000011       XR: 0..EXTRFMAX; I: EXTIDRANGE;                                     COMP  2267
000013    BEGIN CODEADDR:= CODEADDR + IC;                                        COMP  2268
000007     WITH STRUCTURES,BUFF DO                                               COMP  2269
000007     BEGIN IF EXTRX<>0 THEN                                  (*LINK*)      COMP  2270
000011      BEGIN CVAL:= 0; IDW.CN:= 44B; IDW.WC:= EXTIDX+(EXTRX+1) DIV 2;       COMP  2271
000020       LGO↑:= CVAL; PUT(LGO); PAR:= TRUE;                                  COMP  2272
000023       FOR I:= 1 TO EXTIDX DO                                              COMP  2273
000025       BEGIN EXTIDS:= EXTID[I]; XR:= EXTIDS.LINK; IDW.L:= 0;               COMP  2274
000036        SHORTNAME(EXTIDS.NAME,EXTIDS.NAME);                                COMP  2275
000040        IF PAR THEN LGO↑:= CVAL ELSE                                       COMP  2276
000043        BEGIN BHS.RH:= HS.LH; LGO↑:= BUF0; BHS.LH:= HS.RH END;             COMP  2277
000052        PUT(LGO);                                                          COMP  2278
000053        WHILE XR <> 0 DO                                                   COMP  2279
000055         BEGIN IF PAR THEN BHS.LH:= EXTRFS[XR].LOC ELSE                    COMP  2280
000062          BEGIN BHS.RH:= EXTRFS[XR].LOC; LGO↑:= BUF0; PUT(LGO) END;        COMP  2281
000067          PAR:= NOT PAR; XR:= EXTRFS[XR].LINK;                             COMP  2282
000073         END;                                                              COMP  2283
000073       END;                                                                COMP  2284
000075       IF NOT PAR THEN                                                     COMP  2285
000077       BEGIN BHS.RH:= 0; LGO↑:= BUF0; PUT(LGO);                            COMP  2286
000103       END;                                                                COMP  2287
000103      END;                                                                 COMP  2288
000103      CVAL:= 0; IDW.CN:= 45B; IDW.WC:= 1;                     (*XFER*)    COMP  2289
000111      LGO↑:= CVAL; PUT(LGO);                                               COMP  2290
000112      IF LEVEL=1 THEN SHORTNAME(PROGNAME,CNAM)                             COMP  2291
000115      ELSE CNAM := Ξ           Ξ;                                          COMP  2292
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000122      IDW.L:= 0; LGO↑:= CVAL; PUT(LGO);                                  COMP     2293
000126      PUTSEG(LGO);                                                       COMP     2294
000127     END;                                                                COMP     2295
000127    END (*LGOEND*);                                                      COMP     2296
000134                                                                         COMP     2297
000003    PROCEDURE SEARCHEXTID(FNAME: ALFA; VAR FIX: EXTIDRANGE);             COMP     2298
000005     (*RETURN INDEX FIX OF FNAME-ENTRY IN EXTID*)                        COMP     2299
000005     LABEL 1;                                                            COMP     2300
000005     VAR I: EXTIDRANGE;                                                  COMP     2301
000006    BEGIN                                                                COMP     2302
000006     FOR I := 1 TO EXTIDX DO                                             COMP     2303
000011      IF EXTID[I].NAME = FNAME THEN                                      COMP     2304
000015       BEGIN FIX := I; GOTO 1 END;                                       COMP     2305
000020     IF EXTIDX = EXTIDMAX THEN ERROR(257)                                COMP     2306
000023     ELSE EXTIDX := EXTIDX + 1;                                          COMP     2307
000026     FIX := EXTIDX;                                                      COMP     2308
000030     WITH EXTID[EXTIDX] DO                                               COMP     2309
000032      BEGIN NAME := FNAME; LINK := 0 END;                                COMP     2310
000035   1:END (*SEARCHEXTID*) ;                                               COMP     2311
000041                                                                         COMP     2312
000003    PROCEDURE RJTOEXT;                                                   COMP     2313
000003    BEGIN IF CP = 3 THEN NOOP ;          (*PREVENT BXI XJ*)              UPDATE9    9
000011     GEN30( 018,0,0,0,0 ) ;  NOOP ;                                     UPDATE9   10
000016    END (*RJTOEXT*) ;                                                    COMP     2315
000022                                                                         RUNMODS    6
000003    PROCEDURE RJTORUNEXT( FI,FJ : INTEGER ) ;                            RUNMODS    7
000005    BEGIN  GEN30( 018,0,0,0,0 ) ;  GEN30( 078,FI,FJ,101B,0 ) ;          RUNMODS    8
000016    END ;                                                                RUNMODS    9
000022                                                                         COMP     2316
000003    PROCEDURE ENTERCST(FCSTP: CTAILP);                                   COMP     2317
000004     (*ENTER CONST POINTED AT BY FCSTP INTO CONSTANT TABLE AND CHAIN      COMP     2318
000004     ACTUAL OCCURENCE IN CODE (AT <CIX,CP>) WITH EARLIER OCCURENCE*)     COMP     2319
000004     LABEL 1,2;                                                          COMP     2320
000004     VAR LCSP: CSP; P1,P2: CTAILP; LFSTOCC: LOCOFREF;                    COMP     2321
000010    BEGIN LCSP := FSTCSP;                                                COMP     2322
000010     WHILE LCSP <> NIL DO                                                COMP     2323
000012      BEGIN P1 := LCSP↑.CSTP; P2 := FCSTP;                               COMP     2324
000015       WHILE (P1 <> NIL)AND (P2 <> NIL) DO                               COMP     2325
000021        BEGIN IF P1↑.CSVAL <> P2↑.CSVAL THEN GOTO 1;                     COMP     2326
000023         P1 := P1↑.NXTCSP; P2 := P2↑.NXTCSP                              COMP     2327
000025        END;                                                             COMP     2328
000026       IF P1 = P2 THEN GOTO 2;                                           COMP     2329
000030    1: LCSP := LCSP↑.NXTCSP                                              COMP     2330
000031      END;                                                               COMP     2331
000033     (*NEW ENTRY:*)                                                      COMP     2332
000033     NEW(LCSP);                                                          COMP     2333
000035     WITH LCSP↑ DO                                                       COMP     2334
000037      BEGIN NXTCSP := FSTCSP; CSTP := FCSTP; CREF := NIL END;            COMP     2335
000046     FSTCSP := LCSP;                                                     COMP     2336
000046    2:(*CHAIN OCCURENCES:*)                                              COMP     2337
000046     LFSTOCC := LCSP↑.CREF; LINKOCC(LFSTOCC);                            COMP     2338
000052     LCSP↑.CREF := LFSTOCC                                               COMP     2339
000053    END (*ENTERCST*) ;                                                   COMP     2340
000061                                                                         COMP     2341
000003    PROCEDURE OPENFILES(FCP: CTP) ;                                      COMP     2342
000004     LABEL 1;                                                            COMP     2343
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A.

```
000004      VAR EXTFILE: BOOLEAN; LNAME: ALFA;                                  COMP     2344
000006        LCST: INTEGER; LCSP: CTAILP;                                      COMP     2345
000010                                                                          COMP     2346
000003      PROCEDURE OPENFL(FSP: STP; FADDR: ADDRRANGE);                       COMP     2347
000005       VAR I,LMIN,LMAX: INTEGER; LCP: CTP;                                COMP     2348
000011      BEGIN                                                               COMP     2349
000011       IF FSP <> NIL THEN                                                 COMP     2350
000011        WITH FSP↑ DO                                                      COMP     2351
000012         IF FTYPE THEN                                                    COMP     2352
000013         CASE FORM OF                                                     COMP     2353
000015          RECORDS:                                                        COMP     2354
000015           BEGIN LCP := FSTFLD;                                           COMP     2355
000016            WHILE LCP <> NIL DO                                           COMP     2356
000020             WITH LCP↑ DO                                                 COMP     2357
000021              BEGIN                                                       COMP     2358
000021               IF FILECNT < 36 THEN OPENFL(IDTYPE,FADDR+FLDADDR)          COMP     2359
000026               ELSE ERROR(258);                                           COMP     2360
000032               LCP:= NEXT                                                 COMP     2361
000032              END                                                         COMP     2362
000035           END;                                                           COMP     2363
000036          ARRAYS:                                                         COMP     2364
000036           IF INXTYPE <> NIL THEN                                         COMP     2365
000040            BEGIN GETBOUNDS(INXTYPE,LMIN,LMAX);                           COMP     2366
000042             IF FILECNT + LMAX - LMIN < 36 THEN                           COMP     2367
000046              FOR I := LMIN TO LMAX DO                                    COMP     2368
000047               BEGIN OPENFL(AELTYPE,FADDR);                               COMP     2369
000055                IF AELTYPE <> NIL THEN                                    COMP     2370
000060                 FADDR := FADDR + AELTYPE↑.SIZE.WORDS                     COMP     2371
000061               END                                                        COMP     2372
000053             ELSE ERROR(258)                                              COMP     2373
000067            END;                                                          COMP     2374
000071          FILES:                                                          COMP     2375
000071           BEGIN LOCFILES := TRUE;                                        COMP     2376
000073            IF TEXTFILE THEN I := FADDR + CHEFET                          COMP     2377
000074            ELSE I := FADDR + BINEFET;                                    COMP     2378
000077            IF (FCP <> INPUTPTR)AND (FCP <> OUTPUTPTR) THEN               COMP     2379
000104             BEGIN NEW(CLSP); (* CLOSE LIST ENTRY *)                      COMP     2380
000106              WITH CLSP↑ DO                                               COMP     2381
000111               BEGIN NXTP := FCLSP; EFETADDR := I END;                    COMP     2382
000116              FCLSP := CLSP;                                              COMP     2383
000116             END;                                                         COMP     2384
000116            GEN30(51B,0,BRGILEVEL],I,0); (*EFET ADDRESS*)                 COMP     2385
000124             IF TEXTFILE THEN I := 208 ELSE I := 0 ;                      UPDATE9    11
000131             IF SEGFILE THEN I := I + 1008;                               COMP     2397
000135             IF EXTFILE THEN                                              COMP     2398
000137              BEGIN                                                       COMP     2399
000137               IF EXFILP↑.READONLY THEN I := I + 1 ELSE I := I+3;         COMP     2400
000145               SHORTNAME(EXFILP↑.FILENAME,LNAME);                         COMP     2401
000151               ORDALFA(LNAME,LCST);                                       COMP     2402
000154              END                                                         COMP     2403
000154             ELSE                                                         COMP     2404
000155              IF FILECNT = 36 THEN ERROR(258)                             COMP     2405
000160              ELSE                                                        COMP     2406
000162               BEGIN FILECNT := FILECNT + 1;                              COMP     2407
000163                PASCF[7] := CHR(FILECNT);                                 COMP     2408
000166                ORDALFA(PASCF,LCST);                                      COMP     2409
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000167                END;                                                    COMP  2410
000167              GEN30(71B,1,0,I,0); (*DISPOSITION CODE*)                  COMP  2412
000173              IF EXTFILE THEN                                           COMP  2413
000175               BEGIN GEN30(51B,2,0,EXFILP↑.SYSLOC,0); (*ACTUAL LFN*)    COMP  2414
000201                GEN30(03B,1,2,IC+1,2)                                   COMP  2415
000203               END;                                                      COMP  2416
000205              GEN30(51B,2,0,0,2);  (*(FORMAL) LFN*)                     COMP  2417
000211              NEW(LCSP);                                                COMP  2418
000213              WITH LCSP↑ DO                                             COMP  2419
000215               BEGIN NXTCSP := NIL; CSVAL := LCST END;                  COMP  2420
000217              ENTERCST(LCSP);                                           COMP  2421
000221              IF EXTFILE THEN NOOP;                                     COMP  2422
000224              IF TEXTFILE THEN I := SIZE.WORDS - CHEFETSZ               COMP  2423
000226              ELSE I := SIZE.WORDS - EFETSZ;                            COMP  2424
000234              GEN30(71B,3,0,I,0); (*BUFFER SIZE*)                       COMP  2425
000240              IF FILTYPE <> NIL THEN                                    COMP  2426
000243               GEN30(71B,6,0,FULLWORDS(FILTYPE↑.SIZE),0); (*LRL*)       COMP  2427
000260              IF TEXTFILE THEN I := FADDR + CHEFETSZ                    COMP  2428
000262              ELSE I := FADDR + EFETSZ;                                 COMP  2429
000267              GEN30(71B,7,BRG[LEVEL],I,0); (*BUFFER ADDRESS*)           COMP  2430
000275              SEARCHEXTID(ΞP.OPEN    Ξ,EXT); RJTOEXT                    COMP  2431
000301            END (*FILES*)                                               COMP  2432
000303          END (*CASE*)                                                  COMP  2433
000307       END (*OPENFL*) ;                                                 COMP  2434
000314                                                                        COMP  2435
000314      BEGIN (*OPENFILES*)                                               COMP  2436
000314       IF FCP <> NIL THEN                                               COMP  2437
000010        WITH FCP↑ DO                                                    COMP  2438
000011         BEGIN OPENFILES(LLINK); OPENFILES(RLINK);                      COMP  2439
000016          IF (KLASS = VARS)AND (VKIND = DRCT) THEN                      COMP  2440
000024           BEGIN EXTFILE := FALSE;                                      COMP  2441
000025            IF (LEVEL = 1)AND (IDTYPE <> NIL) THEN                      COMP  2442
000032             IF IDTYPE↑.FORM = FILES THEN                               COMP  2443
000035              BEGIN EXFILP := FEXFILP;                                  COMP  2444
000037                WHILE EXFILP <> NIL DO                                  COMP  2445
000042                WITH EXFILP↑ DO                                         COMP  2446
000043                 BEGIN                                                  COMP  2447
000043                  IF FILENAME = NAME THEN                               COMP  2448
000044                   BEGIN EXTFILE := TRUE; DECLARED := TRUE;             COMP  2449
000047                    GOTO 1                                              COMP  2450
000050                   END;                                                 COMP  2451
000050                  EXFILP := NXTP                                        COMP  2452
000050                 END;                                                   COMP  2453
000053      1:       END;                                                      COMP  2454
000053           OPENFL(IDTYPE,VADDR)                                         COMP  2455
000056         END                                                            COMP  2456
000057       END (*WITH*)                                                     COMP  2457
000057      END (*OPENFILES*);                                                COMP  2458
000053                                                                        COMP  2459
000003      PROCEDURE ROTATEX(FI: REGNR);                                     COMP  2460
000004       (*IF X-FI IS SHIFTED, SHIFT IT BACK*)                            COMP  2461
000004      BEGIN                                                             COMP  2462
000004       WITH XRGS[FI] DO                                                 COMP  2463
000011        IF XCONT IN [SIMPVAR,INDVAR] THEN                               COMP  2464
000014         IF SHFTCNT <> 0 THEN                                           COMP  2465
000016          BEGIN GEN15(20B,FI,0,WORDSIZE-SHFTCNT); SHFTCNT := 0          COMP  2466
```

MOORE BUSINESS FORMS, INC., FORM 1413 PRINTED IN U.S.A. 22

```
000023        END                                                                COMP     2467
000025    END (*ROTATEX*);                                                       COMP     2468
000031                                                                           COMP     2469
000003    PROCEDURE DECREFX(FI: REGNR);                                          COMP     2470
000004     (*DECREASE NUMBER OF REFERENCES TO X-FI BY ONE*)                       COMP     2471
000004    BEGIN                                                                  COMP     2472
000004     WITH XRGS[FI] DO                                                      COMP     2473
000011      CASE XCONT OF                                                        COMP     2474
000013       AVAIL: ;                                                            COMP     2475
000014       SHRTCST,LONGCST,                                                    COMP     2476
000014       SIMPVAR,INDVAR:                                                     COMP     2477
000014        IF REFNR > 0 THEN                                                  COMP     2478
000016         BEGIN REFNR := REFNR - 1;                                        COMP     2479
000021          IF REFNR = 0 THEN LASTREF := CIX                                COMP     2480
000023         END;                                                             COMP     2481
000027       OTHER:                                                              COMP     2482
000027        IF REFNR > 0 THEN                                                  COMP     2483
000031         BEGIN REFNR := REFNR - 1;                                        COMP     2484
000034          IF REFNR = 0 THEN XCONT := AVAIL                                COMP     2485
000036         END                                                              COMP     2486
000040      END                                                                 COMP     2487
000047    END (*DECREFX*) ;                                                      COMP     2488
000053                                                                           COMP     2489
000003    PROCEDURE BXIXJ(FI,FJ: REGNR);                                         COMP     2490
000005     (*AVOID GENERATION OF B XI XJ INSTRUCTIONS WHENEVER APPROPRIATE BY     COMP     2491
000005      ALTERING PREVIOUSLY GENERATED INSTRUCTION*)                          COMP     2492
000005     VAR I: REGNR;                                                         COMP     2494
000006    BEGIN XRGS[FI] := XRGS[FJ];                                            COMP     2495
000014     IF FI <> FJ THEN                                                      COMP     2496
000015      IF (LASTI = FJ)AND ((LASTOP >= 10B)AND (LASTOP <= 47B)AND            COMP     2497
000020       NOT (LASTOP IN [20B,21B,43B,46B]) OR (LASTOP >= 70B))               UPDATE1    2
000023       AND (XRGS[FJ].REFNR <= 1) THEN                                      UPDATE1    3
000027       BEGIN                                                               COMP     2499
000027        IF (LASTOP < 70B)OR (LASTOP >= 73B) THEN                           COMP     2500
000031         CBUF := CBUF - (LASTI - FI)*100B                                  COMP     2501
000031        ELSE                                                               COMP     2502
000033         CBUF := CBUF - (LASTI - FI)*10000B*1000B;                         COMP     2503
000035        LASTI := FI; XRGS[FJ].XCONT := AVAIL                               COMP     2504
000040       END                                                                 COMP     2505
000043      ELSE                                                                 COMP     2506
000044       BEGIN GEN15( 10B,FI,FJ,0 ); DECREFX(FJ); XRGS[FI].REFNR := 1;       UPDATE22   1
000056        WITH XRGS[ FJ ] DO                                                 UPDATE22   2
000060         IF XCONT = INDVAR THEN                                           UPDATE22   3
000062           WITH XRGS[ XREG ] DO REFNR := REFNR+1 ;                        UPDATE22   4
000070       END ;                                                              UPDATE22   5
000070    END (*BXIXJ*) ;                                                       COMP     2508
000075                                                                           COMP     2509
000003    PROCEDURE CLEARREGS;                                                   COMP     2510
000003     VAR I: REGNR;                                                         COMP     2511
000004    BEGIN                                                                  COMP     2512
000004     FOR I := 0 TO 7 DO                                                    COMP     2513
000007      BEGIN XRGS[I].XCONT := AVAIL; ARGS[I].ACONT := UNSPECADDR;           COMP     2514
000021       IF I IN [3,7] THEN BRGS[I].BCONT := FREE                            COMP     2515
000022       ELSE BRGS[I].BCONT := SPECPURP                                      COMP     2516
000026      END;                                                                 COMP     2517
000034     LEVELS := [0,1,LEVEL]                                                 COMP     2518
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000036    END (*CLEARREGS*) ;                                                    COMP      2519
000043                                                                           COMP      2520
000003    PROCEDURE SAVEREFXRGS(VAR FXRGS: XRGSTATUS);                           COMP      2521
000004     VAR I,J,K: REGNR; LXRGS: XRGSTATUS;                                   COMP      2522
000027    BEGIN LXRGS := XRGS;                                                   COMP      2523
000013     FOR I := 0 TO 7 DO                                                    COMP      2524
000014      WITH LXRGS[I] DO                                                     COMP      2525
000020       IF XCONT = INDVAR THEN                                              COMP      2526
000022        IF REFNR = 0 THEN                                                  COMP      2527
000024         BEGIN                                                            COMP      2528
000024          WITH LXRGS[XREG] DO                                             COMP      2529
000026           IF REFNR > 0 THEN REFNR := REFNR - 1;                          COMP      2530
000033          XCONT := AVAIL                                                  COMP      2531
000034         END;                                                             COMP      2532
000040     K := 0;                                                             COMP      2533
000041     FOR I := 6 TO 7 DO                                                  COMP      2534
000042      WITH LXRGS[I] DO                                                   COMP      2535
000046       IF XCONT <> AVAIL THEN                                            COMP      2536
000050        IF REFNR <> 0 THEN                                               COMP      2537
000051         BEGIN                                                           COMP      2538
000051          IF K = 0 THEN                                                  COMP      2539
000053           BEGIN GEN30(51B,I,BRG[LEVEL],LC,0);                           COMP      2540
000061            ARGS[I].ACONT := UNSPECADDR                                  COMP      2541
000062           END                                                           COMP      2542
000066          ELSE                                                           COMP      2543
000067           BEGIN GEN15(54B,7,6,1);                                       COMP      2544
000073            ARGS[7].ACONT := UNSPECADDR                                  COMP      2545
000073           END;                                                          COMP      2546
000077         IF XCONT = OTHER THEN XRGS[I].XCONT := AVAIL                    COMP      2547
000103         ELSE                                                            COMP      2548
000110          WITH XRGS[I] DO                                                COMP      2549
000113           BEGIN REFNR := 0; LASTREF := CIX END;                         COMP      2550
000120         K := K + 1; J := I                                              COMP      2551
000122        END                                                              COMP      2552
000123       ELSE XCONT := AVAIL;                                              COMP      2553
000127     FOR I := 0 TO 5 DO                                                  COMP      2554
000130      WITH LXRGS[I] DO                                                   COMP      2555
000134       IF XCONT <> AVAIL THEN                                            COMP      2556
000136        IF REFNR <> 0 THEN                                               COMP      2557
000137         BEGIN GEN15(108,7,I,0);                                         COMP      2558
000143          IF XCONT = OTHER THEN XRGS[I].XCONT := AVAIL                   COMP      2559
000147          ELSE                                                           UPDATE8     3
000154           WITH XRGS[ I ] DO                                             UPDATE8     4
000157            BEGIN  REFNR := 0 ;  LASTREF := CIX ;  END ;                 UPDATE8     5
000164         XRGS[7].XCONT := AVAIL;                                         COMP      2561
000170         IF K = 0 THEN GEN30(518,7,BRG[LEVEL],LC,0)                      COMP      2562
000174         ELSE GEN15(548,7,J,1);                                          COMP      2563
000201         ARGS[7].ACONT := UNSPECADDR;                                    COMP      2564
000205         K := K + 1; J := 7                                              COMP      2565
000206        END                                                              COMP      2566
000207       ELSE XCONT := AVAIL;                                              COMP      2567
000213     LC := LC+K ;                                                        UPDATE22    6
000215     IF LC > LCMAX THEN LCMAX := LC ;                                    UPDATE22    7
000217     FXRGS := LXRGS                                                      COMP      2569
000220    END (*SAVEREFXRGS*) ;                                                COMP      2570
000227                                                                         COMP      2571
```

MOORE BUSINESS FORMS, INC., F
FORM 1413
PRINTED IN U.S.A.
22

```
000003    PROCEDURE RELOADREFXRGS(VAR FXRGS: XRGSTATUS);                          COMP     2572
000004     VAR I,J,K,L,M : REGNR ;  LCIX : CODERANGE ;  LCP : POSRANGE ;          UPDATE22    8
000013    BEGIN K := 0 ;  M := 0 ;                                                UPDATE22    9
000011     FOR I := 0 TO 7 DO                                                     COMP     2575
000012      BEGIN J := (I+6) MOD 8;                                               COMP     2576
000016       WITH FXRGS[J] DO                                                     COMP     2577
000017        IF XCONT <> AVAIL THEN                                              COMP     2578
000021         IF REFNR <> 0 THEN                                                 COMP     2579
000023          BEGIN IF I <= 2 THEN L := 5 ELSE L := J;                          COMP     2580
000026           IF K = 0 THEN                                                    UPDATE22   10
000030            BEGIN GEN30(51B,L,BRG[LEVEL],0,0); LCIX := CIX ; LCP := CP      UPDATE22   11
000037            END                                                             UPDATE22   12
000040           ELSE GEN15(54B,L,K,1);                                           COMP     2582
000045           IF I <= 2 THEN GEN15(10B,J,5,0);                                 COMP     2583
000052           XRGS[ J ] := FXRGS[ J ] ;                                        UPDATE9    12
000057           K := L ;  M := M+1 ;                                             UPDATE22   13
000061          END                                                               COMP     2585
000061       END ;                                                                UPDATE22   14
000063      IF M<> 0 THEN                                                         UPDATE22   15
000065       BEGIN LC := LC-M ;  INS( LC,LCIX,LCP ) END                           UPDATE22   16
000071    END (*RELOADXRGS*) ;                                                    COMP     2587
000075                                                                            COMP     2588
000003    PROCEDURE NEEDB(VAR FI: REGNR);                                         COMP     2589
000004     (*RETURN INDEX OF AVAILABLE B-REGISTER*)                               COMP     2590
000004     (*ASSUMES BRGS[I].BCONT <> SPECPURP FOR SOME 1 <= I <= 7*)             COMP     2591
000004     LABEL 1;                                                               COMP     2592
000004     VAR MAXLEV: LEVRANGE; I,NR: REGNR;                                     COMP     2593
000007    BEGIN MAXLEV := 0; NR := 0;                                             COMP     2594
000010     FOR I := 1 TO 7 DO                                                     COMP     2595
000011      WITH BRGS[I] DO                                                       COMP     2596
000014       CASE BCONT OF                                                        COMP     2597
000016        FREE:                                                               COMP     2598
000016         BEGIN NR := I; GOTO 1 END;                                         COMP     2599
000020        BASADDR:                                                            COMP     2600
000020         IF BLEV > MAXLEV THEN                                              COMP     2601
000023          BEGIN MAXLEV := BLEV; NR := I END;                                COMP     2602
000026        SPECPURP:                                                           COMP     2603
000026       END;                                                                 COMP     2604
000034     LEVELS := LEVELS-[MAXLEV];                                             COMP     2605
000037    1:BRGS[NR].BCONT := SPECPURP; FI := NR                                  COMP     2606
000045    END (*NEEDB*) ;                                                         COMP     2607
000051                                                                            COMP     2608
000003    PROCEDURE NEEDX(FLOW,FHIGH: REGNR; VAR FI: REGNR);                      COMP     2609
000006     (*RETURN INDEX FI (FLOW <= FI <= FHIGH) OF AVAILABLE X-REG.;           COMP     2610
000006      DONET TOUCH ANY X-REG. CONTENTS*)                                     COMP     2611
000006     LABEL 1;                                                               COMP     2612
000006     VAR I,NR: REGNR; PR,MAXPR: INTEGER; FIRSTTIME: BOOLEAN;                COMP     2613
000013    BEGIN MAXPR := 0; NR := FLOW; FIRSTTIME := TRUE;                        COMP     2614
000013     REPEAT                                                                 COMP     2615
000013     FOR I := FLOW TO FHIGH DO                                              COMP     2616
000015      WITH XRGS[I] DO                                                       COMP     2617
000021       IF XCONT = AVAIL THEN                                                COMP     2618
000023        BEGIN NR := I; GOTO 1 END                                           COMP     2619
000024       ELSE                                                                 COMP     2620
000025        IF XCONT <> OTHER THEN                                              COMP     2621
000027         IF REFNR = 0 THEN                                                  COMP     2622
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000030          BEGIN PR := CIX - LASTREF + BONUS[XCONT];                                  COMP     2623
000034           IF PR > MAXPR THEN                                                       COMP     2624
000035            BEGIN MAXPR := PR; NR := I END                                          COMP     2625
000036          END;                                                                      COMP     2626
000040       IF MAXPR = 0 THEN                                                            COMP     2627
000042        IF FIRSTTIME THEN                                                           COMP     2628
000043         BEGIN                                                                      COMP     2629
000043          FOR I := 0 TO 7 DO                                                        COMP     2630
000044           WITH XRGS[I] DO                                                          COMP     2631
000050            IF XCONT = INDVAR THEN                                                  COMP     2632
000053             BEGIN DECREFX(XREG);                                                   COMP     2633
000055              IF REFNR = 0 THEN XCONT := AVAIL                                      UPDATE21    2
000060              ELSE                                                                  UPDATE21    3
000062               BEGIN IF SHFTCNT <> 0 THEN GEN15(20B,I,0,WORDSIZE-SHFTCNT);          UPDATE21    4
000071                XCONT := OTHER                                                      UPDATE21    5
000072               END                                                                  UPDATE21    6
000074             END;                                                                   COMP     2635
000076          FIRSTTIME := FALSE                                                        COMP     2636
000076         END                                                                        COMP     2637
000077        ELSE                                                                        COMP     2638
000100         BEGIN IF FLOW <> FHIGH THEN ERROR(259); MAXPR := 1 END                    COMP     2639
000104       UNTIL MAXPR > 0;                                                             COMP     2640
000106     1:WITH XRGS[NR] DO                                                             COMP     2641
000111       BEGIN                                                                        COMP     2642
000111        IF XCONT = INDVAR THEN DECREFX(XREG)                                        COMP     2643
000115        ELSE                                                                        COMP     2644
000117         FOR I := 1 TO 7 DO                                                         COMP     2645
000120          WITH ARGS[I] DO                                                           COMP     2646
000124           IF ACONT = INDADDR THEN                                                  COMP     2647
000126            IF AREG = NR THEN ACONT := UNSPECADDR;                                  COMP     2648
000134        XCONT := OTHER; REFNR := 1                                                  COMP     2649
000137       END;                                                                         COMP     2650
000141     FI := NR                                                                       COMP     2651
000142    END (*NEEDX*) ;                                                                 COMP     2652
000146                                                                                    COMP     2653
000003    PROCEDURE LOADBASE(FLEV: LEVRANGE; FB: BOOLEAN; VAR FI: REGNR);                 COMP     2654
000006     (*LOAD BASE ADDRESS OF DATA SEGMENT WITH LEVEL FLEV INTO X-FI*)                 COMP     2655
000006     LABEL 1;                                                                       COMP     2656
000006     VAR L: REGNR; I,J,K: LEVRANGE; EXITLOOP: BOOLEAN;                              COMP     2657
000013    BEGIN NEEDX(1,5,I);                                                             COMP     2658
000013     FOR J := FLEV + 1 TO LEVEL DO                                                  COMP     2659
000016      IF J IN LEVELS THEN                                                           COMP     2660
000022       BEGIN K := J - 1;                                                            COMP     2661
000023        GEN15(568,I,BRG[J],0);                                                      COMP     2662
000027        (*LOOP UNTIL K = FLEV:*)                                                    COMP     2663
000027        REPEAT                                                                      COMP     2664
000027         IF FB THEN                                                                 COMP     2665
000030          BEGIN NEEDB(L);                                                           COMP     2666
000032           WITH BRGS[L] DO                                                          COMP     2667
000035            BEGIN BCONT := BASADDR; BLEV := K END;                                  COMP     2668
000041           GEN15(63B,L,I,0);                                                        COMP     2669
000044           BRG[K] := L; LEVELS := LEVELS+[K];                                       COMP     2670
000050          END;                                                                      COMP     2671
000050         EXITLOOP := K = FLEV;                                                      COMP     2672
000053         IF NOT EXITLOOP THEN                                                       COMP     2673
000054          BEGIN GEN15(53B,I,I,0); K := K - 1 END                                    COMP     2674
```

MOORE BUSINESS FORMS, INC. F
FORM 1413
PRINTED IN U.S.A.

```
000061          UNTIL EXITLOOP;                                                 COMP     2675
000062          ARGS[I].ACONT := UNSPECADDR;FI := I;                            COMP     2676
000070          GOTO 1                                                          COMP     2677
000070         END;                                                             COMP     2678
000072    1:END (*LOADBASE*) ;                                                  COMP     2679
000076                                                                          COMP     2680
000003     PROCEDURE LOADADDRESS(VAR FATTR: ATTR; VAR FI: REGNR);               COMP     2681
000005      (*LOAD WORD-ADDRESS OF FATTR INTO X-FI*)                            COMP     2682
000005     LABEL 1,2;                                                           COMP     2684
000005     VAR I,J: REGNR;                                                      COMP     2685
000007     BEGIN                                                                COMP     2686
000007      WITH FATTR DO                                                       COMP     2687
000010       BEGIN                                                             COMP     2688
000010       IF TYPTR <> NIL THEN                                              COMP     2689
000012        CASE KIND OF                                                     UPDATE21    7
000013        CST:                                                             UPDATE21    8
000013          (*MUST BE A STRING CONSTANT*)                                  COMP     2691
000013          BEGIN NEEDX(0,7,I); GEN30(71B,I,0,0,2);                        COMP     2692
000022           IF STRING(TYPTR) THEN ENTERCST(CVAL.VALP)                     UPDATE21    9
000032          END;                                                           UPDATE21   10
000034        VARBL:                                                           UPDATE21   11
000034        CASE WORDACC OF                                                  COMP     2696
000036         DRCT:                                                           COMP     2697
000036          BEGIN                                                          COMP     2698
000036           FOR J := 1 TO 7 DO                                            COMP     2699
000037            WITH ARGS[J] DO                                              COMP     2700
000043             IF ACONT = SIMPADDR THEN                                    COMP     2701
000045              IF (ALEV = VLEVEL)AND (AADDR = CWDISPL) THEN               COMP     2702
000052               BEGIN NEEDX(0,7,I); GEN15(74B,I,J,0); GOTO 1 END;         COMP     2703
000063           IF VLEVEL IN LEVELS THEN                                      COMP     2704
000067            BEGIN NEEDX(0,7,I); GEN30(71B,I,BRG[VLEVEL],CWDISPL,1)       COMP     2705
000077            END                                                          COMP     2706
000101           ELSE                                                          COMP     2707
000102            BEGIN LOADBASE(VLEVEL,TRUE,I); GEN30(72B,I,I,CWDISPL,0)      COMP     2708
000110            END;                                                         COMP     2709
000111    1:    END;                                                          COMP     2710
000112         INDRCT:                                                         COMP     2711
000112          BEGIN                                                          COMP     2712
000112           IF CWDISPL = 0 THEN I := VWDISPL                              COMP     2713
000114           ELSE                                                          COMP     2714
000115            BEGIN DECREFX(VWDISPL); NEEDX(0,7,I);                        COMP     2715
000123             FOR J := 1 TO 7 DO                                          COMP     2716
000124              WITH ARGS[J] DO                                            COMP     2717
000130               IF ACONT = INDADDR THEN                                   COMP     2718
000132                IF (AREG = VWDISPL)AND (ADISPL = CWDISPL) THEN           COMP     2719
000137                 BEGIN GEN15(74B,I,J,0); GOTO 2 END;                     COMP     2720
000145           IF CWDISPL = 1 THEN GEN15(73B,I,VWDISPL,1)                    COMP     2721
000152           ELSE GEN30(72B,I,VWDISPL,CWDISPL,0);                          COMP     2722
000161          END;                                                           COMP     2723
000161    2:    END;                                                          COMP     2724
000162         INXD:                                                           COMP     2725
000162          BEGIN DECREFX(VWDISPL);                                        COMP     2726
000165           IF VLEVEL IN LEVELS THEN                                      COMP     2727
000171            BEGIN  NEEDX( 0,7,I ) ;  GEN30( 72B,I,VWDISPL,CWDISPL,0 ) ;  UPDATE5     2
000201             GEN15( 73B,I,I,BRG[ VLEVEL ] ) ;                            UPDATE5     3
000206            END                                                          COMP     2734
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000206          ELSE                                                                COMP     2735
000207           BEGIN NEEDB(J); GEN30(62B,J,VWDISPL,CWDISPL,0);                    COMP     2736
000216            LOADBASE(VLEVEL,TRUE,I); GEN15(73B,I,I,J);                        COMP     2737
000225            BRGS[J].BCONT := FREE                                             COMP     2738
000226           END                                                                COMP     2739
000232         END                                                                  COMP     2740
000232       END (*CASE*);                                                          UPDATE21   12
000237       COND,EXPR:                                                             UPDATE21   13
000237       NEEDX(0,7,I);                                                          UPDATE21   14
000243       END (*CASE*)                                                           UPDATE21   15
000247      ELSE                                                                    COMP     2742
000250       NEEDX(0,7,I);                                                          COMP     2743
000253      WORDACC := INDRCT; VWDISPL := I; CWDISPL := 0                           COMP     2744
000256     END (*WITH FATTR*) ;                                                     COMP     2745
000257     FI := I                                                                  COMP     2746
000257    END (*LOADADDRESS*) ;                                                     COMP     2747
000263                                                                              COMP     2748
000003    PROCEDURE LOAD(VAR FATTR: ATTR; VAR FI: REGNR);                           COMP     2749
000005     (*LOAD FATTR INTO X-FI*)                                                 COMP     2750
000005     LABEL 1,2,4,5,6;                                                         COMP     2751
000005     VAR I,J,K: REGNR; SHRT,SIMPIND: BOOLEAN;                                 COMP     2752
000012       BITSZ,SHIFT,MASK: BITRANGE; LADDR: INTEGER;                            COMP     2753
000016       SVAL: SHRTINT; CSHFT: INTEGER; LCSP: CTAILP;                           COMP     2754
000021       LCST: INTEGER; LMODE: (USRADJ,SRADJ,USLADJ);                           COMP     2755
000023    BEGIN                                                                     COMP     2756
000023     WITH FATTR DO                                                            COMP     2757
000010      BEGIN                                                                   COMP     2758
000010      IF TYPTR <> NIL THEN                                                    COMP     2759
000012       CASE KIND OF                                                           COMP     2760
000013      CST:                                                                    COMP     2761
000013       BEGIN SHRT := FALSE; SVAL := 0; LCSP := NIL;                           COMP     2762
000016        IF STRING(TYPTR) THEN LCSP := CVAL.VALP                               COMP     2763
000023        ELSE                                                                  COMP     2764
000025         BEGIN CONVERTCST(CVAL,TYPTR,LCST);                                   COMP     2765
000031          IF ABS(LCST) < TWOTO17 THEN                                         COMP     2766
000034           BEGIN SVAL := LCST; SHRT := TRUE END                               COMP     2767
000036          ELSE                                                                COMP     2768
000036           BEGIN NEW(LCSP);                                                   COMP     2769
000040            WITH LCSP↑ DO                                                     COMP     2770
000042             BEGIN NXTCSP := NIL; CSVAL := LCST END                           COMP     2771
000043           END                                                                COMP     2772
000043         END;                                                                 COMP     2773
000043        IF SHRT THEN                                                          COMP     2774
000044         BEGIN                                                                COMP     2775
000044          FOR I := 0 TO 7 DO                                                  COMP     2776
000045           WITH XRGS[I] DO                                                    COMP     2777
000051            IF XCONT = SHRTCST THEN                                           COMP     2778
000053             IF CSTVAL = SVAL THEN                                            COMP     2779
000055              BEGIN REFNR := REFNR + 1; GOTO 1 END                            COMP     2780
000060         END                                                                  COMP     2781
000062        ELSE                                                                  COMP     2782
000063         FOR I := 0 TO 7 DO                                                   COMP     2783
000064          WITH XRGS[I] DO                                                     COMP     2784
000070           IF XCONT = LONGCST THEN                                           COMP     2785
000072            IF CPTR↑.CSVAL = LCSP↑.CSVAL THEN                                COMP     2786
000075             BEGIN REFNR := REFNR + 1; GOTO 1 END;                           COMP     2787
```

MOORE BUSINESS FORMS, INC. F
FORM 141
PRINTED IN U.S.A.

```
000102          IF SHRT THEN                                                    COMP    2788
000103           BEGIN NEEDX(0,7,I);                                            COMP    2789
000106            WITH XRGS[I] DO                                               COMP    2790
000111             BEGIN XCONT := SHRTCST; REFNR := 1; CSTVAL := SVAL END;       COMP    2791
000120            IF SVAL = 0 THEN GEN15(13B,I,I,I)                             COMP    2792
000122            ELSE                                                          COMP    2793
000125             IF SVAL = 1 THEN GEN15(76B,I,1,0)                            COMP    2794
000127             ELSE                                                         COMP    2795
000131              IF SVAL = 2 THEN GEN15(76B,I,1,1)                           COMP    2796
000133              ELSE GEN30(71B,I,0,SVAL,0)                                  COMP    2797
000140           END                                                           COMP    2798
000141          ELSE                                                           COMP    2799
000142           BEGIN NEEDX(1,5,I);                                           COMP    2800
000145            WITH XRGS[I] DO                                              COMP    2801
000150             BEGIN XCONT := LONGCST; REFNR := 1; CPTR := LCSP END;        COMP    2802
000157            ARGS[I].ACONT := UNSPECADDR;                                  COMP    2803
000162            GEN30(51B,I,0,0,2);                                           COMP    2804
000164            ENTERCST(LCSP)                                                COMP    2805
000165           END;                                                           COMP    2806
000167     1:   END;                                                            COMP    2807
000170          VARBL:                                                         COMP    2808
000170           BEGIN                                                         COMP    2809
000170            CASE WORDACC OF                                              COMP    2810
000172            DRCT:                                                        COMP    2811
000172             BEGIN                                                       COMP    2812
000172              FOR I := 0 TO 7 DO                                         COMP    2813
000173               WITH XRGS[I] DO                                           COMP    2814
000177                IF XCONT = SIMPVAR THEN                                  COMP    2815
000202                 IF (XLEV = VLEVEL)AND (XADDR = CWDISPL) THEN            COMP    2816
000210                  BEGIN REFNR := REFNR + 1; GOTO 4 END;                  COMP    2817
000215              FOR J := 1 TO 7 DO                                         COMP    2818
000216               WITH ARGS[J] DO                                           COMP    2819
000222                IF ACONT = SIMPADDR THEN                                 COMP    2820
000224                 IF VLEVEL = ALEV THEN                                   COMP    2821
000226                  BEGIN LADDR := CWDISPL - AADDR;                        COMP    2822
000230                   IF ABS(LADDR) <= 1 THEN                               COMP    2823
000232                    BEGIN NEEDX(1,5,I);                                  COMP    2824
000235                     IF LADDR >= 0 THEN GEN15(54B,I,J,LADDR)             COMP    2825
000241                     ELSE GEN15(55B,I,J,1);                              COMP    2826
000250                     GOTO 2                                              COMP    2827
000251                    END                                                  COMP    2828
000251                  END;                                                   COMP    2829
000253              IF VLEVEL IN LEVELS THEN                                   COMP    2830
000257               BEGIN NEEDX(1,5,I); GEN30(51B,I,BRG[VLEVEL],CWDISPL,1)    COMP    2831
000267               END                                                        COMP    2832
000271              ELSE                                                       COMP    2833
000272               BEGIN LOADBASE(VLEVEL,TRUE,I);                             COMP    2834
000275                GEN30(52B,I,I,CWDISPL,0)                                 COMP    2835
000300               END;                                                       COMP    2836
000302     2:       WITH ARGS[I] DO                                            COMP    2837
000305               BEGIN ACONT := SIMPADDR; ALEV := VLEVEL;                   COMP    2838
000312                AADDR := CWDISPL                                          COMP    2839
000312               END;                                                       COMP    2840
000315              WITH XRGS[I] DO                                            COMP    2841
000316               BEGIN XCONT := SIMPVAR; REFNR := 1; VPADDR := FALSE;       COMP    2842
000324                SHFTCNT := 0; XLEV := VLEVEL; XADDR := CWDISPL            COMP    2843
```

MOORE BUSINESS FORMS, INC. F
FORM 113
PRINTED IN U.S.A
22

```
000330                END;                                                    COMP     2844
000333     4:       END;                                                     COMP     2845
000334             INDRCT:                                                   COMP     2846
000334              BEGIN SIMPIND := XRGS[VWDISPL].XCONT = SIMPVAR;          COMP     2847
000342              IF SIMPIND THEN                                          COMP     2848
000342               BEGIN                                                   COMP     2849
000342                FOR I := 0 TO 7 DO                                     COMP     2850
000343                 WITH XRGS[I] DO                                       COMP     2851
000347                  IF XCONT = INDVAR THEN                               COMP     2852
000352                   IF (XREG = VWDISPL)AND (XDISPL = CWDISPL) THEN      COMP     2853
000360                    BEGIN REFNR := REFNR + 1;                          COMP     2854
000363                     DECREFX(VWDISPL); GOTO 6                          COMP     2855
000366                    END;                                               COMP     2856
000370                FOR J := 1 TO 7 DO                                     COMP     2857
000371                 WITH ARGS[J] DO                                       COMP     2858
000375                  IF ACONT = INDADDR THEN                              COMP     2859
000377                   IF (AREG = VWDISPL)AND (ADISPL = CWDISPL) THEN      COMP     2860
000404                    BEGIN NEEDX(1,5,I); GEN15(54B,I,J,0);GOTO 5        COMP     2861
000413                    END                                                COMP     2862
000413               END                                                     COMP     2863
000415              ELSE                                                     COMP     2864
000416               DECREFX(VWDISPL);                                       COMP     2865
000420              NEEDX(1,5,I);                                            COMP     2866
000423              IF CWDISPL IN [0,1] THEN GEN15(53B,I,VWDISPL,CWDISPL)    COMP     2867
000432              ELSE GEN30(52B,I,VWDISPL,CWDISPL,0);                     COMP     2868
000441     5:       IF SIMPIND THEN                                          COMP     2869
000442               BEGIN                                                   COMP     2870
000442                WITH ARGS[I] DO                                        COMP     2871
000444                 BEGIN ACONT := INDADDR;                               COMP     2872
000447                  AREG := VWDISPL; ADISPL := CWDISPL                   COMP     2873
000451                 END;                                                  COMP     2874
000454                WITH XRGS[I] DO                                        COMP     2875
000455                 BEGIN XCONT := INDVAR; REFNR := 1;                    COMP     2876
000462                  SHFTCNT := 0;                                        COMP     2877
000464                  XREG := VWDISPL; XDISPL := CWDISPL                   COMP     2878
000466                 END                                                   COMP     2879
000470               END                                                     COMP     2880
000470              ELSE ARGS[I].ACONT := UNSPECADDR;                        COMP     2881
000475     6:       END;                                                     COMP     2882
000476             INXD:                                                    COMP     2883
000476              BEGIN DECREFX(VWDISPL);                                 COMP     2884
000501              IF VLEVEL IN LEVELS THEN                                COMP     2885
000505                BEGIN  NEEDX( 1,5,I ) ;                               UPDATE5     4
000510                 IF CP = 3 THEN                                       UPDATE5     5
000513                  BEGIN  GEN15( 73B,I,VWDISPL,BRG[ VLEVEL ] ) ;       UPDATE5     6
000520                   GEN30( 52B,I,I,CWDISPL,0 ) ;                       UPDATE5     7
000525                  END                                                 UPDATE5     8
000525                 ELSE                                                 UPDATE5     9
000526                  BEGIN  GEN30( 72B,I,VWDISPL,CWDISPL,0 ) ;           UPDATE5    10
000534                   GEN15( 53B,I,I,BRG[ VLEVEL ] ) ;                   UPDATE5    11
000542                  END ;                                               UPDATE5    12
000542                END                                                   COMP     2900
000542               ELSE                                                   COMP     2901
000543                BEGIN NEEDB(J); GEN30(62B,J,VWDISPL,CWDISPL,0);       COMP     2902
000552                 LOADBASE(VLEVEL,TRUE,I); GEN15(53B,I,I,J);           COMP     2903
000561                 BRGS[J].BCONT := FREE                                COMP     2904
```

MOORE BUSINESS FORMS, INC. FORM 113 PRINTED IN U.S.A. 22

```
000562            END;                                                    COMP    2905
000566           ARGS[I].ACONT := UNSPECADDR                              COMP    2906
000567          END                                                       COMP    2907
000573         END (*CASE*) ;                                             COMP    2908
000577         IF PCKD THEN                                               COMP    2909
000601          BEGIN                                                     COMP    2910
000601           WITH TYPTR↑ DO                                           COMP    2911
000602            BEGIN                                                   COMP    2912
000602             IF FORM = SUBRANGE THEN                                COMP    2913
000604              IF MIN.IVAL < 0 THEN LMODE := SRADJ                   COMP    2914
000605              ELSE LMODE := USRADJ                                  COMP    2915
000606             ELSE                                                   COMP    2916
000610              IF FORM IN [ARRAYS,RECORDS] THEN LMODE := USLADJ      COMP    2917
000613              ELSE LMODE := USRADJ;                                 COMP    2918
000615             BITSZ := SIZE.BITS                                     COMP    2919
000616            END;                                                    COMP    2920
000617           WITH XRGS[I] DO                                          COMP    2921
000622            IF XCONT IN [SIMPVAR,INDVAR] THEN SHIFT := SHFTCNT      COMP    2922
000625            ELSE SHIFT := 0;                                        COMP    2923
000630           IF LMODE = USLADJ THEN MASK := BITSZ                     COMP    2924
000632           ELSE MASK := WORDSIZE - BITSZ;                           COMP    2925
000636           CSHFT := CBDISPL - SHIFT;                                COMP    2926
000641           IF LMODE = USRADJ THEN CSHFT := CSHFT + BITSZ;           COMP    2927
000643           IF BITREG = XREG THEN                                    COMP    2928
000645            BEGIN                                                   COMP    2929
000645             IF SHIFT <> 0 THEN (*TO GUARANTEE 0 <= B-K <= 60*)     COMP    2930
000647              BEGIN GEN15(20B,I,0,WORDSIZE-SHIFT);                  COMP    2931
000653               XRGS[I].SHFTCNT := 0; CSHFT := CSHFT + SHIFT         COMP    2932
000660            END;                                                    COMP    2933
000662           NEEDB(K);                                                COMP    2934
000664           IF CSHFT IN [0,1] THEN GEN15(63B,K,VBDISPL,CSHFT)        COMP    2935
000672           ELSE GEN30(62B,K,VBDISPL,CSHFT,0);                       COMP    2936
000702           DECREFX(VBDISPL); DECREFX(I);                            COMP    2937
000710           NEEDX(0,7,J); GEN15(22B,J,K,I);                          COMP    2938
000716           BRGS[K].BCONT := FREE;                                   COMP    2939
000723           IF LMODE = SRADJ THEN GEN15(21B,J,0,MASK)                COMP    2940
000727           ELSE                                                     COMP    2941
000731            BEGIN NEEDX(0,7,K); GEN15(43B,K,0,MASK);                COMP    2942
000737             IF LMODE = USRADJ THEN GEN15(15B,J,J,K)                COMP    2943
000743             ELSE GEN15(11B,J,J,K);                                 COMP    2944
000751             DECREFX(K)                                             COMP    2945
000752            END;                                                    COMP    2946
000754           I := J                                                   COMP    2947
000754          END                                                       COMP    2948
000755         ELSE                                                       COMP    2949
000756          BEGIN IF CSHFT < 0 THEN CSHFT := CSHFT + WORDSIZE         COMP    2950
000760           ELSE                                                     COMP    2951
000761            IF CSHFT = WORDSIZE THEN CSHFT := 0;                    COMP    2952
000763           WITH XRGS[I] DO                                          COMP    2953
000766            IF XCONT IN [SIMPVAR,INDVAR] THEN                       COMP    2954
000771             IF LMODE = SRADJ THEN                                  COMP    2955
000773              BEGIN NEEDX(0,7,J); DECREFX(I);                       COMP    2956
001001               GEN15(10B,J,I,0); I := J                             COMP    2957
001004               END                                                  COMP    2958
001005             ELSE                                                   COMP    2959
001006              SHFTCNT := (SHFTCNT + CSHFT) MOD WORDSIZE;            COMP    2960
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
001015            IF CSHFT <> 0 THEN GEN15(20B,I,0,CSHFT);                    COMP     2961
001022            IF LMODE = SRADJ THEN GEN15(21B,I,0,MASK)                   COMP     2962
001026            ELSE                                                        COMP     2963
001030             BEGIN NEEDX(0,7,J); GEN15(43B,J,0,MASK);                   COMP     2964
001037              IF LMODE = USRADJ THEN GEN15(15B,J,I,J)                   COMP     2965
001043              ELSE GEN15(11B,J,I,J);                                    COMP     2966
001051              DECREFX(I); I := J                                        COMP     2967
001054             END                                                        COMP     2968
001055           END                                                          COMP     2969
001055         END (*PCKD*)                                                   COMP     2970
001055       END;                                                             COMP     2971
001056      COND:                                                             COMP     2972
001056       BEGIN NEEDX(0,7,I);                                              COMP     2973
001061        IF CONDCD IN [ZR,NZ] THEN                                       COMP     2974
001064         BEGIN GEN15(13B,I,I,I); GEN15(37B,I,I,CDR); NEEDX(0,7,K);      COMP     2975
001100          IF CONDCD = ZR THEN GEN15(13B,K,CDR,I)                        COMP     2976
001105          ELSE GEN15(17B,K,I,CDR);                                      COMP     2977
001115          GEN15(43B,I,0,59); GEN15(15B,I,K,I); DECREFX(K)               COMP     2978
001126         END                                                            COMP     2979
001130        ELSE                                                            COMP     2980
001131         BEGIN GEN15(43B,I,0,1);                                        COMP     2981
001135          IF CONDCD = PL THEN GEN15(11B,I,CDR,I)                        COMP     2982
001142          ELSE GEN15(15B,I,I,CDR);                                      COMP     2983
001151          GEN15(20B,I,0,1)                                              COMP     2984
001153         END;                                                           COMP     2985
001154        DECREFX(CDR)                                                    COMP     2986
001156       END;                                                             COMP     2987
001160      EXPR:                                                             COMP     2988
001160        I := EXPREG                                                     COMP     2989
001160       END (*CASE*)                                                     COMP     2990
001166      ELSE NEEDX(0,7,I);                                                COMP     2991
001172      KIND := EXPR; EXPREG := I                                         COMP     2992
001174      END (*WITH FATTR*) ;                                              COMP     2993
001175     FI := I                                                            COMP     2994
001175    END (*LOAD*) ;                                                      COMP     2995
001202                                                                        COMP     2996
000003    PROCEDURE STORE(VAR FATTR: ATTR; FI: REGNR);                        COMP     2997
000005     (*STORE X-FI AT FATTR*)                                            COMP     2998
000005     (*ASSUMES FATTR.KIND = VARBL*)                                     COMP     2999
000005     LABEL 1,2;                                                         COMP     3000
000005     VAR I,J,K,LNR: REGNR; LATTR: ATTR; LXRG: XRGSTAT;                  COMP     3001
000025       BITSZ,SHIFT,MASK: BITRANGE; CSHFT: INTEGER; LCST: SHRTINT;       COMP     3002
000032       LCP: POSRANGE; LADDR: INTEGER; LFTADJ,LBX,LXFICST: BOOLEAN;      COMP     3003
000037                                                                        COMP     3004
000037    BEGIN                                                               COMP     3005
000037     WITH FATTR DO                                                      COMP     3006
000010      IF TYPTR <> NIL THEN                                              COMP     3007
000012       BEGIN                                                            COMP     3008
000012        IF PCKD THEN                                                    COMP     3009
000013         BEGIN LATTR := FATTR;                                          COMP     3010
000016          IF WORDACC <> DRCT THEN                                       UPDATE9    13
000020           WITH XRGS[ VWDISPL ] DO REFNR := REFNR+1 ;                   UPDATE9    14
000025          LATTR.PCKD := FALSE; LOAD(LATTR,I);                           COMP     3017
000031          WITH TYPTR^ DO                                                COMP     3018
000033           BEGIN LFTADJ := FORM IN [ARRAYS,RECORDS];                    COMP     3019
000036            BITSZ := SIZE.BITS                                          COMP     3020
```

MOORE BUSINESS FORMS, INC. F413 PRINTED IN U.S.A 22

```
000036          END;                                                                COMP    3021
000037         WITH XRGS[I] DO                                                      COMP    3022
000042          IF XCONT IN [SIMPVAR,INDVAR] THEN SHIFT := SHFTCNT                  COMP    3023
000045          ELSE SHIFT := 0;                                                    COMP    3024
000050         IF LFTADJ THEN                                                       COMP    3025
000051          BEGIN MASK := BITSZ; CSHFT := CBDISPL - SHIFT END                   COMP    3026
000055         ELSE                                                                 COMP    3027
000055          BEGIN MASK := WORDSIZE - BITSZ;                                     COMP    3028
000057           CSHFT := CBDISPL - SHIFT + BITSZ                                   COMP    3029
000060          END;                                                                COMP    3030
000061         IF BITREG = XREG THEN                                                COMP    3031
000063          BEGIN                                                               COMP    3032
000063           IF BITSZ < SHIFT THEN (*TO GUARANTEE 0 <= B-K <= 60*)              COMP    3033
000065            BEGIN GEN15(20B,I,0,WORDSIZE - SHIFT);                           COMP    3034
000071             XRGS[I].SHFTCNT := 0; CSHFT := CSHFT + SHIFT                    COMP    3035
000076            END;                                                             COMP    3036
000100           NEEDB(K);                                                         COMP    3037
000102           IF CSHFT IN [0,1] THEN GEN15(63B,K,VBDISPL,CSHFT)                 COMP    3038
000110           ELSE GEN30(62B,K,VBDISPL,CSHFT,0);                                COMP    3039
000120           DECREFX(VBDISPL); DECREFX(I); NEEDX(0,7,J);                       COMP    3040
000131           GEN15(22B,J,K,I);                                                 COMP    3041
000134          END                                                                COMP    3042
000134         ELSE                                                                 COMP    3043
000135          BEGIN                                                               COMP    3044
000135           IF CSHFT < 0 THEN CSHFT := CSHFT + WORDSIZE                        COMP    3045
000137           ELSE IF CSHFT = WORDSIZE THEN CSHFT := 0;                          COMP    3046
000142           IF CSHFT <> 0 THEN GEN15(20B,I,0,CSHFT); J := I;                   COMP    3047
000150            WITH XRGS[I] DO                                                   COMP    3048
000152             IF XCONT IN [SIMPVAR,INDVAR] THEN                                COMP    3049
000155              BEGIN SHFTCNT := (SHFTCNT + CSHFT) MOD WORDSIZE;                COMP    3050
000163               CSHFT := 0                                                     COMP    3051
000163              END                                                             COMP    3052
000164          END;                                                                COMP    3053
000164         NEEDX(0,7,I); GEN15(43B,I,0,MASK);                                   COMP    3054
000173         IF LFTADJ THEN                                                       COMP    3055
000174          BEGIN GEN15(15B,J,J,I); GEN15(11B,I,FI,I) END                       COMP    3056
000204         ELSE                                                                 COMP    3057
000205          BEGIN GEN15(11B,J,J,I); GEN15(15B,I,FI,I) END;                      COMP    3058
000215         DECREFX(FI); GEN15(12B,J,J,I);                                       COMP    3059
000223         IF BITREG = XREG THEN                                                COMP    3060
000225          BEGIN GEN30(61B,K,K,-WORDSIZE,0); GEN15(23B,J,K,J);                 COMP    3061
000235           BRGS[K].BCONT := FREE                                              COMP    3062
000236          END                                                                 COMP    3063
000242         ELSE IF CSHFT <> 0 THEN GEN15(20B,J,0,WORDSIZE-CSHFT);               COMP    3064
000251         DECREFX(I); FI := J                                                  COMP    3065
000254        END (*PCKD*) ;                                                        COMP    3066
000256       LCP := CP; LNR := FI;                                                  COMP    3067
000261       IF NOT (FI IN [6,7]) THEN                                              COMP    3068
000263        BEGIN NEEDX(6,7,I); BXIXJ(I,FI);                                      COMP    3069
000272         FI := I                                                             COMP    3070
000272        END;                                                                  COMP    3071
000274       LBX := LCP <> CP;                                                      COMP    3072
000300       WITH XRGS[FI] DO                                                       COMP    3073
000302        BEGIN LXFICST := XCONT = SHRTCST;                                     COMP    3074
000305         IF LXFICST THEN LCST := CSTVAL                                      COMP    3075
000305         ELSE ROTATEX(FI);                                                   COMP    3076
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000311      END;                                                              COMP     3077
000311     CASE WORDACC OF                                                    COMP     3078
000313      DRCT:                                                             COMP     3079
000313       BEGIN                                                            COMP     3080
000313        FOR I := 0 TO 7 DO                                              COMP     3081
000314         IF I <> FI THEN                                                COMP     3082
000317          WITH XRGS[I] DO                                               COMP     3083
000321           IF XCONT = SIMPVAR THEN                                      COMP     3084
000324            IF (XLEV = VLEVEL)AND (XADDR = CWDISPL) THEN                COMP     3085
000332             XCONT := AVAIL;                                            COMP     3086
000336        IF TYPTR↑.FORM = POINTER THEN                                   COMP     3087
000341         BEGIN                                                          COMP     3088
000341          FOR I := 0 TO 7 DO                                            COMP     3089
000342           WITH XRGS[I] DO                                              COMP     3090
000346            IF XCONT = INDVAR THEN                                      COMP     3091
000351             IF XRGS[XREG].XCONT = AVAIL THEN XCONT := AVAIL;           COMP     3092
000360          FOR I := 1 TO 7 DO                                            COMP     3093
000361           WITH ARGS[I] DO                                              COMP     3094
000365            IF ACONT = INDADDR THEN                                     COMP     3095
000367             IF XRGS[AREG].XCONT = AVAIL THEN                           COMP     3096
000372              ACONT := UNSPECADDR                                       COMP     3097
000372         END;                                                           COMP     3098
000376        WITH LXRG DO                                                    COMP     3099
000376         BEGIN XCONT := SIMPVAR; REFNR := 1; VPADDR := FALSE;           COMP     3100
000405          XLEV := VLEVEL; XADDR := CWDISPL; SHFTCNT := 0                COMP     3101
000412         END                                                            COMP     3102
000414        END;                                                            COMP     3103
000414       INDRCT:                                                          COMP     3104
000414        BEGIN                                                           COMP     3105
000414         FOR I := 0 TO 7 DO                                             COMP     3106
000415          IF I <> FI THEN                                               COMP     3107
000420           WITH XRGS[I] DO                                              COMP     3108
000422            IF XCONT = INDVAR THEN                                      COMP     3109
000425             IF (XREG = VWDISPL)AND (XDISPL = CWDISPL) THEN             COMP     3110
000433              BEGIN DECREFX(VWDISPL); XCONT := AVAIL END;               COMP     3111
000442        IF XRGS[VWDISPL].XCONT = SIMPVAR THEN                           COMP     3112
000447         WITH LXRG DO                                                   COMP     3113
000447          BEGIN XCONT := INDVAR ;  REFNR := 1 ;                         UPDATE9    15
000454           XREG := VWDISPL; XDISPL := CWDISPL; SHFTCNT := 0             COMP     3115
000457          END                                                           COMP     3116
000461        ELSE                                                            COMP     3117
000462         WITH LXRG DO                                                   COMP     3118
000462          BEGIN XCONT := OTHER; REFNR := 1 END                          COMP     3119
000467        END;                                                            COMP     3120
000470       INXD:                                                            COMP     3121
000470        WITH LXRG DO                                                    COMP     3122
000470         BEGIN XCONT := OTHER; REFNR := 1 END                           COMP     3123
000475     END (*CASE*);                                                      COMP     3124
000501     IF WORDACC <> DRCT THEN DECREFX( VWDISPL ) ;                       UPDATE9    16
000506     IF LXRG.XCONT = OTHER THEN                                         COMP     3125
000511      BEGIN                                                             UPDATE3     4
000511       IF NOT LXFICST AND LBX THEN                                      UPDATE3     5
000513        WITH XRGS[FI] DO                                                UPDATE3     6
000515         BEGIN IF XCONT = INDVAR THEN DECREFX(XREG);                    UPDATE3     7
000522          XCONT := OTHER                                                UPDATE3     8
000523         END                                                            UPDATE3     9
```

MOORE BUSINESS FORMS, INC. F
FORM 1413
PRINTED IN U.S.A
22

```
000525          END                                                                 UPDATE3    10
000525         ELSE                                                                 COMP      3127
000526          BEGIN  IF LBX THEN K := LNR ELSE K := FI ;                          UPDATE9    17
000533           IF ( LXRG.XCONT <> INDVAR ) OR ( LXRG.XREG <> K ) THEN             UPDATE9    18
000541            BEGIN                                                             UPDATE9    19
000541             WITH XRGS[ K ] DO                                                UPDATE9    20
000543              IF XCONT = INDVAR THEN DECREFX( XREG )                          UPDATE9    21
000546              ELSE IF XCONT = SIMPVAR THEN                                    UPDATE9    22
000552                    FOR I := 0 TO 7 DO                                        UPDATE9    23
000553                     WITH XRGS[ I ] DO                                        UPDATE9    24
000557                      IF XCONT = INDVAR THEN                                  UPDATE9    25
000562                       IF XREG = K THEN XCONT := OTHER ;                      UPDATE9    26
000570              IF K = LNR THEN                                                 UPDAT10    18
000572               BEGIN  LXRG.REFNR := XRGS[ K ].REFNR ;                         UPDAT10    19
000577                IF LXRG.REFNR = 0 THEN LXRG.LASTREF := CIX ;                  UPDAT10    20
000603               END ;                                                          UPDAT10    21
000603              XRGS[ K ] := LXRG ;                                             UPDATE9    27
000610              IF LXRG.XCONT = INDVAR THEN                                     UPDATE9    28
000512               WITH XRGS[ LXRG.XREG ] DO REFNR := REFNR+1 ;                   UPDATE9    29
000620            END                                                               UPDATE9    31
000620           END ;                                                              UPDATE9    32
000620          CASE WORDACC OF                                                     COMP      3131
000622           ORCT:                                                              COMP      3132
000622            BEGIN                                                             COMP      3133
000622             FOR I := 1 TO 7 DO                                               COMP      3134
000623              WITH ARGS[I] DO                                                 COMP      3135
000627               IF ACONT = SIMPADDR THEN                                       COMP      3136
000631                IF ALEV = VLEVEL THEN                                         COMP      3137
000633                 BEGIN LADDR := CWDISPL - AADDR;                              COMP      3138
000635                  IF ABS(LADDR) <= 1 THEN                                     COMP      3139
000637                   BEGIN                                                      COMP      3140
000637                    IF LADDR >= 0 THEN GEN15(54B,FI,I,LADDR)                  COMP      3141
000642                    ELSE GEN15(55B,FI,I,1);                                   COMP      3142
000651                    GOTO 1                                                    COMP      3143
000652                   END;                                                       COMP      3144
000652                 END;                                                         COMP      3145
000654            IF VLEVEL IN LEVELS THEN GEN30(51B,FI,BRG[VLEVEL],CWDISPL,1)      COMP      3146
000664            ELSE                                                              COMP      3147
000667             BEGIN LOADBASE(VLEVEL,TRUE,I); GEN30(52B,FI,I,CWDISPL,0);       COMP      3148
000677              DECREFX(I)                                                      COMP      3149
000700             END;                                                            COMP      3150
000702   1:       WITH ARGS[FI] DO                                                  COMP      3151
000705             BEGIN ACONT := SIMPADDR; ALEV := VLEVEL; AADDR := CWDISPL       COMP      3152
000712             END                                                             COMP      3153
000715           END;                                                              COMP      3154
000715          INDRCT:                                                            COMP      3155
000715           BEGIN                                                             COMP      3156
000715            IF LXRG.XCONT = INDVAR THEN                                      COMP      3157
000720             BEGIN                                                           COMP      3158
000720              FOR I := 1 TO 7 DO                                             COMP      3159
000721               WITH ARGS[I] DO                                               COMP      3160
000725                IF ACONT = INDADDR THEN                                      COMP      3161
000727                 IF (AREG = VWDISPL)AND (ADISPL = CWDISPL) THEN              COMP      3162
000734                  BEGIN GEN15(54B,FI,I,0); GOTO 2 END                        COMP      3163
000741             END ;                                                           UPDATE9    33
000743            IF CWDISPL IN [0,1] THEN GEN15(53B,FI,VWDISPL,CWDISPL)           COMP      3166
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000752          ELSE GEN30(52B,FI,VWDISPL,CWDISPL,0);                          COMP     3167
000761   2:     WITH ARGS[FI] DO                                               COMP     3168
000764          IF LXRG.XCONT = INDVAR THEN                                    COMP     3169
000767           BEGIN ACONT := INDADDR; AREG := VWDISPL;                      COMP     3170
000774            ADISPL := CWDISPL                                            COMP     3171
000774           END                                                           COMP     3172
000777          ELSE ACONT := UNSPECADDR                                       COMP     3173
000777         END;                                                            COMP     3174
001003        INXD:                                                            COMP     3175
001003         BEGIN                                                           UPDATE9    34
001003          IF VLEVEL IN LEVELS THEN                                       COMP     3177
001006            BEGIN NEEDB(I);                                              COMP     3180
001010             IF (CP = 3)OR (I = BRG[VLEVEL]) THEN                        COMP     3181
001017              BEGIN GEN15(63B,I,VWDISPL,BRG[VLEVEL]);                    COMP     3182
001023               GEN30(51B,FI,I,CWDISPL,0)                                 COMP     3183
001026              END                                                        COMP     3184
001030             ELSE                                                        COMP     3185
001031              BEGIN GEN30(62B,I,VWDISPL,CWDISPL,0);                      COMP     3186
001035               GEN15(56B,FI,BRG[VLEVEL],I)                               COMP     3187
001041              END;                                                       COMP     3188
001043             BRGS[I].BCONT := FREE                                       COMP     3189
001044            END                                                          COMP     3190
001050          ELSE                                                           COMP     3191
001051           BEGIN NEEDB(J); GEN30(62B,J,VWDISPL,CWDISPL,0);               COMP     3192
001060            LOADBASE(VLEVEL,TRUE,I); GEN15(53B,FI,I,J);                  COMP     3193
001070            BRGS[J].BCONT := FREE; DECREFX(I)                            COMP     3194
001076           END;                                                          COMP     3195
001077          ARGS[FI].ACONT := UNSPECADDR                                   COMP     3196
001100         END                                                             COMP     3197
001104        END (*CASE*);                                                    COMP     3198
001110        IF VLEVEL > 0 THEN                                               UPDATE22   17
001112        (*UPDATE OF REGISTER CONTENTS:*)                                 COMP     3199
001112        IF WORDACC = DRCT THEN   (*SIMPLE VAR HAS GOT NEW VALUE. DISPOSE*) COMP   3200
001114         BEGIN                  (*X-REGS CONTAINING VAR PARAMS*)         COMP     3201
001114          FOR I := 0 TO 7 DO                                             COMP     3202
001115           IF I <> FI THEN                                               COMP     3203
001120            WITH XRGS[I] DO                                              COMP     3204
001122             IF XCONT = INDVAR THEN                                      COMP     3205
001125              IF XRGS[XREG].VPADDR THEN                                  COMP     3206
001130               IF XRGS[XREG].XLEV > VLEVEL THEN                          COMP     3207
001134                BEGIN DECREFX(XREG);                                     UPDATE22   18
001136                 IF REFNR > 0 THEN XCONT := OTHER ELSE XCONT := AVAIL    UPDATE22   19
001144                END                                                      UPDATE22   20
001146         END                                                             COMP     3209
001150        ELSE  (*ASSUME COINCIDANCE. DISPOSE X-REGS NOT CONT. SIMPLE VARS*) COMP   3210
001151         BEGIN                                                           COMP     3211
001151          FOR I := 0 TO 7 DO                                             COMP     3212
001152           IF I <> FI THEN                                               COMP     3213
001155            WITH XRGS[I] DO                                              COMP     3214
001157             IF XCONT = INDVAR THEN                                      COMP     3215
001162              BEGIN DECREFX(XREG);                                       UPDATE22   21
001164               IF REFNR > 0 THEN XCONT := OTHER ELSE XCONT := AVAIL      UPDATE22   22
001172              END ;                                                      UPDATE22   23
001176         IF XRGS[VWDISPL].VPADDR THEN   (*DISPOSE X-REGS CONTAINING*)    COMP     3217
001202          BEGIN                        (*SIMPLE VARS OF LEVEL < XLEV*)   COMP     3218
001202           FOR I := 0 TO 7 DO                                            COMP     3219
```

MOORE BUSINESS FORMS, INC. F
FORM 1413
PRINTED IN U.S.A
22

```
001203            IF I <> FI THEN                                                    COMP     3220
001206             WITH XRGS[I] DO                                                   COMP     3221
001210              IF XCONT = SIMPVAR THEN                                          COMP     3222
001213               IF XLEV < XRGS[VWDISPL].XLEV THEN                               UPDATE22   24
001217                IF REFNR > 0 THEN XCONT := OTHER ELSE XCONT := AVAIL ;         UPDATE22   25
001230           FOR I := 1 TO 7 DO                                                  COMP     3224
001231            WITH ARGS[I] DO                                                   COMP     3225
001235             IF ACONT = INDADDR THEN                                          COMP     3226
001237              IF XRGS[AREG].XCONT = AVAIL THEN                                COMP     3227
001242               ACONT := UNSPECADDR                                            COMP     3228
001242          END                                                                 COMP     3229
001246        END;                                                                  COMP     3230
001246     END (*TYPTR <> NIL*);                                                    COMP     3236
001246    DECREFX(FI)                                                               COMP     3237
001247  END (*STORE*) ;                                                             COMP     3238
001255                                                                              COMP     3239
000003  PROCEDURE CHECKBNDS(FI: REGNR; FMIN,FMAX: INTEGER; FADDR: ADDRRANGE);      COMP     3240
000007   (*TEST X-FI AGAINST BOUNDS FMIN AND FMAX.IF OUT OF BOUNDS JUMP            COMP     3241
000007    TO FADDR*)                                                               COMP     3242
000007    VAR BOUND: ATTR; I,J,K: REGNR;                                           COMP     3243
000024  BEGIN GEN30(51B,0,0,IC,0);                                                  COMP     3244
000014    WITH BOUND DO                                                            COMP     3245
000014     BEGIN TYPTR := INTPTR; KIND := CST; CVAL.IVAL := FMIN END;               COMP     3246
000020    IF FMIN <> 0 THEN                                                        COMP     3247
000021     BEGIN LOAD(BOUND,I); DECREFX(I); NEEDX(0,7,K);                          COMP     3248
000031      GEN15(37B,K,FI,I)                                                      COMP     3249
000033     END;                                                                     COMP     3250
000035    WITH BOUND DO                                                            COMP     3251
000035     BEGIN TYPTR := INTPTR; KIND := CST; CVAL.IVAL := FMAX END;              COMP     3252
000041    LOAD(BOUND,I);                                                           COMP     3253
000043    DECREFX(I); NEEDX(0,7,J); GEN15(37B,J,I,FI);                             COMP     3254
000055    IF FMIN <> 0 THEN                                                        COMP     3255
000057     BEGIN GEN15(12B,J,J,K); DECREFX(K) END                                  COMP     3256
000065    ELSE GEN15(12B,J,J,FI);                                                  COMP     3257
000072    GEN30(03B,3,J,FADDR,0); DECREFX(J)                                       COMP     3258
000077  END (*CHECKBNDS*) ;                                                        COMP     3259
000105                                                                             COMP     3260
000003    PROCEDURE STATEMENT(FSYS: SETOFSYS);                                     COMP     3261
000004     LABEL 1;                                                                COMP     3262
000004     VAR LCP: CTP; LLP: LBP; LOCP: LOCOFREF;                                 COMP     3263
000007                                                                             COMP     3264
000003     PROCEDURE PACKOFL(FI: REGNR);                                           COMP     3265
000004      VAR K: REGNR;                                                          COMP     3266
000005     BEGIN GEN30(51B,0,0,IC,0); NEEDX(0,7,K); GEN15(10B,K,FI,0);             COMP     3267
000022      GEN15(21B,K,0,48); GEN30(03B,1,K,OVLERR,0); DECREFX(K)                 COMP     3268
000033     END (*PACKOFL*) ;                                                       COMP     3269
000041                                                                             COMP     3270
000003     PROCEDURE PACKANDNORM(VAR FI: REGNR);                                   COMP     3271
000004      VAR K: REGNR;                                                          COMP     3272
000005     BEGIN IF DEBUG THEN PACKOFL(FI); DECREFX(FI); NEEDX(0,7,K);             COMP     3273
000021      GEN15(27B,K,0,FI); GEN15(24B,K,0,K);                                   COMP     3274
000031      FI := K                                                                COMP     3275
000032     END (*PACKANDNORM*) ;                                                   COMP     3276
000036                                                                             COMP     3277
000003     PROCEDURE OPERATION(FOP: OPRANGE; VAR FK: REGNR; FI,FJ: REGNR);         COMP     3278
000007     BEGIN DECREFX(FI); DECREFX(FJ); NEEDX(0,7,FK); GEN15(FOP,FK,FI,FJ)      COMP     3279
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000024      END (*OPERATION*) ;                                                  COMP    3280
000031                                                                           COMP    3281
000003      PROCEDURE EXPREP(FVAL: INTEGER;VAR FREC: CSTREC);                    COMP    3282
000005       (*RETURN EXPONENTIAL REPRESENTATION OF FVAL:                        COMP    3283
000005        CKIND = PUREP    IF    FVAL = 2**EXP,                             COMP    3284
000005        CKIND = POSP     IF    FVAL = 2**EXP1*(2**EXP2 + 1),              COMP    3285
000005        CKIND = NEGP     IF    FVAL = 2**EXP1*(2**EXP2 - 1),              COMP    3286
000005        CKIND = NOP      ELSE.*)                                           COMP    3287
000005       VAR E1,E2: BITRANGE;                                                COMP    3288
000007      BEGIN                                                                COMP    3289
000007       IF FVAL > 0 THEN                                                    COMP    3290
000010        BEGIN E1 := 0;                                                     COMP    3291
000011         WHILE NOT ODD(FVAL) DO                                            COMP    3292
000013          BEGIN FVAL := FVAL DIV 2; E1 := E1 + 1 END;                      COMP    3293
000016         IF FVAL = 1 THEN                                                  COMP    3294
000017          WITH FREC DO                                                     COMP    3295
000020           BEGIN CKIND := PUREP; EXP := E1 END                             COMP    3296
000024         ELSE                                                              COMP    3297
000025          BEGIN FVAL := FVAL DIV 2; E2 := 1;                               COMP    3298
000027           IF ODD(FVAL) THEN                                               COMP    3299
000031            BEGIN                                                          COMP    3300
000031             REPEAT FVAL := FVAL DIV 2; E2 := E2 + 1                       COMP    3301
000033             UNTIL NOT ODD(FVAL);                                          COMP    3302
000036             IF FVAL > 0 THEN FREC.CKIND := NOP                            COMP    3303
000037             ELSE                                                          COMP    3304
000042              WITH FREC DO                                                 COMP    3305
000043               BEGIN CKIND := NEGP; EXP1 := E1; EXP2 := E2 END             COMP    3306
000052             END                                                           COMP    3307
000052            ELSE                                                           COMP    3308
000053             BEGIN                                                         COMP    3309
000053              REPEAT FVAL := FVAL DIV 2; E2 := E2 + 1                      COMP    3310
000055              UNTIL ODD(FVAL);                                             COMP    3311
000060              IF FVAL > 1 THEN FREC.CKIND := NOP                           COMP    3312
000061              ELSE                                                         COMP    3313
000064               WITH FREC DO                                                COMP    3314
000065                BEGIN CKIND := POSP; EXP1 := E1; EXP2 := E2 END            COMP    3315
000073             END                                                           COMP    3316
000073           END                                                             COMP    3317
000073         END                                                               COMP    3318
000073        ELSE FREC.CKIND := NOP                                             COMP    3319
000074      END (*EXPREP*) ;                                                     COMP    3320
000102                                                                           COMP    3321
000003      PROCEDURE OPTMULT(FI: REGNR; FREC: CSTREC; FEQ: BOOLEAN;             COMP    3322
000006      VAR FK: REGNR);                                                      COMP    3323
000007       (*GENERATE CODE FOR X-FK := X-FI*FREC.  FEQ <=> FI=FK IS ALLOWED*)  COMP    3324
000007       VAR E: BITRANGE; I,K: REGNR; B: BOOLEAN;                            COMP    3325
000013      BEGIN B := FALSE;                                                    COMP    3326
000011       WITH FREC DO                                                        COMP    3327
000011        BEGIN IF CKIND = PUREP THEN E := EXP ELSE E := EXP1;               COMP    3328
000017         IF E <> 0 THEN                                                    COMP    3329
000021          IF E = 1 THEN                                                    COMP    3330
000022           BEGIN NEEDX(0,7,K); GEN15(36B,K,FI,FI);                         COMP    3331
000031            IF FEQ OR (CKIND = PUREP) THEN DECREFX(FI)                     COMP    3332
000036            ELSE                                                           COMP    3333
000040             BEGIN B := TRUE; I := FI END;                                 COMP    3334
000042            FI := K                                                        COMP    3335
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000042          END                                                                 COMP     3336
000044         ELSE                                                                 COMP     3337
000044          IF FEQ AND (XRGS[FI].REFNR = 1) THEN (*DESTROY X-FI*)               COMP     3338
000052           BEGIN GEN15(20B,FI,0,E);                                           UPDATE1     5
000055            WITH XRGS[FI] DO                                                  UPDATE1     6
000061             BEGIN IF XCONT = INDVAR THEN DECREFX(XREG);                      UPDATE1     7
000065              XCONT := OTHER                                                  UPDATE1     8
000066             END                                                             UPDATE1     9
000070           END                                                               COMP     3340
000070          ELSE (*COPY X-FI*)                                                 COMP     3341
000071           BEGIN NEEDX(0,7,K); GEN15(10B,K,FI,0);                            COMP     3342
000100            GEN15(20B,K,0,E);                                                COMP     3343
000104            IF FEQ OR (CKIND = PUREP) THEN DECREFX(FI)                       COMP     3344
000111            ELSE                                                             COMP     3345
000113             BEGIN B := TRUE; I := FI END;                                   COMP     3346
000115            FI := K                                                          COMP     3347
000115           END;                                                              COMP     3348
000117         IF CKIND <> PUREP THEN                                              COMP     3349
000122          BEGIN NEEDX(0,7,K);                                                COMP     3350
000125           IF B THEN DECREFX(I);                                             COMP     3351
000131           GEN15(10B,K,FI,0); GEN15(20B,K,0,EXP2);                           COMP     3352
000142           IF CKIND = POSP THEN GEN15(36B,K,K,FI)                            COMP     3353
000147           ELSE GEN15(37B,K,K,FI);                                           COMP     3354
000156           DECREFX(FI); FK := K                                              COMP     3355
000162          END                                                                COMP     3356
000163         ELSE FK := FI                                                       COMP     3357
000164       END                                                                   COMP     3358
000165      END (*OPTMULT*) ;                                                      COMP     3359
000171                                                                             COMP     3360
000003      PROCEDURE EXPRESSION(FSYS: SETOFSYS); FORWARD;                         COMP     3361
000004                                                                             COMP     3362
000003    PROCEDURE ASSIGNTO( VAR LATTR : ATTR ) ;                                 UPDATE7     6
000004      VAR  I,J,K,L,M : REGNR ; FLOAT : BOOLEAN ;                             UPDATE7     7
000012           LWORDS : ADDRRANGE ;  LMIN,LMAX : INTEGER ;                       UPDATE7     8
000015           N : REGNR ;                                                       OPT6600A    1
000016    BEGIN                                                                    UPDATE7     9
000016        IF (LATTR.TYPTR <> NIL)AND (GATTR.TYPTR <> NIL) THEN                 UPDATE7    10
000014         BEGIN                                                               UPDATE7    11
000014          IF COMPTYPES(REALPTR,LATTR.TYPTR,FALSE)                            UPDATE7    12
000016           AND COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN                      UPDATE7    13
000027           FLOAT := TRUE                                                     UPDATE7    14
000027          ELSE FLOAT := FALSE;                                               UPDATE7    15
000031          IF COMPTYPES( LATTR.TYPTR,GATTR.TYPTR,FALSE ) OR FLOAT THEN        UPDATE7    16
000040           CASE LATTR.TYPTR↑.FORM OF                                         UPDATE7    17
000042            SCALAR,                                                          UPDATE7    18
000042            SUBRANGE:                                                        UPDATE7    19
000042             BEGIN                                                           UPDATE7    20
000042              IF (LATTR.TYPTR=INTPTR)OR (COMPTYPES(LATTR.TYPTR,              UPDATE7    21
000044               REALPTR,FALSE)) THEN LOAD(GATTR,I)                            UPDATE7    22
000054              ELSE                                                           UPDATE7    23
000056               BEGIN GETBOUNDS(LATTR.TYPTR,LMIN,LMAX);                       UPDATE7    24
000051                IF GATTR.KIND = CST THEN                                     UPDATE7    25
000064                 BEGIN                                                       UPDATE7    26
000064                  IF (GATTR.CVAL.IVAL<LMIN)OR (GATTR.CVAL.IVAL               UPDATE7    27
000065                   >LMAX) THEN ERROR(303);                                   UPDATE7    28
000070                  LOAD(GATTR,I)                                              UPDATE7    29
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000072                END                                                    UPDATE7    30
000074               ELSE                                                    UPDATE7    31
000075                BEGIN LOAD(GATTR,I);                                   UPDATE7    32
000100                 IF DEBUG THEN CHECKBNDS(I,LMIN,LMAX,ASSERR)           UPDATE7    33
000104                END                                                    UPDATE7    34
000106              END;                                                     UPDATE7    35
000106             IF FLOAT THEN PACKANDNORM(I);                             UPDATE7    36
000111             STORE(LATTR,I)                                            UPDATE7    37
000113            END;                                                       UPDATE7    38
000116           POINTER,                                                    UPDATE7    39
000116           POWER:                                                      UPDATE7    40
000116            BEGIN LOAD(GATTR,I);                                       UPDATE7    41
000122             STORE(LATTR,I)                                            UPDATE7    42
000124            END;                                                       UPDATE7    43
000127           ARRAYS,                                                     UPDATE7    44
000127           RECORDS:                                                    UPDATE7    45
000127            BEGIN LWORDS := FULLWORDS(LATTR.TYPTR↑.SIZE);              UPDATE7    46
000135             IF LWORDS = 1 THEN                                        UPDATE7    47
000136            IF LATTR.TYPTR↑.FTYPE THEN ERROR( 146 ) ELSE               UPDATE8     6
000143              BEGIN LOAD(GATTR,I); ROTATEX(I); STORE(LATTR,I) END      UPDATE7    48
000156             ELSE                                                      UPDATE7    49
000157              BEGIN                                                    OPT6600A    2
000157               LOADADDRESS( GATTR,I ) ;                                OPT6600A    3
000153               IF I IN [ 0,6,7 ] THEN                                  OPT6600A    4
000166                BEGIN                                                  OPT6600A    5
000166                 NEEDX( 1,5,L ) ;  BXIXJ( L,I ) ;  I := L ;            OPT6600A    6
000177                END ;                                                  OPT6600A    7
000177              LOADADDRESS( LATTR,K ) ;                                 OPT6600A    8
000203              IF K IN [ 0..5 ] THEN                                    OPT6600A    9
000206               BEGIN                                                  OPT6600A   10
000206                NEEDX( 6,7,L ) ;  BXIXJ( L,K ) ;  K := L ;            OPT6600A   11
000215               END ;                                                  OPT6600A   12
000215              NEEDX(1,5,L); NEEDX(6,7,M);                             UPDATE7    53
000225              ARGS[L].ACONT := UNSPECADDR;                            UPDATE7    54
000233              ARGS[M].ACONT := UNSPECADDR;                            UPDATE7    55
000236              IF LWORDS > 4 THEN                                      OPT6600A   13
000240               BEGIN NEEDB(J); GEN30(61B,J,0,LWORDS-1,0); NOOP;       UPDATE7    57
000251                GEN15(53B,L,I,J); GEN15(10B,M,L,0);                   UPDATE7    58
000261                GEN15(53B,M,K,J); GEN15(67B,J,J,1);                   UPDATE7    59
000271                GEN30(06B,J,0,IC-1,2); BRGS[J].BCONT := FREE          UPDATE7    60
000277               END                                                     UPDATE7    61
000304              ELSE (*EXPAND LOOP*)                                     UPDATE7    62
000304               BEGIN                                                  OPT6600A   14
000304                ARGS[I].ACONT := UNSPECADDR ;                         OPT6600A   15
000310                ARGS[K].ACONT := UNSPECADDR ;                         OPT6600A   16
000314                GEN15( 53B,L,I,0 ) ;  GEN15( 53B,I,I,1 ) ;           OPT6600A   17
000322                IF LWORDS > 2 THEN                                    OPT6600A   18
000324                 BEGIN                                                OPT6600A   19
000324                  NEEDX( 1,5,J ) ;  GEN15( 54B,J,I,1 ) ;              OPT6600A   20
000333                  ARGS[J].ACONT := UNSPECADDR ;                       OPT6600A   21
000341                  IF LWORDS > 3 THEN                                  OPT6600A   22
000343                   BEGIN                                              OPT6600A   23
000343                    NEEDX( 1,5,N ) ;  GEN15( 54B,N,J,1 ) ;           OPT6600A   24
000352                    ARGS[N].ACONT := UNSPECADDR ;                     OPT6600A   25
000360                   END ;                                              OPT6600A   26
000360                 END ;                                                OPT6600A   27
```

MOORE BUSINESS FORMS, INC.
FORM 1413
PRINTED IN U.S.A.

```
000350                  GEN15( 10B,M,L,0 ) ;  GEN15( 53B,M,K,0 ) ;          OPT6600A   28
000370                  GEN15( 22B,K,0,I ) ;  GEN15( 54B,K,M,1 ) ;          OPT6600A   29
000400                  IF LWORDS > 2 THEN                                  OPT6600A   30
000402                   BEGIN                                              OPT6600A   31
000402                    GEN15( 10B,M,J,0 ) ;  GEN15( 54B,M,K,1 ) ;        OPT6600A   32
000412                    DECREFX( J ) ;                                    OPT6600A   33
000415                    IF LWORDS > 3 THEN                                OPT6600A   34
000417                     BEGIN                                            OPT6600A   35
000417                      GEN15( 22B,K,0,N ) ;  GEN15( 54B,K,M,1 ) ;      OPT6600A   36
000427                      DECREFX( N ) ;                                  OPT6600A   37
000432                     END ;                                            OPT6600A   38
000432                   END ;                                              OPT6600A   39
000432                 END ;                                                OPT6600A   40
000432                DECREFX(I); DECREFX(K); DECREFX(L); DECREFX(M);        UPDATE7    70
000446                CLEARREGS; (*BE MORE SOPHISTICATED LATER*)            UPDATE7    71
000450               END                                                    UPDATE7    72
000450             END;                                                     UPDATE7    73
000451            FILES: ERROR(146)                                         UPDATE7    74
000452           END                                                        UPDATE7    75
000463          ELSE ERROR(129)                                           UPDATE7    76
000465         END                                                        UPDATE7    77
000466    END (*ASSIGNTO*) ;                                              UPDATE7    78
000472                                                                    UPDATE7    79
000003     PROCEDURE SELECTOR(FSYS: SETOFSYS; FCP: CTP);                  COMP     3363
000005      VAR LATTR: ATTR; LCP: CTP; I: REGNR;                          COMP     3364
000021                                                                    COMP     3365
000003      PROCEDURE IDADDRESS;                                          COMP     3366
000003       VAR I: REGNR;                                                COMP     3367
000004      BEGIN                                                         COMP     3368
000004       WITH FCP↑, LATTR DO                                          COMP     3369
000010        BEGIN TYPTR := IDTYPE; KIND := VARBL;                       COMP     3370
000011         IF TYPTR <> NIL THEN                                       COMP     3371
000013          CASE KLASS OF                                             COMP     3372
000015           VARS:                                                    COMP     3373
000015            BEGIN                                                   COMP     3374
000015             WORDACC := VKIND;                                      COMP     3375
000016             VLEVEL := VLEV; CWDISPL := VADDR;                      COMP     3376
000020             PCKD := FALSE;                                         COMP     3377
000021              IF WORDACC = INDRCT THEN                              COMP     3378
000023               BEGIN WORDACC := DRCT; LOAD(LATTR,I);                COMP     3379
000026                KIND := VARBL; XRGS[I].VPADDR := TRUE;              COMP     3380
000035                WORDACC := INDRCT; CWDISPL := 0;                    COMP     3381
000037                VWDISPL := I; PCKD := FALSE                         COMP     3382
000040               END                                                  COMP     3383
000040            END;                                                    COMP     3384
000041           FIELD:                                                   COMP     3385
000041            WITH DISPLAY[DISX] DO                                   COMP     3386
000043             BEGIN WORDACC := DRCT; VLEVEL := LEV;                  COMP     3387
000045              IF WACC = DRCT THEN CWDISPL := CWDSPL + FLDADDR       COMP     3388
000047              ELSE                                                  COMP     3389
000051               BEGIN CWDISPL := CWDSPL; PCKD := FALSE;              COMP     3390
000053                LOAD(LATTR,I);                                      COMP     3391
000056                KIND := VARBL; WORDACC := INDRCT;                   COMP     3392
000060                CWDISPL := FLDADDR; VWDISPL := I                    COMP     3393
000063               END;                                                 COMP     3394
000064              IF PKD THEN (*IMPLIES (FLDADDR=0)AND PCKDFLD*)        COMP     3395
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000066              BEGIN PCKD := TRUE;                                          COMP     3396
000070               IF BACC = DRCT THEN                                         COMP     3397
000072                BEGIN CBDISPL := BDSPL + BITADDR;                          COMP     3398
000074                 BITREG := NONE                                            COMP     3399
000074                END                                                        COMP     3400
000075               ELSE                                                        COMP     3401
000075                BEGIN                                                      COMP     3402
000075                 WITH GATTR DO                                             COMP     3403
000075                  BEGIN TYPTR := IDTYPE; KIND := VARBL;                    COMP     3404
000100                   WORDACC := DRCT; VLEVEL := LEVEL;                       COMP     3405
000102                   CWDISPL := BDSPL; PCKD := FALSE                         COMP     3406
000103                  END;                                                     COMP     3407
000104                 LOAD(GATTR,I);                                            COMP     3408
000106                 CBDISPL := BITADDR; BITREG := XREG;                       COMP     3409
000112                 VBDISPL := I                                              COMP     3410
000112                END                                                        COMP     3411
000112              END (*PKD*)                                                  COMP     3412
000112             ELSE                                                        COMP     3413
000113              IF PCKDFLD THEN                                            COMP     3414
000115               BEGIN PCKD := TRUE; CBDISPL := BITADDR;                   COMP     3415
000120                BITREG := NONE                                           COMP     3416
000120               END                                                       COMP     3417
000120              ELSE PCKD := FALSE                                         COMP     3418
000121            END (*WITH*) ;                                               COMP     3419
000124          FUNC:                                                          COMP     3420
000124           IF PFDECKIND = STANDARD THEN                                  UPDATE9    35
000126            BEGIN ERROR( 150 ) ;  TYPTR := NIL ;  END                    UPDATE9    36
000132           ELSE                                                          COMP     3422
000132            BEGIN                                                        UPDATE9    37
000132            IF PFDECL IN [EXTDECL,FTNDECL] THEN ERROR(150)               COMP     3423
000136            ELSE                                                         COMP     3424
000140             IF PFKIND = FORMAL THEN ERROR(151)                          COMP     3425
000143             ELSE                                                        COMP     3426
000145              IF PFLEV = LEVEL THEN ERROR( 177 ) ;                       UPDATE9    38
000151               WORDACC := DRCT ;  VLEVEL := PFLEV+1 ;                    UPDATE9    39
000155               CWDISPL := 2; (*IMPLICIT ADDRESS OF FCT RESULT*)          COMP     3428
000156               PCKD := FALSE                                             COMP     3429
000156              END                                                        COMP     3430
000157          END (*CASE*)                                                   COMP     3431
000164        END (*WITH*)                                                     COMP     3432
000164      END (*IDADDRESS*) ;                                              COMP     3433
000170                                                                         COMP     3434
000003      PROCEDURE INDEXCODE;                                             COMP     3435
000003       VAR LBREG: REGKIND; LBITS: BITRANGE; LWORDS: ADDRRANGE;         COMP     3436
000006        LOW,HIGH: INTEGER; LB: BOOLEAN; I,J,K,L: REGNR;                COMP     3437
000015        LACC: ACCESSKIND; LT: INTEGER; LREC: CSTREC;                   COMP     3438
000020        LBOUND: ATTR;                                                  COMP     3439
000032                                                                       COMP     3440
000003       PROCEDURE POWEROF2(I: INTEGER; VAR FB: BOOLEAN;                 COMP     3441
000005                 VAR FEXP: INTEGER);                                   COMP     3442
000006        (*DECIDE WHETHER POSITIVE I IS A POWER OF TWO*)                 COMP     3443
000006        VAR LEXP: INTEGER;                                             COMP     3444
000007       BEGIN LEXP := 0;                                                COMP     3445
000010        WHILE NOT ODD(I) DO                                            COMP     3446
000012         BEGIN I := I DIV 2; LEXP := LEXP + 1 END;                     COMP     3447
000015        FEXP := LEXP; FB := I = 1                                      COMP     3448
```

MOORE BUSINESS FORMS, INC. F 413 PRINTED IN U.S.A 22

```
000017      END (*POWEROF2*) ;                                                  COMP     3449
000025                                                                          COMP     3450
000025     BEGIN (*INDEXCODE*) LACC := DRCT; LBREG := NONE;                     COMP     3451
000010      IF GATTR.KIND <> CST THEN LOAD(GATTR,I);                            COMP     3452
000015      WITH LATTR, TYPTR↑ DO                                               COMP     3453
000017       BEGIN                                                              COMP     3454
000017          GETBOUNDS(INXTYPE,LOW,HIGH);                                    COMP     3455
000022          IF GATTR.KIND = CST THEN                                        COMP     3456
000025           BEGIN IF (GATTR.CVAL.IVAL>HIGH)OR (GATTR.CVAL.IVAL<LOW)        COMP     3457
000027            THEN ERROR(302)                                               COMP     3458
000030           END                                                            COMP     3459
000031          ELSE                                                            COMP     3460
000032           IF DEBUG THEN CHECKBNDS(I,LOW,HIGH,INXERR);                    COMP     3461
000037         IF PCKDARR AND PARTWORDELS THEN (*PARTWORD ACCESS*)               COMP     3462
000042          BEGIN                                                          COMP     3463
000042           IF NOT PCKD THEN                                              COMP     3464
000044            BEGIN PCKD := TRUE; CBDISPL := 0; BITREG := NONE END;        COMP     3465
000046           LBITS := AELTYPE↑.SIZE.BITS;                                  COMP     3466
000051           IF FULLWORDS(SIZE) = 1 THEN                                   COMP     3467
000060            IF GATTR.KIND = CST THEN                                     COMP     3468
000063             CBDISPL := CBDISPL + (GATTR.CVAL.IVAL - LOW)*LBITS          COMP     3469
000064            ELSE                                                         COMP     3470
000070             BEGIN CBDISPL := CBDISPL - LOW*LBITS;                       COMP     3471
000074              EXPREP(LBITS,LREC);                                        COMP     3472
000076              IF LREC.CKIND <> NOP THEN OPTMULT(I,LREC,TRUE,J)           COMP     3473
000103              ELSE                                                       COMP     3474
000106               BEGIN NEEDX(0,7,K); GEN30(718,K,0,LBITS,0);               COMP     3475
000116                OPERATION(42B,J,I,K)                                     COMP     3476
000120               END;                                                       COMP     3477
000122              LBREG := XREG                                               COMP     3478
000122             END                                                          COMP     3479
000123            ELSE                                                          COMP     3480
000124             IF GATTR.KIND = CST THEN                                     COMP     3481
000127              BEGIN CWDISPL := CWDISPL + (GATTR.CVAL.IVAL - LOW)          COMP     3482
000131                    DIV ELSPERWORD;                                       COMP     3483
000136               CBDISPL := CBDISPL + (GATTR.CVAL.IVAL - LOW)               COMP     3484
000136                    MOD ELSPERWORD * LBITS                                COMP     3485
000141              END                                                         COMP     3486
000144             ELSE                                                         COMP     3487
000144              BEGIN                                                       COMP     3488
000144               IF LOW = 0 THEN J := I                                     COMP     3489
000146               ELSE                                                       COMP     3490
000147                BEGIN                                                     COMP     3491
000147                 WITH LBOUND DO                                           COMP     3492
000147                  BEGIN TYPTR := INTPTR; KIND := CST;                     COMP     3493
000151                  CVAL.IVAL := LOW                                        COMP     3494
000151                 END;                                                     COMP     3495
000152                 LOAD(LBOUND,J); OPERATION(37B,J,I,J)                     UPDATE1    10
000156                END;                                                      COMP     3497
000160              POWEROF2(ELSPERWORD,LB,LT);                                 COMP     3498
000164              NEEDX(0,7,K);                                               COMP     3499
000170              IF LB THEN                                                  COMP     3500
000171               BEGIN GEN15(10B,K,J,0); GEN15(21B,K,0,LT);                 COMP     3501
000201               END                                                        COMP     3502
000201              ELSE                                                        COMP     3503
000202               BEGIN                                                      COMP     3504
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000202              IF ELSPERWORD IN [3,6,12] THEN LT := 87382                    COMP  3505
000206              ELSE                                                          COMP  3506
000207               IF ELSPERWORD IN [5,10,20] THEN LT := 52429                  COMP  3507
000212               ELSE                                                        COMP  3508
000213                IF ELSPERWORD = 7 THEN LT := 74899                         COMP  3509
000215                ELSE LT := 69906;                                          COMP  3510
000220              GEN30(71B,K,0,LT,0); GEN15(42B,K,J,K);                       COMP  3511
000230              IF ELSPERWORD <= 5 THEN LT := 18                             COMP  3512
000233              ELSE                                                        COMP  3513
000235               IF ELSPERWORD <= 10 THEN LT := 19                          COMP  3514
000237               ELSE                                                       COMP  3515
000241                IF ELSPERWORD <= 20 THEN LT := 20                         COMP  3516
000243                ELSE                                                      COMP  3517
000244                 IF ELSPERWORD = 30 THEN LT := 21                         COMP  3518
000246                 ELSE LT := 22;                                           COMP  3519
000251              GEN15(21B,K,0,LT)                                           COMP  3520
000253             END;                                                         COMP  3521
000255            IF LB THEN                                                    COMP  3522
000256             BEGIN NEEDX(0,7,L); GEN15(10B,L,K,0);                        COMP  3523
000266              GEN15(20B,L,0,LT)                                           COMP  3524
000270             END                                                          COMP  3525
000272            ELSE                                                          COMP  3526
000273             BEGIN EXPREP(ELSPERWORD,LREC);                               COMP  3527
000277              IF LREC.CKIND <> NOP THEN                                   COMP  3528
000302               BEGIN OPTMULT(K,LREC,FALSE,L);                             COMP  3529
000305                (*RESET REFERENCE:*)                                      COMP  3530
000305                 WITH XRGS[K] DO                                          COMP  3531
000311                  BEGIN XCONT := OTHER; REFNR := 1 END;                   COMP  3532
000316                END                                                       COMP  3533
000316               ELSE                                                       COMP  3534
000317                BEGIN NEEDX(0,7,L); GEN30(71B,L,0,ELSPERWORD,0);          COMP  3535
000330                 GEN15(42B,L,K,L)                                         COMP  3536
000332               END                                                        COMP  3537
000334              END;                                                        COMP  3538
000334             DECREFX(J); I := J; NEEDX(0,7,J); GEN15(37B,J,I,L);          COMP  3539
000347             EXPREP(LBITS,LREC);                                         COMP  3540
000353             IF LREC.CKIND <> NOP THEN OPTMULT(J,LREC,TRUE,J)            COMP  3541
000357             ELSE                                                        COMP  3542
000352              BEGIN GEN30(71B,L,0,LBITS,0); GEN15(42B,J,J,L) END;        COMP  3543
000373             DECREFX(L);                                                 COMP  3544
000376             LACC := INXD; LBREG := XREG                                 COMP  3545
000377            END                                                          COMP  3546
000400          END (*PCKDARR AND PARTWORDELS*)                                COMP  3547
000400         ELSE                                                            COMP  3548
000401          BEGIN                                                          COMP  3549
000401            LWORDS := FULLWORDS(AELTYPE↑.SIZE);                          COMP  3550
000406            IF GATTR.KIND = CST THEN                                     COMP  3551
000411             CWDISPL := CWDISPL + (GATTR.CVAL.IVAL - LOW)*LWORDS         COMP  3552
000412            ELSE                                                         COMP  3553
000415             BEGIN CWDISPL := CWDISPL - LOW*LWORDS;                      COMP  3554
000420              EXPREP(LWORDS,LREC);                                       COMP  3555
000423              IF LREC.CKIND <> NOP THEN OPTMULT(I,LREC,TRUE,K)           COMP  3556
000430              ELSE                                                       COMP  3557
000433               BEGIN NEEDX(0,7,J); GEN30(71B,J,0,LWORDS,0);              COMP  3558
000443                OPERATION(42B,K,I,J)                                     COMP  3559
000445               END;                                                       COMP  3560
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A.

```
000447                LACC := INXC                                          COMP  3561
000447               END;                                                   COMP  3562
000450             END;                                                     COMP  3563
000450            IF LACC <> DRCT THEN                                      COMP  3564
000452             IF WORDACC = DRCT THEN                                   COMP  3565
000454              BEGIN VWDISPL := K; WORDACC := INXD;                    COMP  3566
000456              END                                                     COMP  3567
000456             ELSE                                                     COMP  3568
000456              BEGIN OPERATION(36B,L,VWDISPL,K); VWDISPL := L END;     COMP  3569
000465            IF LBREG <> NONE THEN                                     COMP  3570
000467             IF BITREG = NONE THEN                                    COMP  3571
000471              BEGIN BITREG := XREG; VBDISPL := J END                  COMP  3572
000472             ELSE                                                     COMP  3573
000473              BEGIN OPERATION(36B,L,VBDISPL,J); VBDISPL := L END      COMP  3574
000501           END (*WITH LATTR*)                                         COMP  3575
000501         END (*INDEXCODE*) ;                                          COMP  3576
000506                                                                      COMP  3577
000506       BEGIN (*SELECTOR*)                                             COMP  3578
000506        IDADDRESS;                                                    COMP  3579
000011        IF NOT (SY IN SELECTSYS+FSYS) THEN                            COMP  3580
000014         BEGIN ERROR(59); SKIP(SELECTSYS+FSYS) END;                   COMP  3581
000021        WHILE SY IN SELECTSYS DO                                      COMP  3582
000024         BEGIN                                                        COMP  3583
000024 (*[*)    IF SY = LBRACK THEN                                        COMP  3584
000025           BEGIN                                                      COMP  3585
000025            REPEAT                                                    COMP  3586
000025             WITH LATTR DO                                            COMP  3587
000025              IF TYPTR <> NIL THEN                                    COMP  3588
000027               IF TYPTR↑.FORM <> ARRAYS THEN                          COMP  3589
000032                BEGIN ERROR(138); TYPTR := NIL END;                   COMP  3590
000034              INSYMBOL; EXPRESSION(FSYS+[COMMA,RBRACK]);              COMP  3591
000040              IF GATTR.TYPTR <> NIL THEN                              COMP  3592
000043               IF GATTR.TYPTR↑.FORM > SUBRANGE THEN ERROR(113);       COMP  3593
000047              IF LATTR.TYPTR <> NIL THEN                              COMP  3594
000051               WITH LATTR.TYPTR↑ DO                                   COMP  3595
000052                BEGIN                                                 COMP  3596
000052                 IF COMPTYPES(INXTYPE,GATTR.TYPTR,FALSE) THEN         COMP  3597
000057                  BEGIN                                               COMP  3598
000057                   IF (INXTYPE <> NIL)AND (AELTYPE <> NIL) THEN INDEXCODE COMP  3599
000065                  END                                                 COMP  3600
000066                 ELSE ERROR(139);                                     COMP  3601
000071                 LATTR.TYPTR := AELTYPE                               COMP  3602
000071                END                                                   COMP  3603
000073              UNTIL SY <> COMMA;                                      COMP  3604
000075              IF SY = RBRACK THEN INSYMBOL ELSE ERROR(12)             COMP  3605
000102            END (*IF SY = LBRACK*)                                    COMP  3606
000103          ELSE                                                        COMP  3607
000104 (*.*)      IF SY = PERIOD THEN                                       COMP  3608
000106            BEGIN                                                     COMP  3609
000106             WITH LATTR DO                                            COMP  3610
000106              BEGIN                                                   COMP  3611
000106               IF TYPTR <> NIL THEN                                   COMP  3612
000110                IF TYPTR↑.FORM <> RECORDS THEN                        COMP  3613
000112                 BEGIN ERROR(140); TYPTR := NIL END;                  COMP  3614
000115               INSYMBOL;                                              COMP  3615
000116               IF SY = IDENT THEN                                     COMP  3616
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000120                BEGIN                                                    COMP  3617
000120                 IF TYPTR <> NIL THEN                                    COMP  3618
000122                  BEGIN SEARCHSECTION(TYPTR↑.FIELDS,LCP);                COMP  3619
000124                   IF LCP = NIL THEN                                     COMP  3620
000126                    BEGIN ERROR(152); TYPTR := NIL END                   COMP  3621
000131                   ELSE                                                  COMP  3622
000132                    WITH LCP↑ DO                                         COMP  3623
000133                     BEGIN TYPTR := IDTYPE;                              COMP  3624
000134                      IF PCKD THEN (*IMPLIES (FLDADDR=0)AND PCKDFLD*)    COMP  3625
000135                       CBDISPL := CBDISPL + BITADDR                      COMP  3626
000135                      ELSE                                               COMP  3627
000140                       BEGIN CWDISPL := CWDISPL + FLDADDR;               COMP  3628
000143                        IF PCKDFLD THEN                                  COMP  3629
000144                         BEGIN PCKD := TRUE; BITREG := NONE;             COMP  3630
000145                          CBDISPL := BITADDR                             COMP  3631
000145                         END                                             COMP  3632
000146                       END                                               COMP  3633
000146                     END                                                 COMP  3634
000146                  END;                                                   COMP  3635
000146                 INSYMBOL                                                COMP  3636
000146                END (*SY = IDENT*)                                       COMP  3637
000147               ELSE ERROR(2)                                             COMP  3638
000151             END (*WITH GATTR*)                                          COMP  3639
000152           END (*IF SY = PERIOD*)                                        COMP  3640
000152         ELSE                                                            COMP  3641
000153 (*↑*)     BEGIN                                                        COMP  3642
000153            IF LATTR.TYPTR <> NIL THEN                                   COMP  3643
000155             BEGIN                                                       COMP  3644
000155              WITH LATTR DO                                              COMP  3645
000155               BEGIN                                                     COMP  3646
000155                IF TYPTR↑.FORM = FILES THEN                              COMP  3647
000160                 IF TYPTR↑.TEXTFILE THEN                                 COMP  3648
000161                  CWDISPL := CWDISPL + CHEFET - 1                        COMP  3649
000163                 ELSE CWDISPL := CWDISPL + BINEFET - 1;                  COMP  3650
000166                LOAD(LATTR,I);                                           COMP  3651
000171                WITH TYPTR↑ DO                                           COMP  3652
000173                 IF FORM = POINTER THEN TYPTR := ELTYPE                  COMP  3653
000175                 ELSE                                                    COMP  3654
000176                  IF FORM = FILES THEN TYPTR := FILTYPE                  COMP  3655
000200                  ELSE ERROR(141);                                       COMP  3656
000204                KIND := VARBL; WORDACC := INDRCT;                        COMP  3657
000206                CWDISPL := 0; VWDISPL := I;                              COMP  3658
000207                PCKD := FALSE                                            COMP  3659
000207               END                                                       COMP  3660
000210             END;                                                        COMP  3661
000210            INSYMBOL                                                     COMP  3662
000210          END;                                                           COMP  3663
000211        IF NOT (SY IN FSYS+SELECTSYS) THEN                               COMP  3664
000214         BEGIN ERROR(6); SKIP(FSYS+SELECTSYS) END                        COMP  3665
000221       END (*WHILE*) ;                                                   COMP  3666
000222      GATTR := LATTR;                                                    COMP  3667
000227    END (*SELECTOR*) ;                                                   COMP  3668
000233                                                                         COMP  3669
000003    PROCEDURE CALL(FSYS: SETOFSYS; FCP: CTP);                            COMP  3670
000005     VAR LKEY: 1..25; I,J,K: REGNR;                                      COMP  3671
000011                                                                         COMP  3672
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000003      PROCEDURE VARIABLE(FSYS: SETOFSYS);                                  COMP  3673
000004       VAR LCP: CTP;                                                       COMP  3674
000005      BEGIN                                                                COMP  3675
000005       IF SY = IDENT THEN                                                  COMP  3676
000010        BEGIN SEARCHID([VARS,FIELD],LCP); INSYMBOL END                     COMP  3677
000013       ELSE BEGIN ERROR(2); LCP := UVARPTR END;                            COMP  3678
000020       SELECTOR(FSYS,LCP)                                                  COMP  3679
000022      END (*VARIABLE*) ;                                                   COMP  3680
000027                                                                           COMP  3681
000003      PROCEDURE STDFLPROCS;                                                COMP  3682
000003       VAR CHARFILE,SEGMFILE: BOOLEAN; NAME: ALFA; I: REGNR;               COMP  3683
000007         LDPLMT: ADDRRANGE;                                                COMP  3684
000010      BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                   COMP  3685
000013       CLEARREGS; XRGS[1].XCONT := OTHER; (*RESERVE A1/X1*)                COMP  3686
000022       VARIABLE(FSYS+[COMMA,RPARENT]);                                     COMP  3687
000025       CLEARREGS; (*TO PREVENT BX1... AND GUARANTEE SA1...*)               COMP  3688
000027       CHARFILE := FALSE; SEGMFILE := FALSE;  LDPLMT := BINEFET;           COMP  3689
000031       IF GATTR.TYPTR <> NIL THEN                                          COMP  3690
000035        BEGIN                                                              COMP  3691
000035         IF GATTR.WORDACC <> DRCT THEN                                     COMP  3692
000036          XRGS[GATTR.VWDISPL].XCONT := OTHER;                              COMP  3693
000043         WITH GATTR.TYPTR↑ DO                                              COMP  3694
000046          IF FORM = FILES THEN                                             COMP  3695
000051           BEGIN CHARFILE := TEXTFILE; SEGMFILE := SEGFILE;                COMP  3696
000053            IF CHARFILE THEN LDPLMT := CHEFET                              COMP  3697
000054           END                                                             COMP  3698
000055          ELSE ERROR(116)                                                  COMP  3699
000057        END                                                                COMP  3700
000060      ELSE GATTR.CWDISPL := 0;                                             COMP  3701
000062      CASE LKEY OF                                                         COMP  3702
000064       (*GET*)      1: IF CHARFILE THEN                                    COMP  3703
000065                        BEGIN NAME := =P.GETC    =; LDPLMT := CHEFET-1     COMP  3704
000066                        END                                                COMP  3705
000070                       ELSE NAME := =P.GETB    =;                         COMP  3706
000072       (*PUT*)      2: IF CHARFILE THEN                                    COMP  3707
000073                        BEGIN NAME := =P.PUTC    =; LDPLMT := CHEFET - 1   COMP  3708
000074                        END                                                COMP  3709
000076                       ELSE NAME := =P.PUTB    =;                         COMP  3710
000100       (*RESET*)    3: NAME := =P.RESET   =;                               COMP  3711
000102       (*REWRITE*)  4: NAME := =P.REWRT   =;                               COMP  3712
000104       (*GETSEG*)   5,                                                     COMP  3713
000104       (*PUTSEG*)   6: BEGIN IF NOT SEGMFILE THEN ERROR(116);              COMP  3714
000107                        IF LKEY = 5 THEN NAME := =P.GETS    =              COMP  3715
000112                        ELSE NAME := =P.PUTS    =                          COMP  3716
000114                       END                                                 COMP  3717
000116      END (*CASE*) ;                                                       COMP  3718
000125      GATTR.CWDISPL := GATTR.CWDISPL + LDPLMT;                             COMP  3719
000130      LOAD(GATTR,I);                                                       COMP  3720
000133      IF SY = COMMA THEN                                                   COMP  3721
000135       BEGIN                                                               COMP  3722
000135        IF LKEY = 4 (*REWRITE*) THEN                                       COMP  3723
000140         BEGIN IF NOT SEGMFILE THEN ERROR(126);                            COMP  3724
000143          NAME := =P.RWRTS   =                                             COMP  3725
000143         END                                                               COMP  3726
000145        ELSE                                                               COMP  3727
000145         IF LKEY <> 5 (*GETSEG*) THEN ERROR(126);                          COMP  3728
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000150          INSYMBOL;                                                          COMP     3729
000151          EXPRESSION(FSYS+[RPARENT]);                                        COMP     3730
000155          IF NOT COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN ERROR(127);        COMP     3731
000165          LOAD(GATTR,I); BXIXJ(2,I)                                          COMP     3732
000172         END                                                                 COMP     3733
000174        ELSE                                                                 COMP     3734
000175         IF LKEY = 5 THEN GEN15(76B,2,1,0);                                  COMP     3735
000203        CLEARREGS;                                                           COMP     3736
000205        SEARCHEXTID(NAME,EXT); RJTOEXT;                                      COMP     3737
000213        IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                          COMP     3738
000220      END (*STDFLPROCS*) ;                                                   COMP     3739
000236                                                                             COMP     3740
000003      PROCEDURE LINELIMIT;                                                   COMP     3741
000003       VAR LATTR: ATTR;                                                      COMP     3742
000015      BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                     COMP     3743
000013       VARIABLE(FSYS+[COMMA,RPARENT]);                                       COMP     3744
000016       IF GATTR.TYPTR <> NIL THEN                                            COMP     3745
000022        WITH GATTR.TYPTR↑ DO                                                 COMP     3746
000023         IF FORM = FILES THEN                                                COMP     3747
000025          BEGIN IF NOT TEXTFILE THEN ERROR(116)                              COMP     3748
000027          END                                                                COMP     3749
000030         ELSE ERROR(116);                                                    COMP     3750
000033       LATTR := GATTR;                                                       COMP     3751
000040       IF SY = COMMA THEN                                                    COMP     3752
000042        BEGIN INSYMBOL; EXPRESSION(FSYS+[RPARENT]);                          COMP     3753
000047         IF NOT COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN ERROR(116);        COMP     3754
000057         LOAD(GATTR,I); STORE(LATTR,I)                                       COMP     3755
000066        END                                                                  COMP     3756
000070       ELSE ERROR(20);                                                       COMP     3757
000073       IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                           COMP     3758
000100      END (*LINELIMIT*) ;                                                    COMP     3759
000105                                                                             COMP     3760
000003      PROCEDURE LOADPPTR( FATTR : ATTR ;  FDRCT : BOOLEAN ) ;                UPDATE7    80
000017       (*LOAD PPTR (ADDRESS) OF FILE DESCRIBED BY <FATTR,FDRCT> INTO         UPDATE7    81
000017         X1 (A1); UPON ENTRY, A1,X1 HAVE TO BE RESERVED*)                    UPDATE7    82
000017       VAR  I : REGNR ;  L : ADDRRANGE ;                                     UPDATE7    83
000021      BEGIN                                                                  UPDATE7    84
000021       WITH FATTR DO                                                         UPDATE7    85
000013        BEGIN IF TYPTR↑.TEXTFILE THEN L := CHEFET-1 ELSE L := BINEFET-1 ;    UPDATE7    86
000020         IF FDRCT THEN                                                       UPDATE7    87
000021          BEGIN CWDISPL := CWDISPL+L ;                                       UPDATE7    88
000023           IF VLEVEL IN LEVELS THEN GEN30(518,1,BRG[VLEVEL],CWDISPL,0)       UPDATE7    89
000031           ELSE BEGIN DECREFX(1); (*TO FORCE LOADING OF A1/X1*)               UPDATE7    90
000035                LOADBASE(VLEVEL,TRUE,I); GEN30(528,1,1,CWDISPL,0)            UPDATE7    91
000043               END ;                                                         UPDATE7    92
000045          END                                                                UPDATE7    93
000045         ELSE BEGIN LOAD(FATTR,I); DECREFX(I); GEN30(52B,1,I,L,0); END ;     UPDATE7    94
000060        END ;                                                                UPDATE7    95
000060       ARGS[ 1 ].ACONT := UNSPECADDR ;                                       UPDATE7    96
000065      END (*LOADPPTR*) ;                                                     UPDATE7    97
000070                                                                             UPDATE7    98
000003      PROCEDURE READ;                                                        COMP     3761
000003       VAR PARAM,LATTR,FILATTR: ATTR; I,J: REGNR;                            COMP     3762
000043        LDRCT,GETIN,EXITLOOP: BOOLEAN;                                       COMP     3763
000046                                                                             COMP     3764
000046      BEGIN (*READ*)                                                         COMP     3778
```

MOORE BUSINESS FORMS, INC. F
FORM 1413
PRINTED IN U.S.A
22

```
000046      (*SET DEFAULT FILE ATTRIBUTES:*)                                  COMP     3779
000046      WITH FILATTR DO                                                   COMP     3780
000006       BEGIN TYPTR := TEXTPTR; KIND := VARBL; WORDACC := DRCT;          COMP     3781
000010        VLEVEL := 1; PCKD := FALSE;                                     COMP     3782
000011        IF INPUTPTR = NIL THEN CWDISPL := 0                             COMP     3783
000013        ELSE CWDISPL := INPUTPTR^.VADDR                                 COMP     3784
000014       END;                                                             COMP     3785
000016      LDRCT := TRUE;                                                    COMP     3786
000017      (*SET PARAMETER ATTRIBUTES:*)                                     COMP     3787
000017      WITH PARAM DO                                                     COMP     3788
000017       BEGIN TYPTR := TEXTPTR; KIND := VARBL; WORDACC := DRCT;          COMP     3789
000022        VLEVEL := 0; CWDISPL := 3; PCKD := FALSE                        COMP     3790
000023       END;                                                             COMP     3791
000024      NEEDX( 1,1,I ) ;  (*RESERVE A1/X1*)                               UPDATE8    8
000027      IF SY = LPARENT THEN                                              COMP     3792
000031       BEGIN  GETIN := TRUE ;                                           UPDATE8    9
000032        INSYMBOL; VARIABLE(FSYS+[COMMA,RPARENT]);                       COMP     3794
000036        IF GATTR.TYPTR <> NIL THEN                                      COMP     3795
000042         IF GATTR.TYPTR^.FORM = FILES THEN                              COMP     3796
000044          BEGIN IF NOT GATTR.TYPTR^.TEXTFILE AND (LKEY = 9) (*READLN*)  UPDATE7   100
000047            THEN ERROR(116) ;                                           UPDATE7   101
000052           IF GATTR.WORDACC = DRCT THEN                                 COMP     3798
000055            BEGIN FILATTR := GATTR; LDRCT := TRUE END                   COMP     3799
000052           ELSE                                                         COMP     3800
000062            BEGIN LOADADDRESS(GATTR,I);                                 COMP     3801
000065             WITH FILATTR DO                                            COMP     3802
000065              BEGIN TYPTR := GATTR.TYPTR ;                              UPDATE7   102
000070               VLEVEL := LEVEL ;  CWDISPL := LC ;                       UPDATE7   103
000073              END ;                                                     UPDATE7   104
000073             STORE(FILATTR,I);                                          COMP     3804
000075             LC := LC + 1;                                              COMP     3805
000077             IF LC > LCMAX THEN LCMAX := LC;                            COMP     3806
000102             LDRCT := FALSE                                             COMP     3807
000102            END;                                                        COMP     3808
000103          IF SY = RPARENT THEN                                          COMP     3809
000105           BEGIN IF LKEY = 8 (*READ*) THEN ERROR(116);                  COMP     3810
000111            GETIN := FALSE                                              COMP     3811
000111           END                                                          COMP     3812
000112          ELSE                                                          COMP     3813
000113           IF SY = COMMA THEN                                           COMP     3814
000115            BEGIN INSYMBOL; VARIABLE(FSYS+[COMMA,RPARENT]) END          COMP     3815
000121          END (*FORM = FILES*)                                          COMP     3816
000121         ELSE IF INPUTPTR = NIL THEN ERROR(175);                        COMP     3817
000125        IF GETIN THEN                                                   COMP     3818
000126         (*LOOP UNTIL SY <> COMMA:*)                                    COMP     3819
000126         REPEAT                                                         COMP     3820
000126          IF FILATTR.TYPTR^.TEXTFILE AND NOT COMPTYPES( GATTR.TYPTR,    UPDATE7   105
000132           CHARPTR,FALSE ) THEN                                         UPDATE7   106
000140           BEGIN DECREFX(1) ;                                           UPDATE7   107
000142           IF COMPTYPES(GATTR.TYPTR,INTPTR,FALSE)OR COMPTYPES(          COMP     3830
000150            GATTR.TYPTR,REALPTR,FALSE) THEN                             COMP     3831
000157            BEGIN                                                       COMP     3832
000157             IF GATTR.PCKD THEN ERROR( 127 ) ;                          UPDATE5    19
000163             (*PASS FILE ADDRESS:*)                                     COMP     3833
000163             LATTR := FILATTR;                                          COMP     3834
000167             IF LDRCT THEN LOADADDRESS(LATTR,J) ELSE LOAD(LATTR,J);     COMP     3835
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000177          PARAM.CWDISPL := 3; STORE(PARAM,J);                              COMP     3836
000203          (*PASS ADDRESS OF VARIABLE TO BE READ:*)                         COMP     3837
000203          LOADADDRESS(GATTR,I);                                            COMP     3838
000207          PARAM.CWDISPL := 4; STORE(PARAM,I);                              COMP     3839
000213          CLEARREGS; GEN30(71B,7,0,IC+1,2);                                COMP     3840
000222          IF COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN                      COMP     3841
000232           SEARCHEXTID(=RDI       =,EXT)                                   COMP     3842
000234          ELSE SEARCHEXTID(=RDR       =,EXT);                              COMP     3843
000243          GEN30(048,0,0,0,0); NOOP                                         COMP     3844
000247         END                                                               COMP     3845
000251        ELSE ERROR(116);                                                   COMP     3846
000254        END                                                                UPDATE7   108
000254       ELSE                                                                UPDATE7   109
000255        BEGIN LATTR := GATTR ; LOADPPTR(FILATTR,LDRCT) ;                   UPDATE7   110
000264         WITH GATTR DO                                                     UPDATE7   111
000254          BEGIN TYPTR := FILATTR.TYPTR^.FILTYPE ; KIND := VARBL ;          UPDATE7   112
000270           WORDACC := INDRCT ; CWDISPL := 0 ; VWDISPL := 1 ;               UPDATE7   113
000272           PCKD := FALSE ;                                                 UPDATE7   114
000273          END ;                                                            UPDATE7   115
000273        XRGS[1].REFNR := 2 ; (*TO PROTECT A1/X1*)                          UPDATE7   116
000276        ASSIGNTO(LATTR) ;                                                  UPDATE7   117
000277        NEEDX(1,1,I) ; (*RESET REFERENCE TO X1*)                           UPDATE7   118
000302        IF FILATTR.TYPTR^.TEXTFILE THEN                                    UPDATE7   119
000304         SEARCHEXTID( =P.GETC    =,EXT)                                   UPDATE7   120
000306        ELSE BEGIN GEN15( 548,1,1,1 ) ;                                    UPDATE7   121
000315             SEARCHEXTID( =P.GETB    =,EXT ) ;                           UPDATE7   122
000321           END ;                                                           UPDATE7   123
000321         RJTOEXT ; CLEARREGS ;                                             UPDATE7   124
000325        END ;                                                              UPDATE7   125
000325        EXITLOOP := SY <> COMMA;                                           COMP     3847
000330        IF NOT EXITLOOP THEN                                               COMP     3848
000331         BEGIN NEEDX(1,1,I); (*RESERVE A1/X1*)                             UPDATE7   126
000334          INSYMBOL; VARIABLE( FSYS+[COMMA,RPARENT] ) ;                     UPDATE7   127
000340         END                                                               UPDATE7   128
000340       UNTIL EXITLOOP;                                                     COMP     3850
000341      IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                          COMP     3851
000346      END (*SY = LPARENT*)                                                 COMP     3852
000347     ELSE                                                                  COMP     3853
000350      IF LKEY = 8 THEN (*READ*) ERROR(116)                                 COMP     3854
000353      ELSE                                                                 COMP     3855
000355       IF INPUTPTR = NIL THEN ERROR(175);                                  COMP     3856
000361      IF LKEY = 9 (*READLN*) THEN                                          COMP     3857
000354       BEGIN LOADPPTR( FILATTR,LDRCT ) ;                                   UPDATE7   129
000366        SEARCHEXTID(=P.GETLN    =,EXT); RJTOEXT;                          COMP     3859
000374        CLEARREGS                                                          COMP     3860
000374       END;                                                                COMP     3861
000376     IF NOT LDRCT THEN LC := LC - 1                                        COMP     3862
000400    END (*READ*) ;                                                         COMP     3863
000412                                                                           COMP     3864
000003    PROCEDURE WRITE;                                                       COMP     3865
000003     VAR PARAM,LATTR,FILATTR: ATTR;                                        COMP     3866
000041     LSP: STP; I,J: REGNR; LDEF,LDRCT,PUTOUT,EXITLOOP: BOOLEAN;            COMP     3867
000050                                                                           COMP     3868
000003     PROCEDURE DEFLJMP(FL: INTEGER; FNAME: ALFA);                          COMP     3869
000005      (*IF DEFAULF PASS DEFAULF FIELDLENGTH; JUMP TO EXTERNAL FNAME*)      COMP     3870
000005       VAR I: REGNR;                                                       COMP     3871
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000006        BEGIN                                                                  COMP     3872
000006         IF LDEF THEN                                                          COMP     3873
000011          BEGIN NEEDX(6,7,I); GEN30(71B,I,0,FL,0);                             COMP     3874
000021           PARAM.CWDISPL := 5; STORE(PARAM,I)                                  COMP     3875
000024          END;                                                                  COMP     3876
000026         CLEARREGS; GEN30(71B,7,0,IC+1,2);                                     COMP     3877
000036         SEARCHEXTID(FNAME,EXT); GEN30(04B,0,0,0,0); NOOP                      COMP     3878
000046        END (*DEFLJMP*) ;                                                      COMP     3879
000055                                                                               COMP     3880
000055       BEGIN (*WRITE*)                                                         COMP     3896
000055        (*SET DEFAULT FILE ATTRIBUTES:*)                                       COMP     3897
000055        WITH FILATTR DO                                                        COMP     3898
000006         BEGIN TYPTR := TEXTPTR; KIND := VARBL; WORDACC := DRCT;               COMP     3899
000010          VLEVEL := 1; CWDISPL := OUTPUTPTR↑.VADDR; PCKD := FALSE              COMP     3900
000013         END;                                                                  COMP     3901
000014        LDRCT := TRUE;                                                         COMP     3902
000015        (*SET PARAMETER ATTRIBUTES:*)                                          COMP     3903
000015        WITH PARAM DO                                                          COMP     3904
000015         BEGIN TYPTR := TEXTPTR; KIND := VARBL; WORDACC := DRCT;               COMP     3905
000017          VLEVEL := 0; CWDISPL := 3; PCKD := FALSE                             COMP     3906
000020         END;                                                                  COMP     3907
000021        NEEDX( 1,1,I ) ;  (*RESERVE A1/X1*)                                   UPDATE8    10
000024        IF SY = LPARENT THEN                                                   COMP     3908
000026         BEGIN  PUTOUT := TRUE ;                                               UPDATE8    11
000027          INSYMBOL; EXPRESSION(FSYS+[COMMA,COLON,RPARENT,IDENT]);              COMP     3910
000034          IF GATTR.TYPTR <> NIL THEN                                           COMP     3911
000040           IF GATTR.TYPTR↑.FORM = FILES THEN                                   COMP     3912
000042            BEGIN IF NOT GATTR.TYPTR↑.TEXTFILE AND (LKEY = 11) (*WRITELN*)     UPDATE7    131
000045             THEN ERROR(115) ;                                                 UPDATE7    132
000050            IF SY = RPARENT THEN                                               COMP     3914
000052             BEGIN IF LKEY = 10 (*WRITE*) THEN ERROR(116);                     COMP     3915
000055              PUTOUT := FALSE                                                  COMP     3916
000055             END                                                               COMP     3917
000056            ELSE                                                               COMP     3918
000057             IF SY <> COMMA THEN                                               COMP     3919
000061              BEGIN ERROR(116); SKIP(FSYS+[COMMA,RPARENT]) END;                COMP     3920
000065            IF GATTR.WORDACC = DRCT THEN                                       COMP     3921
000070             BEGIN FILATTR := GATTR; LDRCT := TRUE END                         COMP     3922
000075            ELSE                                                               COMP     3923
000075             BEGIN LOADADDRESS(GATTR,I);                                       COMP     3924
000100              WITH FILATTR DO                                                  COMP     3925
000100               BEGIN TYPTR := GATTR.TYPTR ;                                    UPDATE7    133
000103                VLEVEL := LEVEL ; CWDISPL := LC ;                              UPDATE7    134
000106               END ;                                                           UPDATE7    135
000106              STORE(FILATTR,I);                                                COMP     3927
000110              LC := LC + 1;                                                    COMP     3928
000112              IF LC > LCMAX THEN LCMAX := LC;                                  COMP     3929
000115              LDRCT := FALSE                                                   COMP     3930
000115             END;                                                              COMP     3931
000116            IF SY = COMMA THEN                                                 COMP     3932
000120             BEGIN INSYMBOL;                                                   COMP     3933
000121              EXPRESSION(FSYS+[COMMA,COLON,RPARENT,IDENT])                     COMP     3934
000123             END                                                               COMP     3935
000125           END (*FORM = FILES*) ;                                              COMP     3936
000125         IF PUTOUT THEN                                                        COMP     3937
000126          (*LOOP UNTIL SY <> COMMA*)                                           COMP     3938
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000126          REPEAT                                                          COMP     3939
000126           IF FILATTR.TYPTR↑.TEXTFILE AND (NOT COMPTYPES(GATTR.TYPTR,     UPDATE7   136
000132            CHARPTR,FALSE) OR (SY = COLON)) THEN                          UPDATE7   137
000143           BEGIN LSP := GATTR.TYPTR ; DECREFX(1) ;                        UPDATE7   138
000147             (*PASS FILEADDRESS:*)                                        COMP     3949
000147             LATTR := FILATTR;                                            COMP     3950
000153             IF LDRCT THEN LOADADDRESS(LATTR,J)                           COMP     3951
000155             ELSE LOAD(LATTR,J);                                          COMP     3952
000153             PARAM.CWDISPL := 3; STORE(PARAM,J);                          COMP     3953
000167             (*PASS VALUE TO BE OUTPUT:*)                                 COMP     3954
000167             IF STRING(LSP) THEN LOADADDRESS(GATTR,I)                     COMP     3955
000200             ELSE LOAD(GATTR,I);                                          COMP     3956
000206             PARAM.CWDISPL := 4; STORE(PARAM,I);                          COMP     3957
000212             IF SY = COLON THEN                                           COMP     3958
000214              BEGIN  INSYMBOL ;  B5DPL := 5 ;                             UPDATE5    20
000216               EXPRESSION(FSYS+[COMMA,COLON,RPARENT,IDENT]);              COMP     3960
000222               IF NOT COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN            COMP     3961
000230                ERRCR(116);                                               COMP     3962
000232               (*PASS FILEDLENGTH:*)                                      COMP     3963
000232               LOAD(GATTR,I);                                             COMP     3964
000236               PARAM.CWDISPL := 5; STORE(PARAM,I);                        COMP     3965
000242               LDEF := FALSE                                              COMP     3966
000242              END                                                         COMP     3967
000243             ELSE LDEF := TRUE;                                           COMP     3968
000245             IF SY = COLON THEN                                           COMP     3969
000247              BEGIN  INSYMBOL ;  B5DPL := 6 ;                             UPDATE5    21
000251              EXPRESSION( FSYS+[ COMMA,RPARENT ] ) ;                      UPDATE5    22
000255              IF NOT COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN             COMP     3971
000263               ERRCR(116);                                                COMP     3972
000265              IF NOT COMPTYPES(LSP,REALPTR,FALSE) THEN ERROR(124);        COMP     3973
000275              (*PASS FRACTION LENGTH:*)                                   COMP     3974
000275              LOAD(GATTR,I);                                              COMP     3975
000301              PARAM.CWDISPL := 6; STORE(PARAM,I);                         COMP     3976
000305              DEFLJMP(20,=WRF       =)                                    COMP     3977
000306             END                                                          COMP     3978
000310            ELSE                                                          COMP     3979
000311             IF SY = IDENT THEN                                           COMP     3980
000312              BEGIN                                                       COMP     3981
000312               IF ( ID = =OCT       = ) OR ( ID = =HEX       = ) THEN     HEXOUT     1
000317                BEGIN                                                     COMP     3983
000317                 IF LSP <> NIL THEN                                       COMP     3984
000321                  IF LSP↑.FORM > POWER THEN ERROR(116);                   COMP     3985
000325                 INSYMBOL                                                 COMP     3986
000325                END;                                                      COMP     3987
000326               IF ( ID = =OCT       = ) THEN                             HEXOUT     2
000330                DEFLJMP( 20,=WRO       = )                               HEXOUT     3
000332               ELSE                                                      HEXOUT     4
000334                DEFLJMP( 15,=WRH       = ) ;                             HEXOUT     5
000337               IF NOT (SY IN [COMMA,RPARENT]) THEN                        COMP     3989
000342                BEGIN ERROR(6); SKIP(FSYS+[COMMA,RPARENT]) END            COMP     3990
000346              END (*SY = IDENT*)                                          COMP     3991
000346             ELSE                                                         COMP     3992
000347              IF COMPTYPES(LSP,INTPTR,FALSE) THEN                         COMP     3993
000354               DEFLJMP(10,=WRI       =)                                   COMP     3994
000355              ELSE                                                        COMP     3995
000357               IF COMPTYPES(LSP,REALPTR,FALSE) THEN                       COMP     3996
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A.

```
000366                DEFLJMP(22,ΞWRE       Ξ)                                  COMP     3997
000367               ELSE                                                       COMP     3998
000372                IF COMPTYPES(LSP,CHARPTR,FALSE) THEN                      COMP     3999
000401                 DEFLJMP(1 ,ΞWRC       Ξ)                                COMP     4000
000402                ELSE                                                      COMP     4001
000404                 IF COMPTYPES(LSP,BOOLPTR,FALSE) THEN                     COMP     4002
000413                  DEFLJMP(10,ΞWRB       Ξ)                               COMP     4003
000414                 ELSE                                                     COMP     4004
000417                  IF LSP <> NIL THEN                                      COMP     4005
000421                   IF STRING(LSP) THEN                                    COMP     4006
000426                    WITH LSP↑ DO                                          COMP     4007
000430                     BEGIN LSZ := ALFALENG*SIZE.WORDS+                    COMP     4008
000432                       SIZE.BITS DIV 6;                                   COMP     4009
000437                       (*PASS STRING LENGTH:*)                            COMP     4010
000437                       NEEDX(6,7,I);                                      COMP     4011
000442                       GEN30(71B,I,0,LSZ,0);                              COMP     4012
000447                       PARAM.CWDISPL := 6;                                COMP     4013
000450                       STORE(PARAM,I);                                    COMP     4014
000453                       DEFLJMP(LSZ,ΞWRS       Ξ)                          COMP     4015
000456                     END                                                  COMP     4016
000460                    ELSE ERROR(116);                                      COMP     4017
000463             END                                                         UPDATE7   139
000463            ELSE                                                         UPDATE7   140
000464             BEGIN LOADPPTR( FILATTR,LDRCT ) ;                           UPDATE7   141
000467              WITH LATTR DO                                              UPDATE7   142
000467               BEGIN TYPTR := FILATTR.TYPTR↑.FILTYPE ; KIND := VARBL ;   UPDATE7   143
000472                WORDACC := INDRCT ;  CWDISPL := 0 ;  VWDISPL := 1 ;      UPDATE7   144
000474                 PCKD := FALSE ;                                         UPDATE7   145
000474                END ;                                                    UPDATE7   146
000474               ASSIGNTO( LATTR ) ;                                       UPDATE7   147
000477               NEEDX(1,1,I); (*RESET REFERENCE TO X1*)                   UPDATE7   148
000502               IF FILATTR.TYPTR↑.TEXTFILE THEN                           UPDATE7   149
000504                SEARCHEXTID( ΞP.PUTC    Ξ,EXT )                          UPDATE7   150
000506               ELSE BEGIN GEN15( 548,1,1,1 ) ;                           UPDATE7   151
000515                    SEARCHEXTID( ΞP.PUTB    Ξ,EXT ) ;                    UPDATE7   152
000521                   END ;                                                  UPDATE7   153
000521               RJTOEXT ;  CLEARREGS ;                                    UPDATE7   154
000525              END ;                                                      UPDATE7   155
000525             B6DPL := 3 ;                                                UPDATE5    23
000526             EXITLOOP := SY <> COMMA;                                    COMP     4019
000531             IF NOT EXITLOOP THEN                                        COMP     4020
000532              BEGIN INSYMBOL ;  NEEDX( 1,1,I ) ; (*RESERVE A1/X1*)       UPDATE7   156
000536               EXPRESSION(FSYS+[COMMA,COLON,RPARENT,IDENT])               COMP     4022
000540              END;                                                        COMP     4023
000542            UNTIL EXITLOOP;                                              COMP     4024
000543           IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                   COMP     4025
000550          END (*SY = LPARENT*)                                           COMP     4026
000551         ELSE IF LKEY = 10 THEN (*WRITE*) ERROR(116);                    COMP     4027
000556         IF LKEY = 11 THEN (*WRITELN*)                                   COMP     4028
000561          BEGIN LOADPPTR( FILATTR,LDRCT ) ;                              UPDATE7   157
000563           SEARCHEXTID( ΞP.PUTLN   Ξ,EXT ) ;  RJTOEXT ; CLEARREGS ;      UPDATE7   158
000573          END;                                                           COMP     4031
000573         IF NOT LDRCT THEN LC := LC - 1                                  COMP     4032
000575        END (*WRITE*) ;                                                  COMP     4033
000620                                                                         COMP     4034
000003        PROCEDURE MESSAGE;                                               COMP     4035
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000003        VAR I: REGNR;                                                    COMP  4036
000004       BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                COMP  4037
000013        EXPRESSION(FSYS+[RPARENT]);                                      COMP  4038
000017        LOADADDRESS(GATTR,I); BXIXJ(1,I); CLEARREGS;                     COMP  4039
000030        IF GATTR.TYPTR <> NIL THEN                                       COMP  4040
000034         IF STRING(GATTR.TYPTR) THEN                                     COMP  4041
000040          GEN30(71B,2,0,GATTR.TYPTR↑.SIZE.WORDS*ALFALENG                 COMP  4042
000043             +GATTR.TYPTR↑.SIZE.BITS DIV 6,0)                            COMP  4043
000051         ELSE ERROR(116);                                                COMP  4044
000056        SEARCHEXTID(ΞP.MSG     Ξ,EXT); RJTOEXT;                          COMP  4045
000054        IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                      COMP  4046
000071       END (*MESSAGE*) ;                                                 COMP  4047
000077                                                                         COMP  4048
000003       PROCEDURE PAGE;                                                   COMP  4049
000003        VAR I: REGNR;                                                    COMP  4050
000004       BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                COMP  4051
000013        CLEARREGS; XRGS[1].XCONT := OTHER; (*RESERVES A1/X1*)            COMP  4052
000022        VARIABLE(FSYS+[RPARENT]);                                        COMP  4053
000025        CLEARREGS; (*TO PREVENT EX1... AND GUARANTEE SA1...*)            COMP  4054
000027        IF GATTR.TYPTR <> NIL THEN                                       COMP  4055
000033         BEGIN                                                           COMP  4056
000033          IF GATTR.WORDACC <> DRCT THEN                                  COMP  4057
000034           XRGS[GATTR.VWDISPL].XCONT := OTHER;                           COMP  4058
000041          WITH GATTR.TYPTR↑ DO                                           COMP  4059
000044           IF FORM = FILES THEN                                          COMP  4060
000047            BEGIN IF NOT TEXTFILE THEN ERROR(116);                       COMP  4061
000052             GATTR.CWDISPL := GATTR.CWDISPL + CHEFET - 1                 COMP  4062
000055            END                                                          COMP  4063
000056           ELSE ERROR(116)                                               COMP  4064
000057         END;                                                            COMP  4065
000060        LOAD(GATTR,I); (*I = 1 IS GUARANTEED*) CLEARREGS;                COMP  4066
000066        (*BECAUSE PUTLN AND PUTC GUARANTEE TO LEAVE A1 UNCHANGED, IT NEED COMP  4067
000066        NOT BE SAVED*)                                                   COMP  4068
000066        SEARCHEXTID(ΞP.PUTLN    Ξ,EXT); RJTOEXT;                         COMP  4069
000074        GEN30(71B,6,0,ORD(Ξ1Ξ),0); GEN15(54B,1,1,0); GEN15(53B,6,1,0);   COMP  4070
000110        SEARCHEXTID(ΞP.PUTC     Ξ,EXT); RJTOEXT;                         COMP  4071
000116        GEN15(54B,1,1,0);                                                COMP  4072
000122        SEARCHEXTID(ΞP.PUTLN    Ξ,EXT); RJTOEXT;                         COMP  4073
000130        IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                      COMP  4074
000135       END (*PAGE*) ;                                                    COMP  4075
000144                                                                         COMP  4076
000003       PROCEDURE TIMEDATE;                                               COMP  4077
000003        VAR I: REGNR;                                                    COMP  4078
000004       BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                COMP  4079
000013        VARIABLE(FSYS+[RPARENT]);                                        COMP  4080
000016        IF NOT COMPTYPES(GATTR.TYPTR,ALFAPTR,FALSE) THEN ERROR(116);     COMP  4081
000026        LOADADDRESS(GATTR,I); BXIXJ(1,I); CLEARREGS;                     COMP  4082
000037        IF LKEY = 14 THEN (*TIME*) SEARCHEXTID(ΞP.TIME     Ξ,EXT)        COMP  4083
000043        ELSE (*DATE*) SEARCHEXTID(ΞP.DATE     Ξ,EXT);                    COMP  4084
000052        RJTOEXT;                                                         COMP  4085
000054        IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                      COMP  4086
000061       END (*TIMEDATE*) ;                                                COMP  4087
000070                                                                         COMP  4088
000003       PROCEDURE HALT;                                                   COMP  4089
000003       BEGIN CLEARREGS;                                                  COMP  4090
000010        GEN30(51B,0,0,IC,0); GEN30(04B,0,0,HALTERR,0)                    COMP  4091
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000016      END (*HALT*) ;                                                    COMP    4092
000024                                                                        COMP    4093
000003     PROCEDURE PACK;                                                    COMP    4094
000003      VAR BITS: BITRANGE; FW,LADDR: ADDRRANGE;                          COMP    4095
000006        I,J,K,R,S,T: REGNR; LSP,LSP1: STP; LATTR: ATTR;                 COMP    4096
000030        PW,EPW,LOW,HIGH,LMIN,LMAX: INTEGER; LEFTADJ: BOOLEAN;           COMP    4097
000037                                                                        COMP    4098
000003      PROCEDURE PACKWORD(NROFELS: BITRANGE);                            COMP    4099
000004       VAR LADDR: ADDRRANGE; SHFT: BITRANGE;                            COMP    4100
000006      BEGIN NEEDB(S); GEN30(61B,S,R,NROFELS,0);                         COMP    4101
000017       GEN15(13B,J,J,J); NOOP; LADDR := IC;                             COMP    4102
000030       GEN15(56B,K,R,0);                                                COMP    4103
000034       IF LEFTADJ THEN                                                  COMP    4104
000036        BEGIN GEN15(11B,K,I,K); GEN15(12B,J,J,K);                       COMP    4105
000046         GEN15(20B,J,0,BITS)                                            COMP    4106
000051        END                                                             COMP    4107
000053       ELSE                                                             COMP    4108
000054        BEGIN GEN15(15B,K,K,I); GEN15(20B,J,0,BITS);                    COMP    4109
000065         GEN15(12B,J,J,K)                                               COMP    4110
000067        END;                                                            COMP    4111
000071       GEN15(66B,R,R,1); GEN30(05B,R,S,LADDR,2);                        COMP    4112
000102       BRGS[S].BCONT := FREE;                                           COMP    4113
000110       SHFT := WORDSIZE - NROFELS*BITS;                                 COMP    4114
000114       IF SHFT > 0 THEN GEN15(20B,J,0,SHFT)                             COMP    4115
000117      END (*PACKWORD*) ;                                                COMP    4116
000124                                                                        COMP    4117
000124     BEGIN (*PACK*)                                                     COMP    4118
000124     IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                       COMP    4119
000013     VARIABLE(FSYS+[COMMA,RPARENT]); LATTR := GATTR;                    COMP    4120
000023     LSP := NIL; LSP1 := NIL; LOW := 0; HIGH := 0; LEFTADJ := FALSE;    COMP    4121
000027     IF GATTR.TYPTR <> NIL THEN                                         COMP    4122
000032      WITH GATTR.TYPTR↑ DO                                              COMP    4123
000033       IF FORM = ARRAYS THEN                                            COMP    4124
000035        IF  NOT PCKDARR THEN                                            COMP    4125
000037         BEGIN LSP := INXTYPE; LSP1 := AELTYPE;                         COMP    4126
000041          IF LSP <> NIL THEN GETBOUNDS(LSP,LOW,HIGH);                   COMP    4127
000045          IF LSP1 <> NIL THEN LEFTADJ := LSP1↑.FORM IN [ARRAYS,RECORDS] COMP    4128
000047         END                                                            COMP    4129
000053        ELSE ERROR(116)                                                 COMP    4130
000055       ELSE ERROR(116);                                                 COMP    4131
000061     IF SY = COMMA THEN INSYMBOL ELSE ERROR(20);                        COMP    4132
000067     EXPRESSION(FSYS+[COMMA,RPARENT]);                                  COMP    4133
000073     IF NOT COMPTYPES(GATTR.TYPTR,LSP,FALSE) THEN ERROR(116);           COMP    4134
000103     (*LOAD SOURCE ADDRESS (LATTR[GATTR]) INTO B-R:*)                    COMP    4135
000103     NEEDB(R);                                                          COMP    4136
000106     IF (LATTR.TYPTR <> NIL)AND (GATTR.TYPTR <> NIL) THEN               COMP    4137
000114      BEGIN LATTR.CWDISPL := LATTR.CWDISPL - LOW;                       COMP    4138
000116       IF GATTR.KIND = CST THEN                                         COMP    4139
000117        BEGIN LATTR.CWDISPL := LATTR.CWDISPL + GATTR.CVAL.IVAL;         COMP    4140
000120         LOADADDRESS(LATTR,I)                                           COMP    4141
000121        END                                                             COMP    4142
000122       ELSE                                                             COMP    4143
000123        BEGIN LOADADDRESS(LATTR,I); LOAD(GATTR,J);                      COMP    4144
000131         OPERATION( 36B,K,I,J ) ;  I := K                               UPDATE9   40
000135        END;                                                            COMP    4146
000137       GEN15(63B,R,I,0); DECREFX(I)                                     COMP    4147
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000144      END;                                                                COMP     4148
000146     IF SY = COMMA THEN INSYMBOL ELSE ERROR(20);                          COMP     4149
000154    NEEDX(6,7,J); ARGS[J].ACONT := UNSPECADDR ;                           UPDATE21   19
000156     VARIABLE(FSYS+[RPARENT]);                                            COMP     4150
000171     FW := 1; PW := 0; EPW := 60; BITS := 1;                              COMP     4151
000174     IF GATTR.TYPTR <> NIL THEN                                           COMP     4152
000200      WITH GATTR.TYPTR↑ DO                                                COMP     4153
000201       IF FORM = ARRAYS THEN                                              COMP     4154
000203        IF PCKDARR AND PARTWORDELS AND                                    COMP     4155
000205         COMPTYPES(AELTYPE,LSP1,FALSE) AND COMPTYPES(INXTYPE,             COMP     4156
000214         LSP,FALSE) THEN                                                  COMP     4157
000222         BEGIN LMIN := 0; LMAX := 0;                                      COMP     4158
000223          IF INXTYPE <> NIL THEN GETBOUNDS(INXTYPE,LMIN,LMAX);            COMP     4159
000230          IF LMAX - LMIN > HIGH - LOW THEN ERROR(116);                     COMP     4160
000234          IF AELTYPE <> NIL THEN BITS := AELTYPE↑.SIZE.BITS;               COMP     4161
000241          EPW := ELSPERWORD;                                               COMP     4162
000243          FW := (LMAX-LMIN+1) DIV EPW;                                     COMP     4163
000247          PW := (LMAX-LMIN+1) - FW*EPW                                     COMP     4164
000250         END                                                               COMP     4165
000251        ELSE ERROR(116)                                                    COMP     4166
000253       ELSE ERROR(116);                                                    COMP     4167
000257    NEEDX(1,5,K); ARGS[K].ACONT := UNSPECADDR ;                           UPDATE21   20
000271     NEEDX(0,7,I);                                                        COMP     4168
000274     IF LEFTADJ THEN GEN15(43B,I,0,BITS)                                  COMP     4169
000277     ELSE GEN15(43B,I,0,WORDSIZE-BITS);                                   COMP     4170
000307     IF FW > 0 THEN                                                       COMP     4173
000311      BEGIN                                                               COMP     4174
000311       IF FW <> 1 THEN                                                    COMP     4175
000312        BEGIN (*SAVE B4/B5:*)                                             COMP     4176
000312         NEEDX(0,7,T); GEN15(76B,T,4,0);                                  COMP     4177
000321         GEN15(20B,T,0,18); GEN15(76B,J,6,0);                             COMP     4178
000331         GEN15(12B,T,T,J); LOADADDRESS(GATTR,S); DECREFX(S);              COMP     4179
000344         GEN15(63B,4,S,0); GEN30(61B,6,4,FW,0); NOOP; LADDR := IC         COMP     4180
000357        END;                                                              COMP     4181
000361       PACKWORD(EPW);                                                     COMP     4182
000363       IF FW = 1 THEN STORE(GATTR,J)                                      COMP     4183
000370       ELSE                                                               COMP     4184
000372        BEGIN GEN15(56B,J,4,0); DECREFX(J); GEN15(66B,4,4,1);             COMP     4185
000405         GEN30(05B,4,6,LADDR,2); GEN15(63B,6,T,0); GEN15(20B,T,0,42);     COMP     4186
000422         GEN15(63B,4,T,0); DECREFX(T)                                     COMP     4187
000427        END                                                               COMP     4188
000431      END (*FW > 0*) ;                                                    COMP     4189
000431     IF PW > 0 THEN                                                       COMP     4190
000433      BEGIN PACKWORD(PW);                                                 COMP     4191
000434       IF FW > 0 THEN GEN15(54B,J,J,1) ELSE STORE(GATTR,J);               UPDATE4     3
000447      END;                                                                COMP     4193
000447     DECREFX(I); DECREFX(K); BRGS[R].BCONT := FREE;                       COMP     4194
000463     IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                          COMP     4195
000470    END (*PACK*) ;                                                        COMP     4196
000475                                                                          COMP     4197
000003    PROCEDURE UNPACK;                                                     COMP     4198
000003     VAR BITS: BITRANGE; FW,LADDR: ADDRRANGE;                             COMP     4199
000006       I,J,K,R,T: REGNR; LSP,LSP1: STP; SOURCE,DEST: ATTR;                COMP     4200
000041       EPW,PW,LOW,HIGH,LMIN,LMAX: INTEGER; LMODE: (USRADJ,SRADJ,USLADJ);  COMP     4201
000050                                                                          COMP     4202
000003     PROCEDURE UNPACKWORD(NROFELS: BITRANGE);                             COMP     4203
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A.

```
000004        VAR LADDR: ADDRRANGE; S: REGNR;                                  COMP     4204
000006       BEGIN NEEDB(S); GEN30(61B,S,R,NROFELS,0);                         COMP     4205
000017        NOOP; LADDR := IC;                                               COMP     4206
000024        XRGS[ K ].XCONT := OTHER ;                                       UPDATE9    41
000033        CASE LMODE OF                                                    COMP     4207
000035         USRADJ:                                                         COMP     4208
000035          BEGIN GEN15(20B,K,0,BITS); GEN15(15B,J,K,I) END;               COMP     4209
000046         SRADJ:                                                          COMP     4210
000046          BEGIN GEN15(10B,J,K,0); GEN15(20B,K,0,BITS);                   COMP     4211
000056           GEN15(21B,J,0,WORDSIZE-BITS)                                  COMP     4212
000062          END;                                                           COMP     4213
000065         USLADJ:                                                         COMP     4214
000065          BEGIN GEN15(11B,J,I,K); GEN15(20B,K,0,BITS) END                COMP     4215
000075        END;                                                             COMP     4216
000101        GEN15(56B,J,R,0); GEN15(66B,R,R,1);                              COMP     4217
000112        GEN30(05B,R,S,LADDR,2); BRGS[S].BCONT := FREE                    COMP     4218
000120       END (*UNPACKWORD*);                                               COMP     4219
000131                                                                         COMP     4220
000131       BEGIN (*UNPACK*)                                                  COMP     4221
000131        IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                     COMP     4222
000013        EXPRESSION(FSYS+[COMMA,RPARENT]); SOURCE := GATTR;               COMP     4223
000024        LSP := NIL; LSP1 := NIL; LMIN := 0; LMAX := 0;                   COMP     4224
000027        FW := 1; PW := 0; EPW := 60; BITS := 1; LMODE := USRADJ;          COMP     4225
000033        IF GATTR.TYPTR <> NIL THEN                                       COMP     4226
000036         WITH GATTR.TYPTR↑ DO                                            COMP     4227
000037          IF FORM = ARRAYS THEN                                          COMP     4228
000041           IF PCKDARR AND PARTWORDELS THEN                               COMP     4229
000043            BEGIN LSP := INXTYPE; LSP1 := AELTYPE;                       COMP     4230
000046             IF LSP <> NIL THEN GETBOUNDS(LSP,LMIN,LMAX);                COMP     4231
000051             IF LSP1 <> NIL THEN                                         COMP     4232
000053              WITH LSP1↑ DO                                              COMP     4233
000054               BEGIN BITS := SIZE.BITS;                                  COMP     4234
000055                IF FORM = SUBRANGE THEN                                  COMP     4235
000057                 BEGIN IF MIN.IVAL < 0 THEN LMODE := SRADJ                COMP     4236
000061                 END                                                     COMP     4237
000062                ELSE                                                     COMP     4238
000063                 IF FORM IN [ARRAYS,RECORDS] THEN LMODE := USLADJ        COMP     4239
000066               END;                                                      COMP     4240
000067             EPW := ELSPERWORD;                                          COMP     4241
000071             FW := (LMAX - LMIN + 1) DIV EPW;                            COMP     4242
000075             PW := (LMAX - LMIN + 1) - FW*EPW                            COMP     4243
000076            END                                                          COMP     4244
000077           ELSE ERROR(116)                                               COMP     4245
000101          ELSE ERROR(116);                                               COMP     4246
000105        IF SY = COMMA THEN INSYMBOL ELSE ERROR(20);                      COMP     4247
000113        VARIABLE(FSYS+[COMMA,RPARENT]); DEST := GATTR;                   COMP     4248
000123        IF GATTR.TYPTR <> NIL THEN                                       COMP     4249
000126         WITH GATTR.TYPTR↑ DO                                            COMP     4250
000127          IF FORM = ARRAYS THEN                                          COMP     4251
000131           IF NOT PCKDARR AND COMPTYPES(AELTYPE,LSP1,FALSE)              COMP     4252
000135            AND COMPTYPES(INXTYPE,LSP,FALSE) THEN                        COMP     4253
000147            BEGIN LOW := 0; HIGH := 0;                                   COMP     4254
000150             IF INXTYPE <> NIL THEN GETBOUNDS(INXTYPE,LOW,HIGH);         COMP     4255
000155             IF LMAX - LMIN > HIGH - LOW THEN ERROR(116);                COMP     4256
000161            END                                                          COMP     4257
000161           ELSE ERROR(116)                                               COMP     4258
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000163        ELSE ERROR(116);                                                    COMP     4259
000167       IF SY = COMMA THEN INSYMBOL ELSE ERROR(20);                          COMP     4260
000175       EXPRESSION(FSYS+[RPARENT]);                                          COMP     4261
000201       IF NOT COMPTYPES(GATTR.TYPTR,LSP,FALSE) THEN ERROR(116);             COMP     4262
000211       (*LOAD DESTINATION ADDRESS (DEST[GATTR]) INTO B-R:*)                  COMP     4263
000211       NEEDB(R);                                                            COMP     4264
000214       IF (DEST.TYPTR <> NIL)AND (GATTR.TYPTR <> NIL) THEN                  COMP     4265
000222        BEGIN DEST.CWDISPL := DEST.CWDISPL - LOW;                           COMP     4266
000224         IF GATTR.KIND = CST THEN                                           COMP     4267
000225          BEGIN DEST.CWDISPL := DEST.CWDISPL + GATTR.CVAL.IVAL;             COMP     4268
000226           LOADADDRESS(DEST,I)                                              COMP     4269
000227          END                                                               COMP     4270
000230         ELSE                                                               COMP     4271
000231          BEGIN LOADADDRESS(DEST,I); LOAD(GATTR,J);                         COMP     4272
000237          OPERATION( 36B,K,I,J ) ; I := K                                   UPDATE9    42
000243         END;                                                               COMP     4274
000245        GEN15(63B,R,I,0); DECREFX(I)                                        COMP     4275
000252        END;                                                                COMP     4276
000254      NEEDX(6,7,J); ARGS[J].ACONT := UNSPECADDR; NEEDX(0,7,I);              UPDATE21   21
000271       IF LMODE = USRADJ THEN GEN15(43B,I,0,WORDSIZE-BITS)                  COMP     4278
000276       ELSE                                                                 COMP     4279
000301        IF LMODE = USLADJ THEN GEN15(43B,I,0,BITS);                         COMP     4280
000307       IF FW > 0 THEN                                                       COMP     4281
000311        BEGIN                                                               COMP     4282
000311         IF (FW > 1)OR (PW > 0) THEN                                        COMP     4283
000313          BEGIN NEEDX(0,7,T); GEN15(76B,T,4,0);                             COMP     4284
000322           LOADADDRESS(SOURCE,K); GEN15(63B,4,K,0);                         COMP     4285
000331           DECREFX(K)                                                       COMP     4286
000332          END;                                                              COMP     4287
000334         IF FW > 1 THEN                                                     COMP     4288
000336          BEGIN GEN15(20B,T,0,18); GEN15(76B,J,6,0);                        COMP     4289
000346           GEN15(12B,T,T,J); GEN30(61B,6,4,FW,0);                           COMP     4290
000357           NOOP; LADDR := IC;                                               COMP     4291
000363          END;                                                              COMP     4292
000363         IF (FW = 1)AND (PW = 0) THEN                                       COMP     4293
000370          BEGIN LOAD(SOURCE,K); ROTATEX(K) END                              COMP     4294
000376         ELSE                                                               COMP     4295
000377          BEGIN NEEDX(1,5,K); ARGS[K].ACONT := UNSPECADDR;                  COMP     4296
000410           GEN15(56B,K,4,0)                                                 COMP     4297
000412          END;                                                              COMP     4298
000413         UNPACKWORD(EPW);                                                   COMP     4299
000415         IF FW > 1 THEN                                                     COMP     4300
000417          BEGIN GEN15(66B,4,4,1); GEN30(05B,4,6,LADDR,2);                   COMP     4301
000427           GEN15(63B,6,T,0); GEN15(20B,T,0,42)                              COMP     4302
000435          END                                                               COMP     4303
000437        END;                                                                COMP     4304
000437       IF PW > 0 THEN                                                       COMP     4305
000441        BEGIN                                                               COMP     4306
000441         IF FW > 0 THEN                                                     COMP     4307
000442          BEGIN IF FW = 1 THEN GEN15(56B,K,4,1) ELSE GEN15(56B,K,4,0);      COMP     4308
000454           GEN15(63B,4,T,0); DECREFX(T)                                     COMP     4309
000461          END                                                               COMP     4310
000463         ELSE                                                               COMP     4311
000464          BEGIN LOAD(SOURCE,K); ROTATEX(K) END;                             COMP     4312
000472         UNPACKWORD(PW);                                                    COMP     4313
000474        END                                                                 COMP     4314
```

MOORE BUSINESS FORMS, INC.
FORM 1413
PRINTED IN U.S.A.
22

```
000003        VAR I: REGNR;                                                    COMP     4371
000004       BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                COMP     4372
000013        VARIABLE(FSYS+[COMMA,RPARENT]);                                  COMP     4373
000016        IF GATTR.TYPTR <> NIL THEN                                       COMP     4374
000022         IF GATTR.TYPTR↑.FORM = POINTER THEN                             COMP     4375
000024          BEGIN LOAD(GATTR,I); GEN15(63B,4,I,0); DECREFX(I) END          COMP     4376
000035         ELSE ERROR(116);                                                COMP     4377
000040        IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                      COMP     4378
000045       END (*RELEASE*) ;                                                 COMP     4379
000052                                                                         COMP     4380
000003       PROCEDURE ROUNDF;                                                 COMP     4381
000003       BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                COMP     4382
000013        EXPRESSION(FSYS+[RPARENT]);                                      COMP     4383
000017        IF NOT COMPTYPES(GATTR.TYPTR,REALPTR,FALSE) THEN ERROR(125);     COMP     4384
000027        LOAD(GATTR,I); NEEDX(0,7,J); NEEDX(0,7,K);                       COMP     4385
000044        GEN15(13B,J,J,J); GEN15(27B,J,0,J); GEN15(30B,K,I,J);            COMP     4386
000060        GEN15(32B,J,I,J); GEN15(34B,J,K,J); GEN15(26B,J,0,J);            UPDATE21   22
000074        GEN15(13B,K,K,K); GEN15(36B,J,J,K);                              UPDATE21   23
000104        DECREFX(I); DECREFX(K);                                          COMP     4388
000112        GATTR.TYPTR := INTPTR; GATTR.EXPREG := J;                        COMP     4389
000117        IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                      COMP     4390
000124       END (*ROUNDF*) ;                                                  COMP     4391
000131                                                                         COMP     4392
000003       PROCEDURE ABSF;                                                   COMP     4393
000003       BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                COMP     4394
000013        EXPRESSION(FSYS+[RPARENT]);                                      COMP     4395
000017        IF COMPTYPES(GATTR.TYPTR,INTPTR,FALSE)OR COMPTYPES(GATTR.TYPTR,  COMP     4396
000027         REALPTR,FALSE) THEN                                             COMP     4397
000034         BEGIN LOAD(GATTR,I); NEEDX(0,7,K); GEN15(10B,K,I,0);            COMP     4398
000050          GEN15(21B,K,0,59(*WORDSIZE-1*)); GEN15(13B,K,K,I); DECREFX(I); COMP     4399
000065          GATTR.EXPREG := K                                              COMP     4400
000065         END                                                             COMP     4401
000070        ELSE ERROR(125);                                                 COMP     4402
000073        IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                      COMP     4403
000100       END (*ABSF*) ;                                                    COMP     4404
000105                                                                         COMP     4405
000003       PROCEDURE SQRF;                                                   COMP     4406
000003       BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                COMP     4407
000013        EXPRESSION(FSYS+[RPARENT]);                                      COMP     4408
000017        LOAD(GATTR,I); DECREFX(I); NEEDX(0,7,K); GATTR.EXPREG := K;      COMP     4409
000036        IF COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN GEN15(42B,K,I,I)     COMP     4410
000047        ELSE                                                             COMP     4411
000051         IF COMPTYPES(GATTR.TYPTR,REALPTR,FALSE) THEN GEN15(41B,K,I,I)   COMP     4412
000064         ELSE BEGIN ERROR(125); GATTR.TYPTR := INTPTR END;               COMP     4413
000073        IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                      COMP     4414
000100       END (*SQRF*) ;                                                    COMP     4415
000105                                                                         COMP     4416
000003       PROCEDURE TRUNCF;                                                 COMP     4417
000003       BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                COMP     4418
000013        EXPRESSION(FSYS+[COMMA,RPARENT]);                                COMP     4419
000017        IF NOT COMPTYPES(GATTR.TYPTR,REALPTR,FALSE) THEN ERROR(125);     COMP     4420
000027        LOAD(GATTR,I); DECREFX(I); NEEDX(0,7,K); NEEDB(J);               COMP     4421
000046        GEN15(26B,K,J,I);                                                COMP     4422
000052        IF SY = COMMA THEN                                               COMP     4423
000054         BEGIN INSYMBOL; EXPRESSION(FSYS+[RPARENT]);                     COMP     4424
000061          IF GATTR.TYPTR <> NIL THEN                                     COMP     4425
```

MOORE BUSINESS FORMS, INC., F
FORM 213
PRINTED IN U.S.A.

```
000065          IF COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN                          COMP  4426
000071            IF GATTR.KIND = CST THEN GEN30(61B,J,J,GATTR.CVAL.IVAL,0)          COMP  4427
000077           ELSE                                                                COMP  4428
000101            BEGIN LOAD(GATTR,I);                                               COMP  4429
000104             GEN15(63B,J,I,J); DECREFX(I)                                      COMP  4430
000112            END                                                                COMP  4431
000113          ELSE ERROR(125)                                                      COMP  4432
000115         END;                                                                  COMP  4433
000116        GEN15(22B,K,J,K); BRGS[J].BCONT := FREE; NEEDX(0,7,I);                 COMP  4434
000133        GEN15(13B,I,I,I); GEN15(36B,K,K,I); DECREFX(I);                        COMP  4435
000147        WITH GATTR DO                                                          COMP  4436
000147         BEGIN TYPTR := INTPTR; KIND := EXPR; EXPREG := K END;                 COMP  4437
000155        IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                            COMP  4438
000162       END (*TRUNCF*) ;                                                        COMP  4439
000167                                                                               COMP  4440
000003       PROCEDURE UNDEFINEDP;                                                   COMP  4441
000003        VAR I,K: REGNR;                                                        COMP  4442
000005       BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                      COMP  4443
000013        EXPRESSION(FSYS+[RPARENT]);                                            COMP  4444
000017        LOAD(GATTR,I);                                                         COMP  4445
000023        IF COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN PACKANDNORM(I)             COMP  4446
000034        ELSE                                                                   COMP  4447
000037         IF NOT COMPTYPES(GATTR.TYPTR,REALPTR,FALSE) THEN ERROR(125);          COMP  4448
000047        NEEDX(0,7,K); GEN15(76B,K,1,0);                                        COMP  4449
000057        IF CP >= 3 THEN GEN30(03B,7,I,IC+2,2)                                  COMP  4450
000065        ELSE GEN30(03B,7,I,IC+1,2);                                            COMP  4451
000073        GEN30(03B,5,I,IC+1,2); DECREFX(I);                                     COMP  4452
000103        GEN15(13B,K,K,K); NOOP;                                                COMP  4453
000111        WITH GATTR DO                                                          COMP  4454
000111         BEGIN TYPTR := BOOLPTR; EXPREG := K END;                              COMP  4455
000116        IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                            COMP  4456
000123       END (*UNDEFINEDP*) ;                                                    COMP  4457
000130                                                                               COMP  4458
000003       PROCEDURE ODDP;                                                         COMP  4459
000003       BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                      COMP  4460
000013        EXPRESSION(FSYS+[RPARENT]);                                            COMP  4461
000017        IF NOT COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN ERROR(125);            COMP  4462
000027        LOAD(GATTR,I); NEEDX(0,7,K); GEN15(10B,K,I,0);                         COMP  4463
000044        GEN15(20B,K,0,59(*WORDSIZE-1*)); GEN15(13B,K,I,K); DECREFX(I);         COMP  4464
000060        WITH GATTR DO                                                          COMP  4465
000060         BEGIN TYPTR := BOOLPTR; KIND := COND; CONDCD := PL; CDR := K END;     COMP  4466
000066        IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                            COMP  4467
000073       END (*ODDP*) ;                                                          COMP  4468
000100                                                                               COMP  4469
000003       PROCEDURE ORDF;                                                         COMP  4470
000003       BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                      COMP  4471
000013        EXPRESSION(FSYS+[RPARENT]); LOAD(GATTR,I);                             COMP  4472
000024        IF GATTR.TYPTR <> NIL THEN                                             COMP  4473
000030         IF GATTR.TYPTR↑.FORM > POWER THEN ERROR(125);                         COMP  4474
000034        GATTR.TYPTR := INTPTR;                                                 COMP  4475
000037        IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                            COMP  4476
000044       END (*ORDF*) ;                                                          COMP  4477
000051                                                                               COMP  4478
000003       PROCEDURE CHRF;                                                         COMP  4479
000003       BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                      COMP  4480
000013        EXPRESSION(FSYS+[RPARENT]); LOAD(GATTR,I);                             COMP  4481
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000024      IF NOT COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN ERROR(125);          COMP    4482
000036      GATTR.TYPTR := CHARPTR;                                              COMP    4483
000041      IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                          COMP    4484
000046     END (*CHRF*) ;                                                        COMP    4485
000053                                                                           COMP    4486
000003     PROCEDURE PREDSUCCF;                                                  COMP    4487
000003     BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                    COMP    4488
000013      EXPRESSION(FSYS+[RPARENT]);                                          COMP    4489
000017      IF GATTR.TYPTR <> NIL THEN                                           COMP    4490
000023       IF (GATTR.TYPTR↑.FORM > SUBRANGE)OR COMPTYPES(GATTR.TYPTR,          COMP    4491
000025        REALPTR,FALSE) THEN ERROR(125);                                    COMP    4492
000033      LOAD(GATTR,I); NEEDX(0,7,K); GEN15(76B,K,1,0);                       COMP    4493
000050      GEN15(54B+LKEY(*37B FOR PRED, 36B FOR SUCC*),K,I,K);                 COMP    4494
000055      DECREFX(I); GATTR.EXPREG := K;                                       COMP    4495
000063      IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                          COMP    4496
000070     END (*PREDSUCCF*) ;                                                   COMP    4497
000075                                                                           COMP    4498
000003     PROCEDURE STDFLFUNCS;                                                 COMP    4499
000003     BEGIN                                                                 COMP    4500
000003      IF SY = LPARENT THEN                                                 COMP    4501
000010       BEGIN INSYMBOL; VARIABLE(FSYS+[RPARENT]);                           COMP    4502
000014        IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                        COMP    4503
000021       END                                                                 COMP    4504
000022      ELSE                                                                 COMP    4505
000023       WITH GATTR DO                                                       COMP    4506
000023        BEGIN TYPTR := TEXTPTR; KIND := VARBL; WORDACC := DRCT;            COMP    4507
000027        VLEVEL := 1; PCKD := FALSE;                                        COMP    4508
000030        IF INPUTPTR = NIL THEN                                             COMP    4509
000032         BEGIN ERROR(175); CWDISPL := 0 END                                COMP    4510
000037        ELSE CWDISPL := INPUTPTR↑.VADDR;                                   COMP    4511
000041       END;                                                                COMP    4512
000041      IF GATTR.TYPTR <> NIL THEN                                           COMP    4513
000045       WITH GATTR, TYPTR↑ DO                                               COMP    4514
000046        IF FORM = FILES THEN                                               COMP    4515
000050         BEGIN                                                             COMP    4516
000050          IF LKEY = 3 (*EOLN*) THEN                                        COMP    4517
000053           BEGIN IF NOT TEXTFILE THEN ERROR(125);                          COMP    4518
000056            CWDISPL := CWDISPL + CHEFET -1 (*P-PTR*)                       COMP    4519
000061           END                                                             COMP    4520
000062          ELSE                                                             COMP    4521
000062           IF TEXTFILE THEN CWDISPL := CWDISPL + CHEFET                    COMP    4522
000064           ELSE CWDISPL := CWDISPL + BINEFET; (*EFET*)                     COMP    4523
000070           LOAD(GATTR,I);                                                  COMP    4524
000075          IF LKEY = 2 (*EOS*) THEN                                         COMP    4525
000077           BEGIN IF NOT SEGFILE THEN ERROR(125) END                        COMP    4526
000103          ELSE                                                             COMP    4527
000104           IF LKEY = 1 (*EOF*) THEN                                        COMP    4528
000105            IF SEGFILE THEN                                                COMP    4529
000107             BEGIN NEEDX(0,7,J); GEN15(22B,J,1,I);                         COMP    4530
000117              DECREFX(I); I := J                                           COMP    4531
000122             END;                                                          COMP    4532
000124           TYPTR := BOOLPTR; KIND := COND; CONDCD := PL; CDR := I;         COMP    4533
000132         END                                                               COMP    4534
000132        ELSE ERROR(125);                                                   COMP    4535
000134     END (*STDFLFUNCS*);                                                   COMP    4536
000140                                                                           COMP    4537
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000003      PROCEDURE CARDF;                                                    COMP    4538
000003      BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                  COMP    4539
000013       EXPRESSION(FSYS+[RPARENT]);                                        COMP    4540
000017       IF GATTR.TYPTR <> NIL THEN                                         COMP    4541
000023        IF GATTR.TYPTR↑.FORM <> POWER THEN ERROR(125);                    COMP    4542
000027       LOAD(GATTR,I); DECREFX(I); NEEDX(0,7,K); GEN15(47B,K,0,I);         COMP    4543
000050       GATTR.TYPTR := INTPTR; GATTR.EXPREG := K;                          COMP    4544
000055       IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                        COMP    4545
000062      END (*CARDF*) ;                                                     COMP    4546
000067                                                                          COMP    4547
000003      PROCEDURE EXPOF;                                                    COMP    4548
000003      BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                  COMP    4549
000013       EXPRESSION(FSYS+[RPARENT]);                                        COMP    4550
000017       IF NOT COMPTYPES(GATTR.TYPTR,REALPTR,FALSE) THEN ERROR(125);       COMP    4551
000027       LOAD(GATTR,I); DECREFX(I); NEEDX(0,7,K); NEEDB(J);                 COMP    4552
000046       GEN15(26B,K,J,1); GEN30(71B,K,J,47,0); BRGS[J].BCONT := FREE;      UPDATE5   24
000065       GATTR.TYPTR := INTPTR; GATTR.EXPREG := K;                          COMP    4554
000067       IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                        COMP    4555
000074      END (*EXPOF*) ;                                                     COMP    4556
000101                                                                          COMP    4557
000003      PROCEDURE CLOCKF;                                                   COMP    4558
000003       VAR LSP: STP; LXRGS: XRGSTATJS; I: REGNR;                          COMP    4559
000025      BEGIN SAVEREFXRGS(LXRGS); CLEARREGS;                                COMP    4560
000012       SEARCHEXTID(ΞP.CLOCK   Ξ,EXT); RJTOEXT;                            COMP    4561
000020       XRGS := LXRGS;                                                     COMP    4562
000025       IF XRGS[6].XCONT <> AVAIL THEN                                     COMP    4563
000031        BEGIN NEEDX(0,7,I); GEN15(10B,I,6,0) END                          COMP    4564
000040       ELSE I := 6 ;                                                      UPDATE9   43
000042       CLEARREGS ;                                                        UPDATE9   44
000044       WITH XRGS[ I ] DO                                                  UPDATE9   45
000050        BEGIN XCONT := OTHER ;  REFNR := 1 ;  END ;                       UPDATE9   46
000054       RELOADREFXRGS(LXRGS);                                              COMP    4570
000056       WITH GATTR DO                                                      COMP    4571
000056        BEGIN TYPTR := INTPTR;                                            COMP    4572
000061         KIND := EXPR; EXPREG := I                                       COMP    4573
000062        END                                                               COMP    4574
000063      END (*CLOCKF*) ;                                                    UPDATE7   159
000070                                                                          COMP    4576
000003      PROCEDURE STDARITHFUNCS;                                            COMP    4577
000003       VAR LXRGS: XRGSTATUS; I,K: REGNR;                                  COMP    4578
000025      BEGIN IF SY = LPARENT THEN INSYMBOL ELSE ERROR(9);                  COMP    4579
000013       SAVEREFXRGS(LXRGS); CLEARREGS;                                     COMP    4580
000020       EXPRESSION(FSYS+[RPARENT]);                                        COMP    4581
000024       LOAD(GATTR,I);                                                     COMP    4582
000030       IF COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN                        COMP    4583
000040        BEGIN PACKANDNORM(I); GATTR.TYPTR := REALPTR END;                 COMP    4584
000046       IF I <> 1 THEN                                                     COMP    4585
000050        BEGIN NEEDX(1,1,K); BXIXJ(1,I) END;                               COMP    4586
000056       CLEARREGS ;                                                        UPDATE9   47
000060       IF NOT COMPTYPES(GATTR.TYPTR,REALPTR,FALSE) THEN ERROR(125);       COMP    4587
000072       IF LKEY = 19 THEN GEN15(66B,3,0,0) (*SIN*)                         COMP    4588
000076       ELSE IF LKEY = 20 THEN GEN15(66B,3,1,0); (*COS*)                   COMP    4589
000106       (*SET NEG. REL. ADDRESS TO INDICATE A CALL OF A STD ARITH. FUNC:*)  COMP    4590
000106       GEN30(71B,6,0,-IC,0); GEN15(56B,6,2,1);                           COMP    4591
000117       GEN30(71B,7,0,IC+1,2);                                             COMP    4592
000124       CASE LKEY OF                                                       COMP    4593
```

MOORE BUSINESS FORMS, INC. F
FORM 1413
PRINTED IN U.S.A
22

```
000126          (*SIN*) 19,                                                    COMP     4594
000126          (*COS*) 20: SEARCHEXTID(ΞP.SINCO   Ξ,EXT);                     COMP     4595
000133          (*EXP*) 21: SEARCHEXTID(ΞP.EXP     Ξ,EXT);                     COMP     4596
000140          (*SQRT*)22: SEARCHEXTID(ΞP.SQRT    Ξ,EXT);                     COMP     4597
000145          (*LN*)  23: SEARCHEXTID(ΞP.LN      Ξ,EXT);                     COMP     4598
000152          (*ARCTAN*) 24: SEARCHEXTID(ΞP.ATAN    Ξ,EXT)                   COMP     4599
000154        END;                                                             COMP     4600
000165        RJTOEXT;                                                         COMP     4601
000167        (*CLEAR INDICATION WORD:*)                                       COMP     4602
000167        GEN15(13B,7,7,7); GEN15(56B,7,2,1);                              COMP     4603
000177        XRGS := LXRGS ;                                                  UPDATE9    48
000204        IF XRGS[6].XCONT <> AVAIL THEN                                   COMP     4605
000210         BEGIN NEEDX(0,7,I); GEN15(10B,I,6,0) END                        COMP     4606
000217        ELSE I := 6 ;                                                    UPDATE9    49
000221        CLEARREGS ;                                                      UPDATE9    50
000223        WITH XRGS[ I ] DO                                                UPDATE9    51
000227         BEGIN XCONT := OTHER ;  REFNR := 1 ;  END ;                      UPDATE9    52
000233        RELOADREFXRGS(LXRGS);                                            COMP     4612
000235        WITH GATTR DO                                                    COMP     4613
000235         BEGIN TYPTR := REALPTR; KIND := EXPR; EXPREG := I END;          COMP     4614
000242        IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                      COMP     4615
000247      END (*STDARITHFUNCS*) ;                                            COMP     4616
000261                                                                         COMP     4617
000003      PROCEDURE CALLNONSTANDARD;                                         COMP     4618
000003       VAR NXT,LCP: CTP; LSP: STP; LKIND: IDKIND; LB,FTN: BOOLEAN;       COMP     4619
000011         L,M: LEVRANGE; I,K,LXPAR: REGNR; PARAM: ATTR;                   COMP     4620
000030         PVDISP,LDSP,LB6DPL,NPARMS: ADDRRANGE;  LXRGS: XRGSTATUS;        RUNMODS    10
000054         LMIN,LMAX: INTEGER;                                             COMP     4622
000056         PASS: (VAL,VARADDR,PROCDESC);                                   COMP     4623
000057      BEGIN                                                              COMP     4624
000057       WITH FCP↑ DO                                                      COMP     4625
000010        BEGIN NXT := NEXT; LKIND := PFKIND;                              COMP     4626
000013         FTN := FALSE;                                                   COMP     4627
000014         IF LKIND = ACTUAL THEN FTN := PFDECL = FTNDECL;                 COMP     4628
000020         IF KLASS = FUNC THEN                                            COMP     4629
000024          BEGIN SAVEREFXRGS(LXRGS);                                      COMP     4630
000026           IF B6DPL <> 3 THEN                                            COMP     4631
000030            BEGIN                                                        COMP     4632
000030             FOR I := 0 TO 7 DO                                          COMP     4633
000031              BEGIN                                                      COMP     4634
000033               WITH ARGS[I] DO                                           COMP     4635
000036                IF ACONT = SIMPADDR THEN                                 COMP     4636
000040                 IF ALEV = 0 THEN ACONT := UNSPECADDR;                   COMP     4637
000043               WITH XRGS[I] DO                                           COMP     4638
000047                IF XCONT = SIMPVAR THEN                                  COMP     4639
000052                 IF XLEV = 0 THEN XCONT := AVAIL                         COMP     4640
000054              END;                                                       COMP     4641
000060             GEN30(61B,6,6,B6DPL,0);                                     COMP     4642
000064             GEN30(51B,0,0,IC,0); GEN30(06B,6,4,53B,0)                   COMP     4643
000073            END;                                                         COMP     4644
000075           LB6DPL := B6DPL                                               COMP     4645
000075          END                                                            COMP     4646
000077        END;                                                             COMP     4647
000077       LXPAR := 0; LDSP := 3; PVDISP := LC;                              COMP     4648
000102       IF SY = LPARENT THEN                                              COMP     4649
000104        BEGIN                                                            COMP     4650
```

MOORE BUSINESS FORMS, INC., FORM 3113 PRINTED IN U.S.A. 22

```
000104          REPEAT PASS := VAL;                                                    COMP     4651
000105           IF LKIND = ACTUAL THEN                                                COMP     4652
000107            IF NXT = NIL THEN ERROR(126)                                         COMP     4653
000111            ELSE                                                                 COMP     4654
000113             IF NXT↑.KLASS IN [PROC,FUNC] THEN PASS := PROCDESC;                 COMP     4655
000117           (*NOTE THAT IN THIS IMPLEMENTATION FORMAL PROC/FUNC MUST              COMP     4656
000117           ONLY HAVE VALUE PARAMETERS*)                                          COMP     4657
000117           INSYMBOL;                                                             COMP     4658
000120           IF PASS = PROCDESC THEN                                               COMP     4659
000122            BEGIN                                                                COMP     4660
000122             IF SY <> IDENT THEN                                                 COMP     4661
000123              BEGIN ERROR(2); SKIP(FSYS+[COMMA,RPARENT]); NEEDX(0,7,I) END       UPDATE7   160
000133             ELSE                                                               COMP     4663
000134              BEGIN                                                             COMP     4664
000134               IF NXT↑.KLASS = PROC THEN SEARCHID([PROC],LCP)                   COMP     4665
000141               ELSE                                                             COMP     4666
000143                BEGIN SEARCHID([FUNC],LCP);                                     COMP     4667
000145                 IF NOT COMPTYPES(LCP↑.IDTYPE,NXT↑.IDTYPE,FALSE)                COMP     4668
000151                  THEN ERROR(128)                                               COMP     4669
000156                END;                                                            COMP     4670
000157               IF LCP↑.PFDECKIND = STANDARD THEN                                UPDATE7   161
000162                BEGIN ERROR(164) ; NEEDX( 0,7,I ) ; END                         UPDATE7   162
000170               ELSE                                                             COMP     4672
000171                IF LCP↑.PFKIND = ACTUAL THEN                                    COMP     4673
000173                 BEGIN (*SET UP DESCRIPTOR:*)                                   COMP     4674
000173                  WITH LCP↑ DO                                                  COMP     4675
000174                   BEGIN                                                        COMP     4676
000174                    IF FTN AND (PFDECL<>FTNDECL) THEN ERROR(173)                COMP     4677
000200                    ELSE                                                        COMP     4678
000202                     IF NOT FTN AND (PFDECL=FTNDECL) THEN ERROR(174);           COMP     4679
000207                    NEEDX(6,7,I);                                               COMP     4680
000213                    IF PFCNT = 0 THEN SEARCHEXTID(NAME,EXT)                     COMP     4681
000217                    ELSE                                                        COMP     4682
000222                     BEGIN PFNAME(PFCNT);                                       COMP     4683
000225                      SEARCHEXTID(PNAME,EXT)                                    COMP     4684
000227                     END;                                                       COMP     4685
000231                    GEN30(71B,I,0,0,0);                                         COMP     4686
000235                    IF (PFLEV <> 1)AND NOT FTN THEN (*ADD SURR. BASE ADR*)      COMP     4687
000242                     BEGIN                                                      COMP     4688
000242                      IF PFLEV IN LEVELS THEN                                   COMP     4689
000246                       BEGIN NEEDX(0,7,K);                                      COMP     4690
000251                        GEN15(76B,K,BRG[PFLEV],0)                               COMP     4691
000257                       END                                                      COMP     4692
000260                      ELSE LOADBASE(PFLEV,TRUE,K);                              COMP     4693
000264                      GEN15(20B,K,0,18); GEN15(12B,I,I,K);                      COMP     4694
000274                      DECREFX(K)                                                COMP     4695
000275                     END                                                        COMP     4696
000277                  END;                                                          COMP     4697
000277           (*MAKE SURE THAT PROC/FUNC HAS ONLY VALUE                            COMP     4698
000277           PARAMS OF FIXED SIZE:*)                                              COMP     4699
000277           LCP := LCP↑.NEXT;                                                    COMP     4700
000301           WHILE LCP <> NIL DO                                                  COMP     4701
000303            WITH LCP↑ DO                                                        COMP     4702
000304             BEGIN IF KLASS <> VARS THEN ERROR(170)                             COMP     4703
000307              ELSE                                                              COMP     4704
000311               IF VKIND = INDRCT THEN ERROR(170);                               COMP     4705
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000315                    LCP := NEXT                                          COMP   4706
000315                   END                                                   COMP   4707
000320                 END (*LCP↑.PFKIND = ACTUAL*)                            COMP   4708
000320                ELSE                                                     COMP   4709
000321                 BEGIN (*LOAD DESCRITOR:*)                               COMP   4710
000321                  WITH LCP↑, PARAM DO                                    COMP   4711
000322                   BEGIN TYPTR := INTPTR; KIND := VARBL;                 COMP   4712
000324                    WORDACC := DRCT; VLEVEL := PFLEV;                    COMP   4713
000326                    CWDISPL := PFADDR; PCKD := FALSE                     COMP   4714
000327                   END;                                                  COMP   4715
000330                  LOAD(PARAM,I)                                          COMP   4716
000331                 END;                                                    COMP   4717
000333               END (*SY = IDENT*);                                       COMP   4718
000333              GATTR.TYPTR := INTPTR;                                     COMP   4719
000336              INSYMBOL;                                                  COMP   4720
000337              IF NOT (SY IN FSYS+[COMMA,RPARENT]) THEN                   COMP   4721
000343               BEGIN ERROR(6); SKIP(FSYS+[COMMA,RPARENT]) END            COMP   4722
000350             END (*PASS = PROCDESC*)                                     COMP   4723
000350            ELSE                                                         COMP   4724
000351             BEGIN B6DPL := LDSP;                                        COMP   4725
000353              EXPRESSION(FSYS+[COMMA,RPARENT]); LSP := NIL;              COMP   4726
000357              IF GATTR.TYPTR <> NIL THEN                                 COMP   4727
000363               BEGIN                                                     COMP   4728
000363                IF LKIND = ACTUAL THEN                                   COMP   4729
000364                 BEGIN                                                   COMP   4730
000364                  IF NXT <> NIL THEN                                     COMP   4731
000365                   BEGIN LSP := NXT↑.IDTYPE;                             COMP   4732
000366                    IF LSP <> NIL THEN                                   COMP   4733
000367                     BEGIN                                               COMP   4734
000367                       IF (NXT↑.VKIND<>DRCT)OR (FULLWORDS                COMP   4735
000371                        (LSP↑.SIZE)<>1) THEN                             COMP   4736
000400                        PASS := VARADDR                                  COMP   4737
000400                     END                                                 COMP   4738
000401                   END                                                   COMP   4739
000401                END                                                      COMP   4740
000401               ELSE                                                      COMP   4741
000402                IF FULLWORDS(GATTR.TYPTR↑.SIZE) <> 1 THEN                COMP   4742
000406                 PASS := VARADDR;                                        COMP   4743
000407               IF PASS = VAL THEN                                        COMP   4744
000411                BEGIN LOAD(GATTR,I); ROTATEX(I);                         COMP   4745
000420                 IF (LKIND = ACTUAL)AND (NXT <> NIL) THEN                COMP   4746
000425                  IF COMPTYPES(LSP,REALPTR,FALSE) THEN                   COMP   4747
000432                   BEGIN                                                 COMP   4748
000432                    IF COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN          COMP   4749
000440                     BEGIN PACKANDNORM(I);                               COMP   4750
000442                       GATTR.TYPTR := REALPTR                            COMP   4751
000442                     END                                                 COMP   4752
000445                   END                                                   COMP   4753
000445                  ELSE                                                   COMP   4754
000446                   IF LSP <> NIL THEN                                    COMP   4755
000450                    IF DEBUG AND (LSP<>INTPTR)AND (LSP↑.FORM<=SUBRANGE)  COMP   4756
000454                     THEN                                                COMP   4757
000455                     BEGIN GETBOUNDS(LSP,LMIN,LMAX);                     COMP   4758
000460                      CHECKBNDS(I,LMIN,LMAX,ASSERR)                      COMP   4759
000463                     END                                                 COMP   4760
000465               END                                                       COMP   4761
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A.

```
000465              ELSE                                                                COMP     4762
000466               BEGIN                                                              COMP     4763
000466                IF (LKIND = ACTUAL)AND (NXT <> NIL) THEN                          COMP     4764
000473                 IF NXT↑.VKIND = INDRCT THEN                                      COMP     4765
000475                  IF GATTR.KIND <> VARBL THEN                                     COMP     4766
000500                   BEGIN ERROR(142); GATTR.TYPTR := NIL END                       COMP     4767
000504                  ELSE                                                            COMP     4768
000504                   IF GATTR.PCKD THEN ERROR(127);                                 COMP     4769
000507                LOADADDRESS(GATTR,I)                                              COMP     4770
000512               END;                                                               COMP     4771
000513              IF (LKIND = ACTUAL)AND (NXT <> NIL) THEN                            COMP     4772
000520               IF NOT COMPTYPES(NXT↑.IDTYPE,GATTR.TYPTR,FALSE) THEN               COMP     4773
000530                ERRCR(142)                                                        COMP     4774
000530             END (*GATTR.TYPTR <> NIL*)                                           COMP     4775
000531            ELSE NEEDX(0,7,I);                                                    COMP     4776
000535           END (*PASS <> PROCDESC*) ;                                             COMP     4777
000535          IF (LXPAR < XPARMAX)AND NOT FTN THEN                                    COMP     4778
000541           BEGIN                                                                  COMP     4779
000541            IF I <> LXPAR THEN                                                    COMP     4780
000542             BEGIN NEEDX(LXPAR,LXPAR,LXPAR); BXIXJ(LXPAR,I);                     UPDATE22   26
000551             END;                                                                 COMP     4785
000551            LXPAR := LXPAR + 1                                                    COMP     4786
000551           END                                                                    COMP     4787
000553          ELSE                                                                    COMP     4788
000553           BEGIN                                                                  COMP     4789
000553            WITH PARAM DO                                                         COMP     4790
000553             BEGIN TYPTR := GATTR.TYPTR; KIND := VARBL;                           COMP     4791
000557              WORDACC := DRCT; PCKD := FALSE;                                     COMP     4792
000560              IF FTN AND (PASS IN [VARADDR,PROCDESC]) THEN                        COMP     4793
000563               BEGIN VLEVEL := LEVEL; CWDISPL := LC END                           COMP     4794
000565              ELSE                                                                COMP     4795
000566               BEGIN VLEVEL := 0; CWDISPL := LDSP END;                            COMP     4796
000570             END;                                                                 COMP     4797
000570            STORE(PARAM,I);                                                       COMP     4798
000573            IF FTN THEN                                                           COMP     4799
000574             BEGIN                                                                COMP     4800
000574              IF PASS = VAL THEN                                                  COMP     4801
000576               BEGIN NEEDX(6,7,I); GEN30(71B,I,6,LDSP,0);                         COMP     4802
000606                WITH PARAM DO                                                     COMP     4803
000606                 BEGIN VLEVEL := LEVEL; CWDISPL := LC END;                        COMP     4804
000611                STORE(PARAM,I)                                                    COMP     4805
000612               END;                                                               COMP     4806
000614              LC := LC + 1                                                        COMP     4807
000614             END                                                                  COMP     4808
000616           END;                                                                   COMP     4809
000616          LDSP := LDSP + 1;                                                       COMP     4810
000620          IF (LKIND = ACTUAL)AND (NXT <> NIL) THEN NXT := NXT↑.NEXT               COMP     4811
000624         UNTIL SY <> COMMA;                                                       COMP     4812
000630        IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                               COMP     4813
000635       END (*IF LPARENT*);                                                       COMP     4814
000636       FOR I := 0 TO LXPAR-1 DO ROTATEX( I ) ;                                   UPDATE9    53
000646       IF LKIND = ACTUAL THEN                                                    COMP     4815
000650        IF FTN THEN                                                              COMP     4816
000651         BEGIN CLEARREGS;                                                        COMP     4817
000653          (*SET SENTINEL FOR PARAMETER VECTOR*)                                  COMP     4818
000653          GEN15(13B,6,6,6); GEN30(51B,6,5,LC,0);                                 COMP     4819
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000654          LC := LC + 1;                                                    COMP     4820
000666          IF LC > LCMAX THEN LCMAX := LC;                                  COMP     4821
000671          (*SAVE B4/B5/B6:*)                                               COMP     4822
000671          GEN15(76B,6,4,0);                                                COMP     4823
000675          FOR I := 5 TO 6 DO                                               COMP     4824
000676           BEGIN GEN15(20B,6,0,18); GEN15(76B,0,I,0); GEN15(12B,6,6,0)     COMP     4825
000712           END;                                                            COMP     4826
000716          GEN15(56B,6,2,0);                                                COMP     4827
000722          (*SET FTN INDICATOR FOR PMD:*)                                   COMP     4828
000722          GEN30(71B,6,0,IC,0); GEN15(56B,6,2,1);                           COMP     4829
000732          (*LOAD 1. WORD OF PARAMETER VECTOR AND RJ:*)                      COMP     4830
000732          GEN30(51B,1,5,PVDISP,0);                                         COMP     4831
000737          SEARCHEXTID( FCP^.NAME,EXT1 ) ;                                  RUNMODS    11
000743          IF RUNLINKAGE AND ( LDSP > 3 ) THEN                              RUNMODS    12
000746           BEGIN  NPARMS := LDSP-4 ;                                       RUNMODS    13
000750            IF NPARMS > 1 THEN                                             RUNMODS    14
000751            BEGIN  GEN15( 54B,2,1,1 ) ;  GEN15( 63B,2,2,0 ) ;              RUNMODS    15
000760             NPARMS := NPARMS-1 ;  I := 3 ;                                RUNMODS    16
000763             WHILE ( I < 7 ) AND ( NPARMS > 0 ) DO                         RUNMODS    17
000766              BEGIN  GEN15( 54B,2,2,1 ) ;  GEN15( 63B,I,2,0 ) ;            RUNMODS    18
000776               I := I+1 ;  NPARMS := NPARMS-1 ;                            RUNMODS    19
001001              END ;                                                        RUNMODS    20
001001             IF NPARMS > 0 THEN                                            RUNMODS    21
001002              BEGIN  EXT := EXT1 ;  GEN30( 61B,7,0,0,0 ) ;                 RUNMODS    22
001007               GEN15( 57B,3,7,1 ) ;  GEN30( 62B,7,3,-6,0 ) ;               RUNMODS    23
001020               GEN15( 54B,2,2,1 ) ;  GEN15( 10B,6,2,2 ) ;                  RUNMODS    24
001030               GEN15( 55B,6,3,7 ) ;  NPARMS := NPARMS-1 ;                  RUNMODS    25
001036               WHILE NPARMS > 0 DO                                         RUNMODS    26
001040                BEGIN  GEN15( 54B,I-4,9-I,1 ) ;  GEN15( 10B,I,I-4,I-4 ) ;  RUNMODS    27
001052                 GEN15( 54B,I,13-I,1 );  I := 13-I ;  NPARMS := NPARMS-1;  RUNMODS    28
001061                END ;                                                      RUNMODS    29
001062              END ;                                                        RUNMODS    30
001062             END ;                                                         RUNMODS    31
001062            GEN15( 63B,1,1,0 ) ;                                           RUNMODS    32
001066           END ;                                                           RUNMODS    33
001066          NPARMS := LDSP-3 ;  NOOP ;  EXT := EXT1 ;                        RUNMODS    34
001073          IF RUNLINKAGE THEN RJTORUNEXT( NPARMS DIV 8,NPARMS MOD 8 )       RUNMODS    35
001077          ELSE RJTOEXT ;                                                   RUNMODS    36
001104          (*RELOAD B-REGISTERS:*)                                          COMP     4833
001104          GEN30(61B,1,0,1,0); GEN30(51B,1,0,65B,0);                        COMP     4834
001114          GEN15(63B,2,1,0); GEN15(56B,1,2,0);                              COMP     4835
001124          FOR I := 6 DOWNTO 5 DO                                           COMP     4836
001125           BEGIN GEN15(63B,I,1,0); GEN15(21B,1,0,18) END;                  COMP     4837
001140          GEN15(63B,4,1,0);                                                COMP     4838
001144          (*CLEAR FTN INDICATOR:*)                                         COMP     4839
001144          GEN15(13B,7,7,7); GEN15(56B,7,2,1)                               COMP     4840
001152         END                                                               COMP     4841
001154        ELSE                                                               COMP     4842
001155         BEGIN IF NXT <> NIL THEN ERROR(126);                              COMP     4843
001161          WITH FCP^ DO                                                     COMP     4844
001163           BEGIN                                                           COMP     4845
001163            IF PFLEV <> 1 THEN (*LOAD STATIC LINK INTO X6*)                COMP     4846
001166             IF PFLEV IN LEVELS THEN GEN15(76B,6,BRG[PFLEV],0)             COMP     4847
001175             ELSE                                                          COMP     4848
001177              BEGIN LOADBASE(PFLEV,FALSE,I);                               COMP     4849
001202               GEN15(10B,6,I,0); DECREFX(I)                                COMP     4850
```

MOORE BUSINESS FORMS, INC. F
FORM 213
PRINTED IN U.S.A.
22

```
001207              END;                                                          COMP     4851
001211             GEN30(71B,7,0,IC+1,2);                                         COMP     4852
001216              IF PFCNT = 0 THEN (*TAKE USER DECLARED NAME*)                 COMP     4853
001221               SEARCHEXTID(NAME,EXT)                                        COMP     4854
001222              ELSE                                                          COMP     4855
001225               BEGIN PFNAME(PFCNT);                                         COMP     4856
001230                SEARCHEXTID(PNAME,EXT)                                      COMP     4857
001232               END;                                                         COMP     4858
001234              GEN30(04B,0,0,0,0); NOOP                                      COMP     4859
001240            END                                                             COMP     4860
001242          END                                                               COMP     4861
001242        ELSE                                                                COMP     4862
001243         BEGIN                                                              COMP     4863
001243          WITH FCP↑, PARAM DO                                               COMP     4864
001245           BEGIN TYPTR := INTPTR; KIND := VARBL; WORDACC := DRCT;           COMP     4865
001250            VLEVEL := PFLEV; CWDISPL := PFADDR; PCKD := FALSE               COMP     4866
001252           END;                                                             COMP     4867
001253          LOAD(PARAM,I);                                                    COMP     4868
001256          IF I IN [6,7] THEN                                                UPDATE7  163
001261           BEGIN K := I ; NEEDX(0,5,I); BXIXJ( I,K ) END ;                  UPDATE7  164
001270          GEN15(63B,7,I,0) ; GEN15(73B,6,I,0); GEN15(20B,6,0,36);           UPDATE7  165
001304          LDSP := IC+1 ;                                                    UPDATE7  166
001306          IF CP <> 1 THEN LDSP := LDSP+1 ;                                  UPDATE7  167
001311          GEN30(71B,7,0,LDSP,2); GEN15(12B,7,6,7) ;                         UPDATE7  168
001321          GEN15(20B,I,0,42) ; GEN15(73B,6,I,0) ;                            UPDATE7  169
001331          GEN30(02B,7,0,0,0); NOOP                                          COMP     4871
001335        END;                                                                COMP     4872
001337         CLEARREGS;                                                         COMP     4873
001341         LC := PVDISP;                                                      COMP     4874
001343          IF FCP↑.KLASS = FUNC THEN                                         COMP     4875
001347           BEGIN B6DPL := LB6DPL;                                           COMP     4876
001350            IF B6DPL <> 3 THEN GEN30(61B,6,6,-B6DPL,0);                     COMP     4877
001356            XRGS := LXRGS;                                                  COMP     4878
001363            IF XRGS[6].XCONT <> AVAIL THEN                                  COMP     4879
001367             BEGIN NEEDX(0,7,I); GEN15(10B,I,6,0) END                       COMP     4880
001376            ELSE I := 6 ;                                                   UPDATE9   54
001400            CLEARREGS ;                                                     UPDATE9   55
001402            WITH XRGS[ I ] DO                                               UPDATE9   56
001406             BEGIN XCONT := OTHER ;  REFNR := 1 ;  END ;                    UPDATE9   57
001413            RELOADREFXRGS(LXRGS);                                           COMP     4886
001415            WITH GATTR DO                                                   COMP     4887
001415             BEGIN TYPTR := FCP↑.IDTYPE; KIND := EXPR;                      COMP     4888
001422              EXPREG := I                                                   COMP     4889
001422             END                                                            COMP     4890
001423           END                                                              COMP     4891
001423         END (*CALLNONSTANDARD*) ;                                          COMP     4892
001427                                                                            COMP     4893
001427       BEGIN (*CALL*)                                                       COMP     4894
001427         IF FCP↑.PFDECKIND = STANDARD THEN                                  COMP     4895
000012          BEGIN                                                             COMP     4896
000012           LKEY := FCP↑.KEY;                                                COMP     4897
000013           IF FCP↑.KLASS = PROC THEN                                        COMP     4898
000015            BEGIN                                                           COMP     4899
000015             CASE LKEY OF                                                   COMP     4900
000016              1,2,    (*GET,PUT*)                                           COMP     4901
000016              3,4,    (*RESET,REWRITE*)                                     COMP     4902
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000016          5,6:   (*GETSEG,PUTSEG*)                                  COMP     4903
000016                 STDFLPROCS;                                        COMP     4904
000021            7:   LINELIMIT;                                         COMP     4905
000024          8,9:   (*READ,READLN*)                                    COMP     4906
000024                 READ;                                              COMP     4907
000027        10,11:   (*WRITE,WRITELN*)                                  COMP     4908
000027                 WRITE;                                             COMP     4909
000032           12:   MESSAGE;                                           COMP     4910
000035           13:   PAGE;                                              COMP     4911
000040        14,15:   TIMEDATE;                                          COMP     4912
000043           16:   HALT;                                              COMP     4913
000046           17:   PACK;                                              COMP     4914
000051           18:   UNPACK;                                            COMP     4915
000054        19,20:   NEWDISPOSE;                                        COMP     4916
000057           21:   RELEASE;                                           COMP     4917
000062        22,23:   (*DUMMY*)                                         COMP     4918
000062          END                                                       COMP     4919
000112         END                                                        COMP     4920
000112        ELSE                                                        COMP     4921
000113         BEGIN                                                      COMP     4922
000113          CASE LKEY OF                                              COMP     4923
000114           1,2,  (*EOF,EOS*)                  (*PREDICATES*)        COMP     4924
000114           3:    (*EOLN*)                                           COMP     4925
000114                 STDFLFUNCS;                                        COMP     4926
000117           4:    ODDP;                                              COMP     4927
000122           5:    UNDEFINEDP;                                        COMP     4928
000125           6:    ROUNDF;                    (*FUNCTIONS*)           COMP     4929
000130           7:    TRUNCF;                                            COMP     4930
000133           8:    EXPOF;                                             COMP     4931
000136           9:    ABSF;                                              COMP     4932
000141           10:   SQRF;                                              COMP     4933
000144           11:   ORDF;                                              COMP     4934
000147           12:   CHRF;                                              COMP     4935
000152        13,14:   PREDSUCCF;                                         COMP     4936
000155           15:   CARDF;                                             COMP     4937
000160           16:   CLOCKF;                                            COMP     4938
000163        17,18:   (*DUMMY*);                                         COMP     4939
000164        19,20,                                                      COMP     4940
000164        21,22,                                                      COMP     4941
000164        23,24:   STDARITHFUNCS                                      COMP     4942
000164          END                                                       COMP     4943
000217         END;                                                       COMP     4944
000217       END (*STANDARD PROCEDURES AND FUNCTIONS*)                    COMP     4945
000217      ELSE CALLNONSTANDARD                                          COMP     4946
000220     END (*CALL*) ;                                                 COMP     4947
000226                                                                    COMP     4948
000003     PROCEDURE EXPRESSION;                                          COMP     4949
000004      VAR LATTR: ATTR; LOP: OPERATOR; WRDS,LADDR: ADDRRANGE;        COMP     4950
000021        BTS : BITRANGE ;  I,J,K,L,M,N,II,JJ : REGNR ;               UPDATE9    58
000032        LOPCD: OPRANGE; FLSJMP: 0..3;                               COMP     4952
000034                                                                    COMP     4953
000003      PROCEDURE SIMPLEEXPRESSION(FSYS: SETOFSYS);                   COMP     4954
000004       VAR LATTR: ATTR; LOP: OPERATOR; SIGNED: BOOLEAN;             COMP     4955
000020         I,J,K: REGNR;                                              COMP     4956
000023                                                                    COMP     4957
000003      PROCEDURE BOOLOP(VAR FATTR: ATTR; FOP: OPERATOR);             COMP     4958
```

MOORE BUSINESS FORMS, INC. FORM 213 PRINTED IN U.S.A. 22

```
000005      (*GENERATE CODE FOR <FATTR> <FOP> <GATTR> (FOP IN [ANDOP,OROP])*)  COMP  4959
000005      (*RESULTING ATTRIBUTES IN GATTR*)                                  COMP  4960
000005      VAR I,J,K,L: REGNR; IND1,IND2,SHFT,OPCD: INTEGER;                  COMP  4961
000015                                                                         COMP  4962
000003      PROCEDURE LDOPD(VAR FATTR: ATTR; VAR FI: REGNR;VAR FIND: INTEGER); COMP  4963
000006       (*LOAD OPERAND DESCRIBED BY FATTR INTO X-FI AND SET INDICATOR     COMP  4964
000006       FIND TO DISTINGUISH BETWEEN 5 CASES:                              COMP  4965
000006          VALUE OF FIND:     0     1     2     3     4                  COMP  4966
000006          X-FI CONTAINS:  LOGICAL ZR    NZ    PL    NG*)               COMP  4967
000006       VAR I: REGNR;                                                     COMP  4968
000007      BEGIN                                                              COMP  4969
000007       WITH FATTR DO                                                     COMP  4970
000011        IF KIND = COND THEN                                              COMP  4971
000012         BEGIN                                                           COMP  4972
000012          IF CONDCD IN [ZR,NZ] THEN                                      COMP  4973
000015           BEGIN NEEDX(0,7,I); GEN15(13B,I,I,I); GEN15(37B,I,I,CDR);     COMP  4974
000032            IF CONDCD = ZR THEN GEN15(13B,I,CDR,I)                       COMP  4975
000037            ELSE GEN15(17B,I,CDR,I);                                     COMP  4976
000047            DECREFX(CDR); FI := I                                        COMP  4977
000054           END                                                           COMP  4978
000055          ELSE FI := CDR;                                                COMP  4979
000056          FIND := ORD(CONDCD) + 1                                        COMP  4980
000060         END                                                             COMP  4981
000061        ELSE                                                             COMP  4982
000061         BEGIN LOAD(FATTR,FI); FIND := 0 END                             COMP  4983
000065      END (*LDOPD*) ;                                                    COMP  4984
000071                                                                         COMP  4985
000071      BEGIN (*BOOLOP*) LDOPD(FATTR,I,IND1); LDOPD(GATTR,J,IND2);         COMP  4986
000017       IF IND2 < IND1 THEN (*TRANSPOSE OPS*)                             COMP  4987
000021        BEGIN K := I; I := J; J := K;                                    COMP  4988
000023         K := IND1; IND1 := IND2; IND2 := K                              COMP  4989
000024        END;                                                             COMP  4990
000025       IF (IND1=0)AND (IND2 IN [3,4]) THEN                               COMP  4991
000031        BEGIN                                                            COMP  4992
000031         IF FOP =ANDOP THEN                                              COMP  4993
000033          BEGIN K := J; SHFT := 1 END                                    COMP  4994
000034         ELSE                                                            COMP  4995
000035          BEGIN K := I; SHFT := 59 END;                                  COMP  4996
000040         WITH XRGS[K] DO                                                 COMP  4997
000045          IF REFNR = 1 THEN                                              COMP  4998
000047           BEGIN GEN15(20B,K,0,SHFT); XCONT := OTHER END                 COMP  4999
000054          ELSE                                                           COMP  5000
000055           BEGIN NEEDX(0,7,L); GEN15(10B,L,K,0);                         COMP  5001
000064            GEN15(20B,L,0,SHFT); DECREFX(K);                             COMP  5002
000073            IF FOP = ANDOP THEN J := L ELSE I := L                       COMP  5003
000076           END                                                           COMP  5004
000077        END;                                                             COMP  5005
000077       NEEDX(0,7,K);                                                     COMP  5006
000103       (*SET RESULT ATTRIBUTES:*)                                        COMP  5007
000103       WITH GATTR DO                                                     COMP  5008
000103        BEGIN TYPTR := BOOLPTR;                                          COMP  5009
000107         IF IND1 = 0 THEN                                                COMP  5010
000110          BEGIN KIND := EXPR; EXPREG := K END                            COMP  5011
000112         ELSE                                                            COMP  5012
000113          BEGIN KIND := COND; CONDCD := PL; CDR := K END                 COMP  5013
000116        END;                                                             COMP  5014
```

MOORE BUSINESS FORMS, INC. F
FORM 1413
PRINTED IN U.S.A.
22

```
000116      IF FOP = ANDOP THEN OPCD := 11B ELSE OPCD := 12B;                    COMP  5015
000122      IF IND2 = 4 THEN                                                     COMP  5016
000124       IF IND1 <> 4 THEN OPCD := OPCD + 4                                  COMP  5017
000125       ELSE                                                                COMP  5018
000127        BEGIN OPCD := 23B - OPCD; GATTR.CONDCD := NG END;                  COMP  5019
000134      GEN15(OPCD,K,I,J);                                                   COMP  5020
000141      IF (FOP = OROP)AND (IND1 = 0)AND (IND2 <> 0) THEN                    COMP  5021
000147       BEGIN                                                               COMP  5022
000147        WITH GATTR DO                                                      COMP  5023
000147         BEGIN KIND := COND; CONDCD := PL; CDR := K END;                   COMP  5024
000154        IF IND2 IN [1,2] THEN GEN15(20B,K,0,59)                            COMP  5025
000157       END;                                                                COMP  5026
000161      DECREFX(I); DECREFX(J)                                               COMP  5027
000165     END (*BOOLOP*) ;                                                      COMP  5028
000173                                                                           COMP  5029
000003      PROCEDURE TERM(FSYS: SETOFSYS);                                      COMP  5030
000004       VAR LATTR: ATTR; LOP: OPERATOR;                                     COMP  5031
000017         I,J,K,L: REGNR; LREC: CSTREC;                                     COMP  5032
000024                                                                           COMP  5033
000003       PROCEDURE LOADANDCHECKX(VAR FI: REGNR);                             COMP  5034
000004        (*LOAD GATTR INTO X-FI, TEST IT AGAINST ZERO*)                      COMP  5035
000004       BEGIN                                                               COMP  5036
000004        IF GATTR.KIND = CST THEN                                           COMP  5037
000013         BEGIN IF GATTR.CVAL.IVAL = 0 THEN ERROR(300);                     COMP  5038
000016          LOAD(GATTR,FI)                                                   COMP  5039
000022         END                                                               COMP  5040
000023        ELSE                                                               COMP  5041
000024         BEGIN LOAD(GATTR,FI);                                             COMP  5042
000027          IF DEBUG THEN                                                    COMP  5043
000030           BEGIN GEN30(51B,0,0,IC,0);                                      COMP  5044
000035            GEN30(03B,0,FI,DIVERR,0)                                       COMP  5045
000040           END                                                             COMP  5046
000043         END                                                               COMP  5047
000043       END (*LOADANDCHECKX*) ;                                             COMP  5048
000047                                                                           COMP  5049
000003       PROCEDURE FACTOR(FSYS: SETOFSYS);                                   COMP  5050
000004        VAR LCP: CTP; LSP: STP; I,J,K,L,M: REGNR; LCSTATTR,LATTR: ATTR;    COMP  5051
000037          VARPART,EXITLOOP: BOOLEAN; N: INTEGER;                           COMP  5052
000042       BEGIN                                                               COMP  5053
000042        IF NOT (SY IN FACBEGSYS) THEN                                      COMP  5054
000011         BEGIN ERROR(58); SKIP(FSYS+FACBEGSYS);                            COMP  5055
000016          GATTR.TYPTR := NIL                                               COMP  5056
000016         END;                                                              COMP  5057
000022        REPEAT                                                             COMP  5058
000022         IF SY IN FACBEGSYS THEN                                           COMP  5059
000025         BEGIN                                                             COMP  5060
000025          CASE SY OF                                                       COMP  5061
000026  (*ID*)   IDENT:                                                         COMP  5062
000026            BEGIN SEARCHID([KONST,VARS,FIELD,FUNC],LCP);                   COMP  5063
000030             INSYMBOL;                                                     COMP  5064
000031             CASE LCP^.KLASS OF                                            COMP  5065
000034              KONST:                                                       COMP  5066
000034               WITH LCP^, GATTR DO                                         COMP  5067
000035                BEGIN TYPTR := IDTYPE; KIND := CST;                        COMP  5068
000041                 CVAL := VALUES                                            COMP  5069
000041                END;                                                       COMP  5070
```

MOORE BUSINESS FORMS, INC. F
FORM 313
PRINTED IN U.S.A
22

```
000044                VARS,                                                    COMP  5071
000044                FIELD:                                                   COMP  5072
000044                 SELECTOR(FSYS,LCP);                                     COMP  5073
000050                FUNC:                                                    COMP  5074
000050                 CALL(FSYS,LCP)                                          COMP  5075
000051               END                                                       COMP  5076
000061              END;                                                       COMP  5077
000062  (*CST*)     INTCONST:                                                  COMP  5078
000062               BEGIN                                                     COMP  5079
000062                WITH GATTR DO                                            COMP  5080
000062                 BEGIN TYPTR := INTPTR; KIND := CST;                     COMP  5081
000067                  CVAL.IVAL := IVAL                                      COMP  5082
000067                 END;                                                    COMP  5083
000070               INSYMBOL                                                  COMP  5084
000070              END;                                                       COMP  5085
000072              REALCONST:                                                 COMP  5086
000072               BEGIN                                                     COMP  5087
000072                WITH GATTR DO                                            COMP  5088
000072                 BEGIN TYPTR := REALPTR; KIND := CST;                    COMP  5089
000077                  CVAL.RVAL := RVAL                                      COMP  5090
000077                 END;                                                    COMP  5091
000100               INSYMBOL                                                  COMP  5092
000100              END;                                                       COMP  5093
000102              CHARCONST:                                                 COMP  5094
000102               BEGIN                                                     COMP  5095
000102                WITH GATTR DO                                            COMP  5096
000102                 BEGIN TYPTR := CHARPTR; KIND := CST;                    COMP  5097
000107                  CVAL.IVAL := IVAL                                      COMP  5098
000107                 END;                                                    COMP  5099
000110               INSYMBOL                                                  COMP  5100
000110              END;                                                       COMP  5101
000112              STRINGCONST:                                               COMP  5102
000112               BEGIN                                                     COMP  5103
000112                WITH GATTR DO                                            COMP  5104
000112                 BEGIN STRINGTYPE(TYPTR); KIND := CST;                   COMP  5105
000123                  CVAL.VALP := CONSTP                                    COMP  5106
000123                 END;                                                    COMP  5107
000124               INSYMBOL                                                  COMP  5108
000124              END;                                                       COMP  5109
000126   (*(*)      LPARENT:                                                   COMP  5110
000126               BEGIN INSYMBOL; EXPRESSION(FSYS+[RPARENT]);               COMP  5111
000133                IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)              COMP  5112
000140              END;                                                       COMP  5113
000142  (*NOT*)    NOTSY:                                                     COMP  5114
000142               BEGIN INSYMBOL; FACTOR(FSYS);                             COMP  5115
000146                IF COMPTYPES(GATTR.TYPTR,BOOLPTR,FALSE) AND              COMP  5116
000154                 (GATTR.TYPTR <> NIL) THEN                               COMP  5117
000162                 WITH GATTR DO                                           COMP  5118
000162                  IF KIND = COND THEN                                    COMP  5119
000163                   CASE CONDCD OF                                        COMP  5120
000165                    ZR: CONDCD := NZ;                                    COMP  5121
000166                    NZ: CONDCD := ZR;                                    COMP  5122
000167                    PL: CONDCD := NG;                                    COMP  5123
000171                    NG: CONDCD := PL                                     COMP  5124
000171                   END                                                   COMP  5125
000176                  ELSE                                                   COMP  5126
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000177                  BEGIN LOAD(GATTR,I); NEEDX(0,7,K);                      COMP    5127
000205                   GEN15(43B,K,0,59); GEN15(17B,K,I,K); DECREFX(I);       COMP    5128
000223                   GATTR.EXPREG := K                                      COMP    5129
000223                  END                                                     COMP    5130
000227                ELSE                                                      COMP    5131
000230                 BEGIN ERROR(135); GATTR.TYPTR := NIL END                 COMP    5132
000236               END;                                                       COMP    5133
000237  (*[*)      LBRACK:                                                      COMP    5134
000237               BEGIN INSYMBOL;                                            COMP    5135
000240                NEW(LSP,POWER);                                           COMP    5136
000242                WITH LSP^ DO                                              COMP    5137
000244                 BEGIN ELSET := NIL; FORM := POWER; PCKDSET := FALSE;     COMP    5138
000253                  FTYPE := FALSE;                                         COMP    5139
000254                  WITH SIZE DO                                            COMP    5140
000255                   BEGIN WORDS := 1; BITS := 0 END                        COMP    5141
000261                 END;                                                     COMP    5142
000261                VARPART := FALSE;                                         COMP    5143
000262                WITH LCSTATTR DO                                          COMP    5144
000262                 BEGIN TYPTR := LSP; KIND := CST; CVAL.PVAL := [ ]        COMP    5145
000263                 END;                                                     COMP    5146
000264                IF SY = RBRACK THEN INSYMBOL                              COMP    5147
000266                ELSE                                                      COMP    5148
000270                 BEGIN                                                    COMP    5149
000270                  (*LOOP UNTIL SY <> COMMA:*)                             COMP    5150
000270                  REPEAT EXPRESSION(FSYS+[COMMA,COLON,RBRACK]);           COMP    5151
000274                   IF GATTR.TYPTR <> NIL THEN                             COMP    5152
000301                    IF GATTR.TYPTR^.FORM > SUBRANGE THEN                  COMP    5153
000303                     BEGIN ERROR(136); GATTR.TYPTR := NIL END             COMP    5154
000310                   ELSE                                                   COMP    5155
000310                    IF NOT COMPTYPES(LSP^.ELSET,GATTR.TYPTR,FALSE)        COMP    5156
000312                     THEN ERROR(137);                                     COMP    5157
000317                  IF SY = COLON THEN                                      COMP    5158
000321                   BEGIN LATTR := GATTR; INSYMBOL;                        COMP    5159
000330                    EXPRESSION(FSYS+[COMMA,RBRACK]);                      COMP    5160
000334                    IF GATTR.TYPTR <> NIL THEN                            COMP    5161
000341                     IF GATTR.TYPTR^.FORM > SUBRANGE THEN                 COMP    5162
000343                      BEGIN ERROR(136); GATTR.TYPTR := NIL                COMP    5163
000344                      END                                                 COMP    5164
000350                     ELSE                                                 COMP    5165
000350                      IF NOT COMPTYPES( LATTR.TYPTR,GATTR.TYPTR,          UPDATE7  170
000351                       FALSE) THEN ERROR(137);                            COMP    5167
000356                    IF (LATTR.TYPTR <> NIL)AND (GATTR.TYPTR <> NIL)       COMP    5168
000353                     THEN                                                 COMP    5169
000365                     BEGIN                                                COMP    5170
000365                      IF (LATTR.KIND = CST)AND (GATTR.KIND = CST)         COMP    5171
000367                       THEN                                               COMP    5172
000371                       BEGIN                                              COMP    5173
000371                        IF (LATTR.CVAL.IVAL<0)OR (GATTR.CVAL              COMP    5174
000372                        .IVAL>58) THEN ERROR(304);                        COMP    5175
000376                         FOR N := LATTR.CVAL.IVAL TO GATTR                COMP    5176
000377                          .CVAL.IVAL DO                                   COMP    5177
000403                          LCSTATTR.CVAL.PVAL := LCSTATTR                  COMP    5178
000405                           .CVAL.PVAL+[N]                                 COMP    5179
000406                       END                                                COMP    5180
000410                      ELSE                                                COMP    5181
000411                       BEGIN LOAD(LATTR,I);                               COMP    5182
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000413                    IF DEBUG THEN CHECKBNDS(I,0,58,                    COMP    5183
000416                     ASSERR);                                          COMP    5184
000421                    LOAD(GATTR,J);                                     COMP    5185
000426                    IF DEBUG THEN CHECKBNDS(J,0,58,                    COMP    5186
000431                     ASSERR);                                          COMP    5187
000434                    NEEDX(0,7,K); GEN15(37B,K,J,I);                    COMP    5188
000444                    DECREFX(I); NEEDX(0,7,I);                          COMP    5189
000454                    GEN15(13B,I,I,I);                                  COMP    5190
000460                    IF CP = 1 THEN                                     COMP    5191
000464                     GEN30(03B,3,K,IC+1,2)                             COMP    5192
000467                    ELSE GEN30(03B,3,K,IC+2,2);                        COMP    5193
000475                    NEEDB(L); GEN15(63B,L,K,0);                        COMP    5194
000504                    GEN15(43B,K,0,1); GEN15(23B,K,L,K);                COMP    5195
000514                    GEN15(63B,L,J,1); DECREFX(J);                      COMP    5196
000525                    GEN15(22B,I,L,K); DECREFX(K);                      COMP    5197
000536                    BRGS[L].BCONT := FREE; NOOP;                       COMP    5198
000546                    IF VARPART THEN                                    COMP    5199
000547                     BEGIN GEN15(12B,M,I,M); DECREFX(I)                COMP    5200
000555                     END                                               COMP    5201
000560                    ELSE                                               COMP    5202
000561                     BEGIN M := I; VARPART := TRUE END                 COMP    5203
000563                   END                                                 COMP    5204
000563                 END                                                   COMP    5205
000563               END (*COLON*)                                           COMP    5206
000563             ELSE                                                      COMP    5207
000564               IF GATTR.TYPTR <> NIL THEN                              COMP    5208
000571                IF GATTR.KIND = CST THEN                               COMP    5209
000572                 BEGIN                                                 COMP    5210
000572                  IF (GATTR.CVAL.IVAL<0)OR (GATTR.CVAL.IVAL            COMP    5211
000573                   >58) THEN ERROR(304);                               COMP    5212
000576                  LCSTATTR.CVAL.PVAL := LCSTATTR.CVAL.PVAL             COMP    5213
000576                  +[GATTR.CVAL.IVAL]                                   COMP    5214
000602                 END                                                   COMP    5215
000603                ELSE                                                   COMP    5216
000604                 BEGIN LOAD(GATTR,I);                                  COMP    5217
000606                  IF DEBUG THEN CHECKBNDS(I,0,58,ASSERR);              COMP    5218
000614                  NEEDB(J);                                            COMP    5219
000617                  GEN15(63B,J,I,0); DECREFX(I);                        COMP    5220
000627                  NEEDX(0,7,I); GEN15(76B,I,1,0);                      COMP    5221
000637                  GEN15(22B,I,J,I); BRGS[J].BCONT := FREE;             COMP    5222
000652                  IF VARPART THEN                                      COMP    5223
000653                   BEGIN GEN15(12B,M,I,M); DECREFX(I) END              COMP    5224
000663                  ELSE                                                 COMP    5225
000664                   BEGIN M := I; VARPART := TRUE END                   COMP    5226
000666                 END;                                                  COMP    5227
000666               IF GATTR.TYPTR <> NIL THEN                              UPDATE7  171
000673                BEGIN IF COMPTYPES(GATTR.TYPTR,REALPTR,FALSE)          UPDATE7  172
000674                  THEN ERROR(109) ;                                    UPDATE7  173
000700                 LSP^.ELSET := GATTR.TYPTR ;                           UPDATE7  174
000706                END ;                                                  UPDATE7  175
000706              EXITLOOP := SY <> COMMA;                                 COMP    5229
000711              IF NOT EXITLOOP THEN INSYMBOL                            COMP    5230
000712             UNTIL EXITLOOP;                                           COMP    5231
000714              IF SY = RBRACK THEN INSYMBOL ELSE ERROR(12)              COMP    5232
000720             END;                                                      COMP    5233
000721           IF VARPART THEN                                             COMP    5234
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000722                BEGIN                                                  COMP   5235
000722                 IF LCSTATTR.CVAL.PVAL <> [ ] THEN                     COMP   5236
000724                  BEGIN LOAD(LCSTATTR,I); GEN15(12B,M,I,M);            COMP   5237
000734                   DECREFX(I)                                          COMP   5238
000735                  END;                                                 COMP   5239
000740                 WITH GATTR DO                                         COMP   5240
000740                  BEGIN TYPTR := LSP; KIND := EXPR; EXPREG := M        COMP   5241
000745                  END                                                  COMP   5242
000746                END                                                    COMP   5243
000746               ELSE GATTR := LCSTATTR                                  COMP   5244
000746              END                                                      COMP   5245
000754            END (*CASE*) ;                                             COMP   5246
000771            IF NOT (SY IN FSYS) THEN                                   COMP   5247
000774             BEGIN ERROR(6); SKIP(FSYS+FACBEGSYS) END                  COMP   5248
001000           END (*IF*)                                                  COMP   5249
001000          UNTIL SY IN FSYS                                             COMP   5250
001000         END (*FACTOR*) ;                                              COMP   5251
001010                                                                       COMP   5252
001010        BEGIN (*TERM*)                                                 COMP   5253
001010         FACTOR(FSYS+[MULOP]);                                         COMP   5254
000011         WHILE SY = MULOP DO                                           COMP   5255
000013          BEGIN LATTR := GATTR; LOP := OP;                             COMP   5256
000022           INSYMBOL; FACTOR(FSYS+[MULOP]); K := 0;                     COMP   5257
000027           IF (LATTR.TYPTR <> NIL)AND (GATTR.TYPTR <> NIL) THEN        COMP   5258
000036            CASE LOP OF                                                COMP   5259
000037  (***)        MUL:  IF COMPTYPES(LATTR.TYPTR,INTPTR,FALSE)AND         COMP   5260
000042                  COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN             COMP   5261
000052                  BEGIN                                                COMP   5262
000052                   IF LATTR.KIND = CST THEN                            COMP   5263
000054                    EXPREP(LATTR.CVAL.IVAL,LREC)                       COMP   5264
000055                   ELSE LREC.CKIND := NOP;                             COMP   5265
000063                   IF LREC.CKIND = NOP THEN                            COMP   5266
000066                    BEGIN LOAD(LATTR,I);                               COMP   5267
000071                     IF GATTR.KIND = CST THEN                          COMP   5268
000075                      EXPREP(GATTR.CVAL.IVAL,LREC)                     COMP   5269
000076                    END                                                COMP   5270
000077                   ELSE LOAD(GATTR,I);                                 COMP   5271
000105                   IF LREC.CKIND = NOP THEN                            COMP   5272
000110                    BEGIN LOAD(GATTR,J); OPERATION(42B,K,I,J);         COMP   5273
000120                     NEEDX(0,7,I); GEN15(13B,I,I,I);                   COMP   5274
000130                     GEN15(36B,K,K,I); DECREFX(I)                      COMP   5275
000135                    END                                                COMP   5276
000137                   ELSE                                                COMP   5277
000140                    OPTMULT(I,LREC,NOT (XRGS[I].XCONT IN [SIMPVAR,     COMP   5278
000141                    INDVAR]),K)                                        COMP   5279
000150                 END                                                   COMP   5280
000152                ELSE                                                   COMP   5281
000153                 BEGIN LOAD(LATTR,I); LOAD(GATTR,J);                   COMP   5282
000164                  IF COMPTYPES(LATTR.TYPTR,INTPTR,FALSE) THEN          COMP   5283
000173                   BEGIN PACKANDNORM(I);                               COMP   5284
000176                    LATTR.TYPTR := REALPTR                             COMP   5285
000176                   END                                                 COMP   5286
000200                  ELSE                                                 COMP   5287
000200                   IF COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN         COMP   5288
000207                    BEGIN PACKANDNORM(J);                              COMP   5289
000211                     GATTR.TYPTR := REALPTR                            COMP   5290
```

MOORE BUSINESS FORMS, INC. F 1413 PRINTED IN U.S.A. 22

```
000211                    END;                                                  COMP    5291
000215                  IF COMPTYPES(LATTR.TYPTR,REALPTR,FALSE)                 COMP    5292
000217                   AND COMPTYPES(GATTR.TYPTR,REALPTR,FALSE) THEN          COMP    5293
000232                   OPERATION(41B,K,I,J)                                   COMP    5294
000234                  ELSE                                                    COMP    5295
000237                   IF (LATTR.TYPTR↑.FORM = POWER)                         COMP    5296
000242                    AND COMPTYPES(LATTR.TYPTR,GATTR.TYPTR,FALSE)          COMP    5297
000247                    THEN                                                  COMP    5298
000254                    OPERATION(11B,K,I,J)                                  COMP    5299
000256                   ELSE BEGIN ERROR(134); GATTR.TYPTR := NIL END          COMP    5300
000266                END;                                                      COMP    5301
000267 (*/*)         RDIV: BEGIN LOAD(LATTR,I); LOAD(GATTR,J);                 UPDATE4   4
000276                   IF COMPTYPES(LATTR.TYPTR,INTPTR,FALSE) THEN            COMP    5303
000305                    BEGIN PACKANDNORM(I);                                 COMP    5304
000310                    LATTR.TYPTR := REALPTR                                COMP    5305
000310                    END;                                                  COMP    5306
000312                   IF COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN            COMP    5307
000321                    BEGIN PACKANDNORM(J);                                 COMP    5308
000323                    GATTR.TYPTR := REALPTR                                COMP    5309
000323                    END;                                                  COMP    5310
000327                   IF COMPTYPES(LATTR.TYPTR,REALPTR,FALSE)                COMP    5311
000331                    AND COMPTYPES(GATTR.TYPTR,REALPTR,FALSE) THEN         COMP    5312
000344                    OPERATION(45B,K,I,J)                                  COMP    5313
000346                   ELSE BEGIN ERROR(134); GATTR.TYPTR := NIL END          COMP    5314
000356                 END;                                                     COMP    5315
000357 (*DIV*)       IDIV: IF COMPTYPES(LATTR.TYPTR,INTPTR,FALSE)               COMP    5316
000361                   AND COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN           COMP    5317
000372                  BEGIN LOAD(LATTR,I);                                    COMP    5318
000376                   IF GATTR.KIND = CST THEN                               COMP    5319
000402                    EXPREP(GATTR.CVAL.IVAL,LREC)                          COMP    5320
000403                   ELSE LREC.CKIND := NOP;                                COMP    5321
000410                   IF LREC.CKIND = PUREP THEN                             COMP    5322
000413                    IF LREC.EXP <> 0 THEN                                 COMP    5323
000415                     BEGIN NEEDX(0,7,K); GEN15(10B,K,I,0);                COMP    5324
000425                      DECREFX(I); GEN15(21B,K,0,LREC.EXP);                COMP    5325
000435                      NEEDX(0,7,I); GEN15(13B,I,I,I);                     COMP    5326
000445                      GEN15(36B,K,K,I); DECREFX(I)                        COMP    5327
000452                     END                                                  COMP    5328
000454                    ELSE K := I                                           COMP    5329
000455                   ELSE                                                   COMP    5330
000457                    BEGIN LOADANDCHECK(J);                                COMP    5331
000461                     IF DEBUG THEN PACKOFL(I); DECREFX(I);                COMP    5332
000470                     NEEDX(0,7,K); GEN15(27B,K,0,I);                      COMP    5333
000500                     PACKANDNORM(J); GEN15(44B,K,K,J); NEEDB(I);          COMP    5334
000512                     GEN15(25B,K,I,K); GEN15(22B,K,I,K);                  COMP    5335
000522                     BRGS[I].BCONT := FREE; GEN15(13B,J,J,J);             COMP    5336
000533                     GEN15(36B,K,K,J); DECREFX(J)                         COMP    5337
000540                    END;                                                  COMP    5338
000542                   GATTR.TYPTR := INTPTR                                  COMP    5339
000542                  END                                                     COMP    5340
000546                ELSE BEGIN ERROR(134); GATTR.TYPTR := NIL END;            COMP    5341
000554 (*MOD*)       IMOD: IF COMPTYPES(LATTR.TYPTR,INTPTR,FALSE)               COMP    5342
000556                   AND COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN           COMP    5343
000567                    BEGIN LOAD(LATTR,I);                                  COMP    5344
000573                     IF GATTR.KIND = CST THEN                             COMP    5345
000577                      EXPREP(GATTR.CVAL.IVAL,LREC)                        COMP    5346
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 27

```
000600                    ELSE LREC.CKIND := NOP;                                  COMP    5347
000605                    IF LREC.CKIND = PUREP THEN                               COMP    5348
000610                     BEGIN NEEDX(0,7,K); GEN15(10B,K,I,0);                   COMP    5349
000620                      GEN15(21B,K,0,LREC.EXP);                               COMP    5350
000625                      GEN15(20B,K,0,LREC.EXP);                               COMP    5351
000632                      GEN15(37B,K,I,K); DECREFX(I)                           COMP    5352
000637                     END                                                     COMP    5353
000641                    ELSE                                                     COMP    5354
000642                     BEGIN LOADANDCHECKX(J);                                 COMP    5355
000644                      IF DEBUG THEN PACKOFL(J);                              COMP    5356
000650                      NEEDX(0,7,K); NEEDX(0,7,L);                            COMP    5357
000660                      GEN15(27B,L,0,I); GEN15(27B,K,0,J);                    COMP    5358
000670                      GEN15(24B,K,0,K); GEN15(44B,K,L,K);                    COMP    5359
000700                      DECREFX(L); NEEDB(L); GEN15(26B,K,L,K);                COMP    5360
000712                      GEN15(22B,K,L,K); BRGS[L].BCONT := FREE;               COMP    5361
000725                      IF LREC.CKIND <> NOP THEN                              COMP    5362
000727                       OPTMULT(K,LREC,TRUE,K)                                COMP    5363
000730                      ELSE GEN15(42B,K,J,K);                                 COMP    5364
000736                      DECREFX(J); GEN15(37B,K,I,K);                          COMP    5365
000745                      DECREFX(I)                                             COMP    5366
000746                     END;                                                    COMP    5367
000750                   GATTR.TYPTR := INTPTR                                     COMP    5368
000750                  END                                                        COMP    5369
000754                 ELSE BEGIN ERROR(134); GATTR.TYPTR := NIL END;              COMP    5370
000762 (*AND*)        ANDOP:IF COMPTYPES(LATTR.TYPTR,BOOLPTR,FALSE)AND            COMP    5371
000765                  COMPTYPES(GATTR.TYPTR,BOOLPTR,FALSE) THEN                  COMP    5372
000775                   BOOLOP(LATTR,ANDOP)                                       COMP    5373
000776                  ELSE BEGIN ERROR(134); GATTR.TYPTR := NIL END              COMP    5374
001006                 END (*CASE*)                                               COMP    5375
001014                ELSE GATTR.TYPTR := NIL;                                    COMP    5376
001016                IF LOP <> ANDOP THEN                                        COMP    5377
001020                 WITH GATTR DO                                              COMP    5378
001020                  BEGIN KIND := EXPR; EXPREG := K END;                      COMP    5379
001024               END (*WHILE*)                                                COMP    5380
001024             END (*TERM*) ;                                                 COMP    5381
001031                                                                            COMP    5382
001031           BEGIN (*SIMPLEEXPRESSION*)                                       COMP    5383
001031            SIGNED := FALSE;                                                COMP    5384
000010            IF OP IN [PLUS,MINUS] THEN                                      COMP    5385
000012             BEGIN SIGNED := OP = MINUS; INSYMBOL END;                      COMP    5386
000016            TERM(FSYS+[ADDOP]);                                             COMP    5387
000021            IF SIGNED THEN                                                  COMP    5388
000022             IF COMPTYPES(GATTR.TYPTR,INTPTR,FALSE)OR COMPTYPES(GATTR.TYPTR,COMP    5389
000032              REALPTR,FALSE) THEN                                           COMP    5390
000037              BEGIN LOAD(GATTR,I); NEEDX(0,7,K); GEN15(13B,K,K,K);          COMP    5391
000053               GEN15(37B,K,K,I); DECREFX(I); GATTR.EXPREG := K               COMP    5392
000062              END                                                           COMP    5393
000065             ELSE                                                           COMP    5394
000066              BEGIN ERROR(134); GATTR.TYPTR := NIL END;                     COMP    5395
000073            WHILE SY = ADDOP DO                                             COMP    5396
000075             BEGIN LATTR := GATTR; LOP := OP;                               COMP    5397
000103              INSYMBOL; TERM(FSYS+[ADDOP]); K := 0;                         COMP    5398
000110              IF (LATTR.TYPTR <> NIL)AND (GATTR.TYPTR <> NIL) THEN          COMP    5399
000116               CASE LOP OF                                                  COMP    5400
000120           (*+*)PLUS:                                                       COMP    5401
000120                BEGIN LOAD(LATTR,I); LOAD(GATTR,J);                         COMP    5402
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000126            IF (LATTR.TYPTR↑.FORM = POWER)                                  COMP  5403
000131             AND COMPTYPES(LATTR.TYPTR,GATTR.TYPTR,FALSE) THEN              COMP  5404
000142             OPERATION(12B,K,I,J)                                           COMP  5405
000144            ELSE                                                            COMP  5406
000147              IF COMPTYPES(LATTR.TYPTR,INTPTR,FALSE)                        COMP  5407
000153              AND COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN                  COMP  5408
000170              OPERATION(36B,K,I,J)                                          COMP  5409
000172             ELSE                                                           COMP  5410
000174              BEGIN                                                         COMP  5411
000174               IF COMPTYPES(LATTR.TYPTR,INTPTR,FALSE) THEN                  COMP  5412
000203                BEGIN PACKANDNORM(I);                                       COMP  5413
000206                 LATTR.TYPTR := REALPTR                                     COMP  5414
000206                END                                                         COMP  5415
000210               ELSE                                                         COMP  5416
000210                IF COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN                 COMP  5417
000216                 BEGIN PACKANDNORM(J);                                      COMP  5418
000220                  GATTR.TYPTR := REALPTR                                    COMP  5419
000220                 END;                                                       COMP  5420
000223               IF COMPTYPES(LATTR.TYPTR,REALPTR,FALSE)                      COMP  5421
000225                AND COMPTYPES(GATTR.TYPTR,REALPTR,FALSE) THEN               COMP  5422
000240                BEGIN OPERATION(34B,K,I,J); GEN15(24B,K,0,K) END            COMP  5423
000250               ELSE BEGIN ERROR(134); GATTR.TYPTR := NIL END                COMP  5424
000256              END;                                                          COMP  5425
000256            END;                                                            COMP  5426
000257      (*-*)MINUS:                                                           COMP  5427
000257            BEGIN LOAD(LATTR,I); LOAD(GATTR,J);                             COMP  5428
000265            IF COMPTYPES(LATTR.TYPTR,INTPTR,FALSE)                          COMP  5429
000271             AND COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN                   COMP  5430
000306             OPERATION(37B,K,I,J)                                           COMP  5431
000310            ELSE                                                            COMP  5432
000312             BEGIN                                                          COMP  5433
000312              IF COMPTYPES(LATTR.TYPTR,INTPTR,FALSE) THEN                   COMP  5434
000321               BEGIN PACKANDNORM(I);                                        COMP  5435
000324                LATTR.TYPTR := REALPTR                                      COMP  5436
000324               END                                                          COMP  5437
000326              ELSE                                                          COMP  5438
000326               IF COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN                  COMP  5439
000334               BEGIN PACKANDNORM(J);                                        COMP  5440
000336                GATTR.TYPTR := REALPTR                                      COMP  5441
000336               END;                                                         COMP  5442
000341              IF COMPTYPES(LATTR.TYPTR,REALPTR,FALSE)                       COMP  5443
000343               AND COMPTYPES(GATTR.TYPTR,REALPTR,FALSE) THEN                COMP  5444
000356               BEGIN OPERATION(35B,K,I,J); GEN15(24B,K,0,K) END             COMP  5445
000356              ELSE                                                          COMP  5446
000357               IF (LATTR.TYPTR↑.FORM = POWER)                               COMP  5447
000372                AND COMPTYPES(LATTR.TYPTR,GATTR.TYPTR,FALSE) THEN           COMP  5448
000403                OPERATION(15B,K,I,J)                                        COMP  5449
000405               ELSE BEGIN ERROR(134); GATTR.TYPTR := NIL END                COMP  5450
000415             END;                                                           COMP  5451
000415            END;                                                            COMP  5452
000416      (*OR*)OROP:                                                           COMP  5453
000416            IF COMPTYPES(LATTR.TYPTR,BOOLPTR,FALSE)AND                      COMP  5454
000421             COMPTYPES(GATTR.TYPTR,BOOLPTR,FALSE) THEN                      COMP  5455
000431             BOOLOP(LATTR,OROP)                                             COMP  5456
000432            ELSE BEGIN ERROR(134); GATTR.TYPTR := NIL END                   COMP  5457
000442          END (*CASE*)                                                      COMP  5458
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000446          ELSE GATTR.TYPTR := NIL;                                    COMP     5459
000450          IF LOP <> OROP THEN                                         COMP     5460
000452            WITH GATTR DO                                             COMP     5461
000452              BEGIN KIND := EXPR; EXPREG := K END                     COMP     5462
000456        END (*WHILE*)                                                 COMP     5463
000456      END (*SIMPLEEXPRESSION*) ;                                      COMP     5464
000463                                                                      COMP     5465
000463      BEGIN (*EXPRESSION*)                                            COMP     5466
000463        SIMPLEEXPRESSION(FSYS+[RELOP]);                               COMP     5467
000011        IF SY = RELOP THEN                                            COMP     5468
000013          BEGIN                                                       COMP     5469
000013            LATTR := GATTR; LOP := OP;                                COMP     5470
000021          INSYMBOL; SIMPLEEXPRESSION(FSYS);                           COMP     5471
000024          IF LATTR.TYPTR <> NIL THEN                                  COMP     5472
000026            WITH LATTR.TYPTR↑ DO                                      COMP     5473
000027              IF (FORM = ARRAYS) AND (SIZE.WORDS > 1) AND COMPTYPES   UPDATE21   24
000033                (LATTR.TYPTR,GATTR.TYPTR,FALSE) THEN                  UPDATE21   25
000041                BEGIN  LATTR.CWDISPL := LATTR.CWDISPL+SIZE.WORDS-1 ;  UPDATE5    26
000044                  GATTR.CWDISPL := GATTR.CWDISPL+SIZE.WORDS-1 ;       UPDATE5    27
000047                    LOADADDRESS( LATTR,II ) ;  LOADADDRESS( GATTR,JJ ) ;  UPDATE5  28
000055                  END                                                 UPDATE5    29
000055              ELSE                                                    COMP     5476
000056                BEGIN LOAD(LATTR,I); ROTATEX(I);                      COMP     5477
000064                  LOAD(GATTR,J); ROTATEX(J)                           COMP     5478
000071                END;                                                  COMP     5479
000073                K := 0;                                               COMP     5480
000074          IF (LATTR.TYPTR <> NIL)AND (GATTR.TYPTR <> NIL) THEN        COMP     5481
000102        IF LOP = INOP THEN                                            COMP     5482
000104          IF GATTR.TYPTR↑.FORM = POWER THEN                           COMP     5483
000106            IF COMPTYPES(LATTR.TYPTR,GATTR.TYPTR↑.ELSET,FALSE) THEN   COMP     5484
000112              BEGIN NEEDB(L); GEN15(63B,L,I,0);                       COMP     5485
000120                DECREFX(I); DECREFX(J); NEEDX(0,7,K);                 COMP     5486
000132                GEN15(23B,K,L,J);                                     COMP     5487
000136                GEN15(20B,K,0,59); BRGS[L].BCONT := FREE;             COMP     5488
000150                WITH GATTR DO                                         COMP     5489
000150                  BEGIN TYPTR := BOOLPTR; CONDCD := PL END;           COMP     5490
000152              END                                                     COMP     5491
000152            ELSE BEGIN ERROR(129); GATTR.TYPTR := NIL END             COMP     5492
000156          ELSE BEGIN ERROR(130); GATTR.TYPTR := NIL END               COMP     5493
000163        ELSE                                                          COMP     5494
000164          BEGIN                                                       COMP     5495
000164            IF NOT COMPTYPES(LATTR.TYPTR,GATTR.TYPTR,FALSE) THEN      COMP     5496
000170              IF COMPTYPES(LATTR.TYPTR,INTPTR,FALSE) THEN             COMP     5497
000175                BEGIN PACKANDNORM(I);                                 COMP     5498
000177                  LATTR.TYPTR := REALPTR                              COMP     5499
000177                END                                                   COMP     5500
000201              ELSE                                                    COMP     5501
000201                IF COMPTYPES(GATTR.TYPTR,INTPTR,FALSE) THEN           COMP     5502
000207                  BEGIN PACKANDNORM(J);                               COMP     5503
000211                    GATTR.TYPTR := REALPTR                            COMP     5504
000211                  END;                                                COMP     5505
000214            IF COMPTYPES(LATTR.TYPTR,GATTR.TYPTR,FALSE) THEN          COMP     5506
000222              BEGIN                                                   COMP     5507
000222                CASE LATTR.TYPTR↑.FORM OF                             COMP     5508
000224                  SCALAR,                                             COMP     5509
000224                  SUBRANGE:                                           COMP     5510
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000224                ;                                                          COMP    5511
000225               POINTER:                                                    COMP    5512
000225                BEGIN                                                      COMP    5513
000225                 IF LOP IN [LTOP,LEOP,GTOP,GEOP] THEN ERROR(131);          COMP    5514
000231                END;                                                       COMP    5515
000232               POWER:                                                      COMP    5516
000232                BEGIN IF LOP IN [LTOP,GTOP] THEN ERROR(132);               COMP    5517
000236               END;                                                        COMP    5518
000237               ARRAYS:                                                     COMP    5519
000237                IF NOT STRING( LATTR.TYPTR ) THEN ERROR( 134 ) ;           UPDATE6   12
000246               RECORDS: ERROR( 134 ) ;                                     UPDATE6   13
000251               FILES:                                                      COMP    5522
000251                ERROR(133)                                                 COMP    5523
000252              END;                                                         COMP    5524
000263             CASE LATTR.TYPTR↑.FORM OF                                     COMP    5525
000266              SCALAR,                                                      COMP    5526
000266              SUBRANGE,                                                    COMP    5527
000266              POINTER:                                                     COMP    5528
000266               BEGIN                                                       COMP    5529
000266                IF COMPTYPES(LATTR.TYPTR,REALPTR,FALSE) THEN               COMP    5530
000273                 LOPCD := 31B                                              COMP    5531
000273                ELSE                                                       COMP    5532
000274                 LOPCD := 37B;                                             COMP    5533
000275                CASE LOP OF                                                COMP    5534
000277                 LTOP,GEOP: OPERATION(LOPCD,K,I,J);                        COMP    5535
000304                 LEOP,GTOP: OPERATION(LOPCD,K,J,I);                        COMP    5536
000312                 NEOP,EQOP: OPERATION(37B,K,I,J)                           COMP    5537
000314                END;                                                       COMP    5538
000325                WITH GATTR DO                                              COMP    5539
000325                 BEGIN TYPTR := BOOLPTR;                                   COMP    5540
000330                  CASE LOP OF                                              COMP    5541
000331                   LTOP,GTOP: CONDCD := PL;                                COMP    5542
000333                   LEOP,GEOP: CONDCD := NG;                                COMP    5543
000335                      NEOP: CONDCD := ZR;                                  COMP    5544
000337                      EQOP: CONDCD := NZ                                   COMP    5545
000337                  END                                                      COMP    5546
000347                END                                                        COMP    5547
000347             END ;                                                         COMP    5548
000350            POWER:                                                         COMP    5549
000350             BEGIN                                                         COMP    5550
000350              CASE LOP OF                                                  COMP    5551
000352               LTOP,GTOP: ;                                                COMP    5552
000353               LEOP: OPERATION(15B,K,I,J);                                 COMP    5553
000360               GEOP: OPERATION(15B,K,J,I);                                 COMP    5554
000365               NEOP,EQOP: OPERATION(13B,K,I,J)                             COMP    5555
000367              END;                                                         COMP    5556
000400              WITH GATTR DO                                                COMP    5557
000400               BEGIN TYPTR := BOOLPTR;                                     COMP    5558
000403                IF LOP = NEOP THEN CONDCD := ZR ELSE CONDCD := NZ          COMP    5559
000406               END                                                         COMP    5560
000407             END;                                                          COMP    5561
000410            ARRAYS:                                                        COMP    5562
000410             BEGIN                                                         COMP    5563
000410              WITH LATTR.TYPTR↑.SIZE DO                                    COMP    5564
000411               BEGIN WRDS := WORDS; BTS := BITS END;                       COMP    5565
000414              IF WRDS = 0 THEN (*PART WORD COMPARISON*)                    COMP    5566
```

MOORE BUSINESS FORMS, INC. F FORM 1413 PRINTED IN U.S.A. 22

```
000415              BEGIN                                                          COMP     5567
000415               NEEDX(0,7,L); GEN15(43B,L,0,BTS);                             COMP     5568
000424               DECREFX(I); NEEDX(0,7,II); GEN15(11B,II,L,I);                 COMP     5569
000437               DECREFX(J); NEEDX(0,7,JJ); GEN15(11B,JJ,L,J);                 COMP     5570
000452               DECREFX(L);.                                                  COMP     5571
000455               GEN15(20B,II,0,BTS); GEN15(20B,JJ,0,BTS);                     COMP     5572
000465               IF LOP IN [LEOP,GTOP] THEN OPERATION(37B,K,JJ,II)             COMP     5573
000472               ELSE OPERATION(37B,K,II,JJ);                                  COMP     5574
000500              END                                                            COMP     5575
000500             ELSE                                                            COMP     5576
000501              IF BTS <> 0 THEN ERROR( 398 )                                  UPDATE6    14
000503              ELSE                                                          COMP     5578
000505               BEGIN                                                        COMP     5579
000505                IF WRDS > 1 THEN                                            COMP     5580
000506                 BEGIN                                                      COMP     5581
000506                  NEEDB( L ) ;  GEN30( 61B,L,0,1-WRDS,0 ) ;                 UPDATE5    30
000516                  NOOP; LADDR := IC;                                        COMP     5583
000522                  NEEDX(1,5,I); NEEDX(1,5,J);                               COMP     5584
000531                  ARGS[I].ACONT := UNSPECADDR;                              COMP     5585
000537                  ARGS[J].ACONT := UNSPECADDR;                              COMP     5586
000542                  GEN15(53B,I,II,L); GEN15(53B,J,JJ,L);                     COMP     5587
000552                 END;                                                       COMP     5588
000552                 IF LOP IN [GTOP,GEOP] THEN                                 COMP     5589
000555                  BEGIN K := I; I := J; J := K END;                         COMP     5590
000557                 NEEDX( 0,7,K ) ;                                           UPDATE5    31
000563                 IF LOP IN [ LTOP,LEOP,GEOP,GTOP ] THEN                     UPDATE5    32
000566                  BEGIN GEN15( 17B,K,I,J ); DECREFX( I ); NEEDX( 0,7,M );   UPDATE5    33
000601                   IF ( LOP IN [ LEOP,GEOP ] ) AND ( WRDS = 1 ) THEN        UPDATE5    34
000606                    BEGIN  GEN15( 37B,M,J,I ) ;  NEEDX( 0,7,I ) ;           UPDATE5    35
000616                     GEN15( 15B,I,K,M ) ;                                   UPDATE5    36
000622                    END                                                     UPDATE5    37
000622                   ELSE                                                     UPDATE5    38
000623                    BEGIN  GEN15( 37B,M,I,J ) ;  NEEDX( 0,7,I ) ;           UPDATE5    39
000634                     GEN15( 11B,I,K,M ) ;                                   UPDATE5    40
000640                    END ;                                                   UPDATE5    41
000640                   NEEDX( 0,7,N ); GEN15(15B,N,J,K); GEN15(12B,K,I,N);      UPDATE9    59
000654                   DECREFX( N ) ;  DECREFX( M ) ;                           UPDATE9    60
000662                  END                                                       UPDATE5    43
000662                 ELSE  GEN15( 37B,K,I,J ) ;                                 UPDATE5    44
000670                 DECREFX( I ) ;  DECREFX( J ) ;                             UPDATE5    45
000676                 IF WRDS > 1 THEN                                           UPDATE5    46
000700                  BEGIN  GEN15( 66B,L,L,1 ) ;                               UPDATE5    47
000703                   IF LOP IN [ LTOP,LEOP,GEOP,GTOP ] THEN                   UPDATE5    48
000706                    BEGIN GEN30(03B,3,K,IC+2,2); GEN30(03B,1,M,IC+1,2);     UPDATE5    49
000717                    END                                                     UPDATE5    50
000717                   ELSE  GEN30( 03B,1,K,IC+1,2 ) ;                          UPDATE8    14
000725                   GEN30( 06B,0,L,LADDR,2 ) ;                               UPDATE5    54
000731                   IF LOP IN [ LEOP,GEOP ] THEN GEN15( 14B,K,0,K ) ;        UPDATE5    55
000737                   NOOP ;                                                   UPDATE5    56
000741                  DECREFX(II); DECREFX(JJ);                                 COMP     5620
000747                  BRGS[L].BCONT := FREE                                     COMP     5621
000750                 END                                                        COMP     5622
000755               END;                                                         COMP     5623
000755              WITH GATTR DO                                                 COMP     5624
000755               BEGIN TYPTR := BOOLPTR;                                      COMP     5625
000760                CASE LOP OF                                                 COMP     5626
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000761                LTOP,GTOP: CONDCD := PL;                                  COMP    5627
000763                LEOP,GEOP: IF WRDS = 0 THEN CONDCD := NG                  COMP    5628
000765                      ELSE CONDCD := PL;                                  COMP    5629
000770                   NEOP: CONDCD := ZR;                                    COMP    5630
000772                   EQOP: CONDCD := NZ                                     COMP    5631
000772                 END                                                      COMP    5632
001002               END                                                        COMP    5633
001002             END;                                                         COMP    5634
001003            RECORDS,                                                      COMP    5635
001003            FILES:                                                        COMP    5636
001003           END (*CASE*) ;                                                 COMP    5637
001013            END                                                           COMP    5638
001013           ELSE ERRCR(129);                                               COMP    5639
001016          END (*SY <> INOP*)                                              COMP    5640
001016         ELSE GATTR.TYPTR := NIL;                                         COMP    5641
001020        WITH GATTR DO                                                     COMP    5642
001020         BEGIN KIND := COND; CDR := K END;                                COMP    5643
001023       END (*SY = RELOP*) ;                                               COMP    5644
001023     END (*EXPRESSION*) ;                                                 COMP    5645
001027                                                                          COMP    5646
000003     PROCEDURE GENFJMP(FADDR: ADDRRANGE);                                 COMP    5647
000004      (*GENERATE A FALSE JUMP TO FADDR (ON GATTR)*)                       COMP    5648
000004      VAR I: REGNR;                                                       COMP    5649
000005     BEGIN                                                                COMP    5650
000005      WITH GATTR DO                                                       COMP    5651
000007       IF TYPTR <> NIL THEN                                               COMP    5652
000012        IF KIND = COND THEN                                               COMP    5653
000013         BEGIN GEN30(03B,ORD(CONDCD),CDR,FADDR,2); DECREFX(CDR) END       COMP    5654
000022        ELSE                                                              COMP    5655
000023         BEGIN LOAD(GATTR,I); GEN30(03B,0,I,FADDR,2); DECREFX(I) END      COMP    5656
000034     END (*GENFJMP*) ;                                                    COMP    5657
000040                                                                          COMP    5658
000003     PROCEDURE ASSIGNMENT(FCP: CTP);                                      COMP    5659
000004      VAR  LATTR : ATTR ;                                                 UPDATE7  176
000016     BEGIN SELECTOR(FSYS+[BECOMES],FCP);                                  COMP    5662
000012      IF SY = BECOMES THEN                                                COMP    5663
000014       BEGIN                                                              COMP    5664
000014        LATTR := GATTR;                                                   COMP    5665
000021        INSYMBOL; EXPRESSION(FSYS);                                       COMP    5666
000025        ASSIGNTO( LATTR ) ;                                               UPDATE7  177
000027       END (*SY = BECOMES*)                                               COMP    5735
000027      ELSE ERROR(51)                                                      COMP    5736
000031     END (*ASSIGNMENT*) ;                                                 COMP    5737
000037                                                                          COMP    5738
000003     PROCEDURE GOTOSTATEMENT;                                             COMP    5739
000003      LABEL 1;                                                            COMP    5740
000003      VAR LLP: LBP; I: LEVRANGE; LFSTOCC: LOCOFREF;                       COMP    5741
000006     BEGIN CLEARREGS;                                                     COMP    5742
000010      IF SY = INTCONST THEN                                               COMP    5743
000012       BEGIN LLP := FSTLABP;                                              COMP    5744
000013        WHILE LLP <> FLABP DO (*DECIDE WHETHER LOCALLY DECLARED*)         COMP    5745
000017         WITH LLP^ DO                                                     COMP    5746
000020          IF LABVAL = IVAL THEN                                           COMP    5747
000022           BEGIN                                                          COMP    5748
000022            IF DEFINED THEN GEN30(04B,0,0,LABADDR,2)                      COMP    5749
000026            ELSE                                                          COMP    5750
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 47

```
000030          BEGIN GEN30(04B,0,0,0,2); LFSTOCC := FSTOCC;                    COMP  5751
000035           LINKOCC(LFSTOCC); FSTOCC := LFSTOCC                            COMP  5752
000040          END;                                                            COMP  5753
000042         GOTO 1                                                           COMP  5754
000043        END                                                               COMP  5755
000043       ELSE LLP := NEXTLAB;                                              COMP  5756
000046      WHILE LLP <> NIL DO (*DECIDE WHETHER GLOBALLY DECLARED*)           COMP  5757
000050       WITH LLP^ DO                                                      COMP  5758
000051        IF LABVAL = IVAL THEN                                            COMP  5759
000053         BEGIN                                                           COMP  5760
000053          IF LCNT = 0 THEN                                               COMP  5761
000055           IF LABCNT = 36 THEN ERROR(258)                                COMP  5762
000057           ELSE                                                          COMP  5763
000061            BEGIN LABCNT := LABCNT + 1;                                  COMP  5764
000062             LCNT := LABCNT                                              COMP  5765
000062            END;                                                         COMP  5766
000064          IF LEVEL - LABLEV = 1 THEN GEN15(66B,6,5,0)                    COMP  5767
000072          ELSE                                                           COMP  5768
000074           BEGIN GEN15(56B,1,5,0);                                       COMP  5769
000100            FOR I := LABLEV TO LEVEL - 3 DO GEN15(53B,1,1,0);            COMP  5770
000114            GEN15(63B,6,1,0)                                             COMP  5771
000116           END;                                                          COMP  5772
000120          IF LABLEV = 1 THEN GEN15(66B,5,2,0)                            COMP  5773
000124          ELSE                                                           COMP  5774
000127           BEGIN                                                         COMP  5775
000127            IF LEVEL - LABLEV = 1 THEN GEN15(56B,1,5,0)                  COMP  5776
000133            ELSE GEN15(533,1,1,0);                                       COMP  5777
000141            GEN15(63B,5,1,0)                                             COMP  5778
000143             END;                                                        COMP  5779
000145            PASCL[7] := CHR(LCNT);                                       COMP  5780
000151            SEARCHEXTID(PASCL,EXT); GEN30(04B,0,0,0,0);                  COMP  5781
000160            GOTO 1                                                       COMP  5782
000161           END                                                           COMP  5783
000161          ELSE LLP := NEXTLAB;                                           COMP  5784
000164         ERROR(167);                                                     COMP  5785
000166  1:     INSYMBOL                                                        COMP  5786
000166        END                                                              COMP  5787
000167       ELSE ERROR(15)                                                    COMP  5788
000171      END (*GOTOSTATEMENT*) ;                                            COMP  5789
000176                                                                         COMP  5790
000003      PROCEDURE COMPOUNDSTATEMENT;                                       COMP  5791
000003      BEGIN                                                              COMP  5792
000003       (*LOOP UNTIL SY <> SEMICOLON:*)                                   COMP  5793
000003       REPEAT STATEMENT(FSYS+[SEMICOLON,ENDSY]);                         COMP  5794
000012        IF SY IN STATBEGSYS THEN ERROR(14);                              COMP  5795
000016       UNTIL NOT (SY IN STATBEGSYS);                                     COMP  5796
000021       WHILE SY = SEMICOLON DO                                           COMP  5797
000023        BEGIN INSYMBOL;                                                  COMP  5798
000024         REPEAT STATEMENT(FSYS+[SEMICOLON,ENDSY]);                       COMP  5799
000030          IF SY IN STATBEGSYS THEN ERROR(14)                             COMP  5800
000033         UNTIL NOT (SY IN STATBEGSYS)                                    COMP  5801
000036        END;                                                             COMP  5802
000037       IF SY = ENDSY THEN INSYMBOL ELSE ERROR(13)                        COMP  5803
000044      END (*COMPOUNDSTATEMENET*) ;                                       COMP  5804
000052                                                                         COMP  5805
000003      PROCEDURE IFSTATEMENT;                                             COMP  5806
```

MOORE BUSINESS FORMS, INC. FORM 3013 PRINTED IN U.S.A. 22

```
000003      VAR LCIX1,LCIX2: CODERANGE; LCP1,LCP2: POSRANGE; I: REGNR;          COMP  5807
000010        LARGS: ARGSTATUS; LXRGS: XRGSTATUS; LBRGS: BRGSTATUS;             COMP  5808
000050        LBRG: BASREGS; LLEVELS: SET OF LEVRANGE;                          COMP  5809
000054     BEGIN EXPRESSION(FSYS+[THENSY]);                                     COMP  5810
000011      IF NOT COMPTYPES(GATTR.TYPTR,BOOLPTR,FALSE) THEN ERROR(144);        COMP  5811
000021      GENFJMP(0);                                                         COMP  5812
000023      LCIX1 := CIX; LCP1 := CP;                                           COMP  5813
000027      LARGS := ARGS; LXRGS := XRGS; LBRGS := BRGS;                        COMP  5814
000043      LBRG := BRG; LLEVELS := LEVELS;                                     COMP  5815
000051      IF SY = THENSY THEN INSYMBOL ELSE ERROR(52);                        COMP  5816
000057      STATEMENT(FSYS+[ELSESY]);                                           COMP  5817
000063      IF SY = ELSESY THEN                                                 COMP  5818
000065       BEGIN GEN30(04B,0,0,0,2); LCIX2 := CIX; LCP2 := CP;                COMP  5819
000075        NOOP; INS(IC,LCIX1,LCP1);                                         COMP  5820
000103        ARGS := LARGS; XRGS := LXRGS; BRGS := LBRGS;                      COMP  5821
000120        BRG := LBRG; LEVELS := LLEVELS;                                   COMP  5822
000126        INSYMBOL; STATEMENT(FSYS);                                        COMP  5823
000132        NOOP; INS(IC,LCIX2,LCP2)                                          COMP  5824
000137       END                                                                COMP  5825
000141      ELSE                                                                COMP  5826
000142       BEGIN NOOP; INS(IC,LCIX1,LCP1) END;                                COMP  5827
000151      CLEARREGS                                                           COMP  5828
000151     END (*IFSTATEMENT*) ;                                                COMP  5829
000161                                                                          COMP  5830
000003     PROCEDURE CASESTATEMENT;                                             COMP  5831
000003      LABEL 1;                                                            COMP  5832
000003      TYPE CIP = ↑CASEREC;                                                COMP  5833
000003        CASEREC = PACKED                                                  COMP  5834
000003             RECORD NEXT: CIP;                                            COMP  5835
000003              CSLAB: INTEGER;                                             COMP  5836
000003              CSADDR: ADDRRANGE                                           COMP  5837
000003             END;                                                         COMP  5838
000003      VAR LSP,LSP1: STP; FSTPTR,LPT1,LPT2,LPT3: CIP; LVAL: VALU;          COMP  5839
000013        FSTLOCP: LOCOFREF; LCIX,LCIX1,LCIX2: CODERANGE;                   COMP  5840
000017        LCP,LCP1,LCP2: POSRANGE; LDEBUG,EXITLOOP: BOOLEAN;                COMP  5841
000024        I,J,K: REGNR; LARGS: ARGSTATUS; LBRGS: BRGSTATUS;                 COMP  5842
000047        LXRGS: XRGSTATUS;                                                 COMP  5843
000067        LBRG: BASREGS; LLEVELS: SET OF LEVRANGE;                          COMP  5844
000103        LADDR: ADDRRANGE;                                                 COMP  5845
000104        LMIN,LMAX: INTEGER;                                               COMP  5846
000106     BEGIN                                                                COMP  5847
000106      EXPRESSION(FSYS+[OFSY,COMMA,COLON]);                                COMP  5848
000011      LSP := GATTR.TYPTR; FSTLOCP := NIL;                                 COMP  5849
000015      IF LSP <> NIL THEN                                                  COMP  5850
000016       IF (LSP↑.FORM > SUBRANGE)OR COMPTYPES(LSP,REALPTR,FALSE) THEN      COMP  5851
000025        BEGIN ERROR(144); LSP := NIL END;                                 COMP  5852
000027      LOAD(GATTR,I);                                                      COMP  5853
000033      IF DEBUG THEN                                                       COMP  5854
000034       BEGIN GEN30(51B,0,0,IC,0); NEEDX(0,7,J);                           COMP  5855
000044        GEN30(71B,J,0,0,0); LCIX1 := CIX; LCP1 := CP;                     COMP  5856
000054        GEN15(37B,J,I,J); NEEDX(0,7,K);                                   COMP  5857
000064        GEN30(71B,K,0,0,0); LCIX2 := CIX; LCP2 := CP;                     COMP  5858
000074        GEN15(37B,K,K,I); GEN15(12B,K,K,J);                               COMP  5859
000104        GEN30(03B,3,K,INXERR,0); DECREFX(J); DECREFX(K)                   COMP  5860
000114       END;                                                               COMP  5861
000116      LDEBUG := DEBUG;                                                    COMP  5862
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000120      NEEDB(J); GEN15(63B,J,I,0);                                                    COMP    5863
000126      DECREFX(I); GEN30(02B,J,0,0,2); LCIX := CIX; LCP := CP;                        COMP    5864
000141      BRGS[J].BCONT := FREE; LARGS := ARGS; LXRGS := XRGS; LBRGS := BRGS;            COMP    5865
000161      LBRG := BRG; LLEVELS := LEVELS;                                                COMP    5866
000157      IF SY = OFSY THEN INSYMBOL ELSE ERROR(8);                                      COMP    5867
000175      FSTPTR := NIL; LPT3 := NIL;                                                    COMP    5868
000177      (*LOOP UNTIL SY <> SEMICOLON*)                                                 COMP    5869
000177      REPEAT                                                                         COMP    5870
000177       IF NOT (SY IN [SEMICOLON,ENDSY]) THEN                                         COMP    5871
000202       BEGIN NOOP;                                                                   COMP    5872
000204       (*LOOP UNTIL SY <> COMMA:*)                                                   COMP    5873
000204       REPEAT CONSTANT(FSYS+[COMMA,COLON],LSP1,LVAL);                                COMP    5874
000211        IF LSP1 <> NIL THEN                                                          COMP    5875
000213         IF COMPTYPES(LSP,LSP1,FALSE) THEN                                           COMP    5876
000221          BEGIN LPT1 := FSTPTR; LPT2 := NIL;                                         COMP    5877
000223           WHILE LPT1 <> NIL DO                                                      COMP    5878
000225            WITH LPT1^ DO                                                            COMP    5879
000226             BEGIN                                                                   COMP    5880
000226              IF CSLAB <= LVAL.IVAL THEN                                             COMP    5881
000230               BEGIN IF CSLAB = LVAL.IVAL THEN ERROR(156);                           COMP    5882
000232                GOTO 1                                                               COMP    5883
000233               END;                                                                  COMP    5884
000233              LPT2 := LPT1; LPT1 := NEXT                                             COMP    5885
000234             END;                                                                    COMP    5886
000236    1:       NEW(LPT3);                                                             COMP    5887
000240            WITH LPT3^ DO                                                          COMP    5888
000242             BEGIN NEXT := LPT1; CSLAB := LVAL.IVAL;                               COMP    5889
000244              CSADDR := IC                                                         COMP    5890
000244               END;                                                                COMP    5891
000247              IF LPT2 = NIL THEN FSTPTR := LPT3                                    COMP    5892
000250              ELSE LPT2^.NEXT := LPT3                                              COMP    5893
000251             END                                                                   COMP    5894
000252           ELSE ERROR(147);                                                        COMP    5895
000255         EXITLOOP := SY <> COMMA;                                                  COMP    5896
000260          IF NOT EXITLOOP THEN INSYMBOL                                            COMP    5897
000261         UNTIL EXITLOOP;                                                           COMP    5898
000263         IF SY = COLON THEN INSYMBOL ELSE ERROR(5);                                COMP    5899
000271         REPEAT STATEMENT(FSYS+[SEMICOLON]);                                       COMP    5900
000275          IF SY IN STATBEGSYS THEN ERROR(14);                                      COMP    5901
000301         UNTIL NOT (SY IN STATBEGSYS);                                             COMP    5902
000304         GEN30(04B,0,0,0,2); LINKOCC(FSTLOCP);                                     COMP    5903
000312         END (*SY <> ENDSY*) ;                                                     COMP    5904
000312         EXITLOOP := SY <> SEMICOLON;                                              COMP    5905
000315         IF NOT EXITLOOP THEN                                                      COMP    5906
000316         BEGIN ARGS := LARGS; XRGS := LXRGS; BRGS := LBRGS;                        COMP    5907
000333          BRG := LBRG; LEVELS := LLEVELS;                                          COMP    5908
000341          INSYMBOL                                                                 COMP    5909
000341         END;                                                                      COMP    5910
000342       UNTIL EXITLOOP;                                                             COMP    5911
000343       IF FSTPTR <> NIL THEN                                                       COMP    5912
000345        BEGIN LMAX := FSTPTR^.CSLAB;                                               COMP    5913
000346         IF LDEBUG THEN INS(LMAX,LCIX2,LCP2);                                      COMP    5914
000353         (*REVERSE POINTERS*)                                                      COMP    5915
000353         LPT1 := FSTPTR; FSTPTR := NIL;                                            COMP    5916
000355         REPEAT LPT2 := LPT1^.NEXT; LPT1^.NEXT := FSTPTR;                          COMP    5917
000357          FSTPTR := LPT1; LPT1 := LPT2                                             COMP    5918
```

MOORE BUSINESS FORMS, INC. FORM 213 PRINTED IN U.S.A. 22

```
000360          UNTIL LPT1 = NIL;                                                                  COMP     5919
000362          LMIN := FSTPTR^.CSLAB;                                                             COMP     5920
000363          IF LDEBUG THEN INS(LMIN,LCIX1,LCP1);                                               COMP     5921
000370          NOOP; INS(IC-LMIN,LCIX,LCP);                                                       COMP     5922
000377          IF LMAX - LMIN < CODEMAX THEN                                                      COMP     5923
000402           BEGIN LADDR := IC + LMAX - LMIN + 1;                                              COMP     5924
000404            REPEAT                                                                           COMP     5925
000404             WITH FSTPTR^ DO                                                                 COMP     5926
000406              BEGIN                                                                          COMP     5927
000406               WHILE CSLAB > LMIN DO                                                         COMP     5928
000410                BEGIN                                                                        COMP     5929
000410                 IF LDEBUG THEN                                                              COMP     5930
000411                  BEGIN GEN30(51B,0,0,IC,0); GEN30(04B,0,0,INXERR,0)                         COMP     5931
000417                  END                                                                        COMP     5932
000421                 ELSE                                                                        COMP     5933
000422                  BEGIN GEN30(04B,0,0,LADDR,2); NOOP END;                                    COMP     5934
000430                LMIN := LMIN + 1                                                             COMP     5935
000430               END;                                                                          COMP     5936
000432              GEN30(04B,0,0,CSADDR,2); NOOP;                                                 COMP     5937
000441              FSTPTR := NEXT; LMIN := LMIN + 1                                               COMP     5938
000443             END                                                                             COMP     5939
000444            UNTIL FSTPTR = NIL;                                                              COMP     5940
000445            WHILE FSTLOCP <> NIL DO                                                          COMP     5941
000447             WITH FSTLOCP^ DO                                                                COMP     5942
000450              BEGIN INS(LADDR,LOC,POS); FSTLOCP := NXTREF                                    COMP     5943
000455              END                                                                            COMP     5944
000457          END                                                                                COMP     5945
000460           ELSE ERROR(157)                                                                   COMP     5946
000462          END (*FSTPTR <> NIL*)                                                              COMP     5947
000463         ELSE ERROR(6);                                                                      COMP     5948
000466         CLEARREGS;                                                                          COMP     5949
000470          IF SY = ENDSY THEN INSYMBOL ELSE ERROR(13)                                         COMP     5950
000475        END (*CASESTATEMENT*) ;                                                             COMP     5951
000505                                                                                            COMP     5952
000003        PROCEDURE REPEATSTATEMENT;                                                          COMP     5953
000003         VAR LADDR: ADDRRANGE; I: REGNR;                                                    COMP     5954
000005        BEGIN CLEARREGS;                                                                    COMP     5955
000010         NOOP; LADDR := IC;                                                                 COMP     5956
000014         (*LOOP UNTIL SY <> SEMICOLON:*)                                                    COMP     5957
000014         REPEAT STATEMENT(FSYS+[SEMICOLON,UNTILSY]);                                        COMP     5958
000020          IF SY IN STATBEGSYS THEN ERROR(14)                                                COMP     5959
000023         UNTIL NOT (SY IN STATBEGSYS);                                                      COMP     5960
000027         WHILE SY = SEMICOLON DO                                                            COMP     5961
000031          BEGIN INSYMBOL;                                                                   COMP     5962
000032           REPEAT STATEMENT(FSYS+[SEMICOLON,UNTILSY]);                                      UPDATE21   27
000036            IF SY IN STATBEGSYS THEN ERROR(14);                                             UPDATE21   28
000042           UNTIL NOT (SY IN STATBEGSYS);                                                    COMP     5964
000045          END;                                                                              COMP     5965
000045         IF SY = UNTILSY THEN                                                               COMP     5966
000047          BEGIN INSYMBOL; EXPRESSION(FSYS);                                                 COMP     5967
000053           IF NOT COMPTYPES(GATTR.TYPTR,BOOLPTR,FALSE) THEN ERROR(144);                     COMP     5968
000063           GENFJMP(LADDR)                                                                   COMP     5969
000064          END                                                                               COMP     5970
000066         ELSE ERROR(53)                                                                     COMP     5971
000070        END (*REPEATSTATEMENT*) ;                                                           COMP     5972
000076                                                                                            COMP     5973
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000003      PROCEDURE WHILESTATEMENT;                                                  COMP     5974
000003       VAR LADDR: ADDRRANGE; LCIX: CODERANGE; LCP: POSRANGE; I: REGNR;           COMP     5975
000007         LARGS: ARGSTATUS; LXRGS: XRGSTATUS; LBRGS: BRGSTATUS;                   COMP     5976
000047         LBRG: BASREGS; LLEVELS: SET OF LEVRANGE;                                COMP     5977
000063      BEGIN CLEARREGS;                                                           COMP     5978
000010       NOOP; LADDR := IC;                                                        COMP     5979
000014       EXPRESSION(FSYS+[DOSY]);                                                  COMP     5980
000017       IF NOT COMPTYPES(GATTR.TYPTR,BOOLPTR,FALSE) THEN ERROR(144);              COMP     5981
000027       GENFJMP(0);                                                               COMP     5982
000031       LCIX := CIX; LCP := CP;                                                   COMP     5983
000035       LARGS := ARGS; LXRGS := XRGS; LBRGS := BRGS;                              COMP     5984
000050       LBRG := BRG; LLEVELS := LEVELS;                                           COMP     5985
000056       IF SY = DOSY THEN INSYMBOL ELSE ERROR(54);                                COMP     5986
000064       STATEMENT(FSYS);                                                          COMP     5987
000067       GEN30(04B,0,0,LADDR,2); NOOP; INS(IC,LCIX,LCP);                           COMP     5988
000101       ARGS := LARGS; XRGS := LXRGS; BRGS := LBRGS;                              COMP     5989
000116       BRG := LBRG; LEVELS := LLEVELS;                                           COMP     5990
000124      END (*WHILESTATEMENT*) ;                                                   COMP     5991
000131                                                                                 COMP     5992
000003      PROCEDURE FORSTATEMENT;                                                    COMP     5993
000003       VAR LATTR,LIMIT: ATTR; LSP: STP; LSY: SYMBOL; LADDR: ADDRRANGE;           COMP     5994
000032         LCIX: CODERANGE; LPOS: POSRANGE; I,J,K: REGNR; LCP: CTP;                COMP     5995
000040         LMIN,LMAX: INTEGER;                                                     COMP     5996
000042      BEGIN  (* DEFINE LATTR TO PREVENT BLOW UP IN CASE OF ERRORS. *)            UPDATE6    19
000042       WITH LATTR DO                                                             UPDATE6    20
000006        BEGIN  TYPTR := NIL ;  KIND := VARBL ;  WORDACC := DRCT ;                UPDATE6    21
000010        VLEVEL := LEVEL ;  CWDISPL := 0 ;  PCKD := FALSE ;                       UPDATE6    22
000012       END ;                                                                     UPDATE6    23
000012       IF SY = IDENT THEN                                                        COMP     5998
000014        BEGIN SEARCHID([VARS],LCP);                                              COMP     5999
000016         WITH LCP↑, LATTR DO                                                     COMP     6000
000020          BEGIN  TYPTR := IDTYPE ;                                               UPDATE6    24
000021           IF VKIND = DRCT THEN                                                  COMP     6002
000024            IF VLEV IN [ 1,LEVEL ] THEN                                          UPDATE6    25
000030             BEGIN  VLEVEL := VLEV ;  CWDISPL := VADDR ;  END                    UPDATE6    26
000032            ELSE ERRCR( 155 )                                                    UPDATE6    27
000033           ELSE ERROR(155)                                                       COMP     6006
000036          END;                                                                   COMP     6007
000037         IF LATTR.TYPTR <> NIL THEN                                              COMP     6008
000041          IF (LATTR.TYPTR↑.FORM > SUBRANGE)                                      COMP     6009
000043           OR COMPTYPES(REALPTR,LATTR.TYPTR,FALSE) THEN                          COMP     6010
000050           BEGIN ERROR(143); LATTR.TYPTR := NIL END;                             COMP     6011
000052         INSYMBOL                                                                COMP     6012
000052        END                                                                      COMP     6013
000053       ELSE                                                                      COMP     6014
000054        BEGIN ERROR(2); SKIP(FSYS+[BECOMES,TOSY,DOWNTOSY,DOSY]) END;             COMP     6015
000061       I := 0;                                                                   COMP     6016
000062       IF SY = BECOMES THEN                                                      COMP     6017
000064        BEGIN INSYMBOL; EXPRESSION(FSYS+[TOSY,DOWNTOSY,DOSY]);                   COMP     6018
000070         IF GATTR.TYPTR <> NIL THEN                                              COMP     6019
000073          IF GATTR.TYPTR↑.FORM > SUBRANGE THEN ERROR(144)                        COMP     6020
000076          ELSE                                                                   COMP     6021
000100           IF COMPTYPES(LATTR.TYPTR,GATTR.TYPTR,FALSE) THEN                      COMP     6022
000104            BEGIN LOAD(GATTR,I);                                                 COMP     6023
000107             IF I <> 6 THEN                                                      COMP     6024
000111              BEGIN NEEDX(6,6,J); BXIXJ(6,I) END;                               COMP     6025
```

MOORE BUSINESS FORMS, INC., F
FORM 14
PRINTED IN U.S.A.
22

```
000117              WITH XRGS[6] DO                                          COMP  6026
000117               BEGIN XCONT := OTHER; REFNR := 1 END                    COMP  6027
000125             END                                                       COMP  6028
000125            ELSE ERROR(145)                                          COMP  6029
000127          END                                                         COMP  6030
000130         ELSE                                                         COMP  6031
000131          BEGIN ERROR(51); SKIP(FSYS+[TOSY,DOWNTOSY,DOSY]) END;       COMP  6032
000136         LSY := SY; LIMIT.TYPTR := NIL;                               COMP  6033
000141         IF SY IN [TOSY,DOWNTOSY] THEN                                COMP  6034
000143          BEGIN INSYMBOL; EXPRESSION(FSYS+[DOSY]);                    COMP  6035
000147           IF GATTR.TYPTR <> NIL THEN                                 COMP  6036
000152           IF GATTR.TYPTR^.FORM > SUBRANGE THEN ERROR(144)            COMP  6037
000155            ELSE                                                      COMP  6038
000157             IF COMPTYPES(LATTR.TYPTR,GATTR.TYPTR,FALSE) THEN         COMP  6039
000163              BEGIN LIMIT := GATTR; LOAD(GATTR,I);                    COMP  6040
000172               IF (LIMIT.KIND = CST) AND (ABS(LIMIT.CVAL.IVAL) < TWOTO17)  COMP  6041
000175                THEN CLEARREGS                                        COMP  6042
000177               ELSE                                                   COMP  6043
000202                BEGIN                                                 COMP  6044
000202                 IF I IN [0,6,7] THEN                                 COMP  6045
000205                  BEGIN NEEDX(1,5,K); BXIXJ(K,I); I := K END;         COMP  6046
000216                 CLEARREGS;                                           COMP  6047
000220                 WITH XRGS[I] DO                                      COMP  6048
000224                  BEGIN XCONT := SIMPVAR; REFNR := 0; LASTREF := IC;   COMP  6049
000233                   SHFTCNT := 0;                                      COMP  6050
000234                   XLEV := LEVEL; XADDR := LC                         COMP  6051
000236                  END;                                                COMP  6052
000240                 WITH LIMIT DO                                        COMP  6053
000240                  BEGIN KIND := VARBL; WORDACC := DRCT;               COMP  6054
000242                   VLEVEL := LEVEL; CWDISPL := LC; PCKD := FALSE      COMP  6055
000243                  END;                                                COMP  6056
000243                 GEN15(10B,7,I,0); GEN30(51B,7,BRG[LEVEL],LC,0);      COMP  6057
000254                 LC := LC + 1;                                        COMP  6058
000256                 IF LC > LCMAX THEN LCMAX := LC;                      COMP  6059
000250                END;                                                  COMP  6060
000260               WITH XRGS[6] DO                                        COMP  6061
000260                BEGIN XCONT := SIMPVAR; REFNR := 1; LASTREF := IC;     COMP  6062
000271                 SHFTCNT := 0;                                        COMP  6063
000273                 XLEV := LATTR.VLEVEL; XADDR := LATTR.CWDISPL         COMP  6064
000274                END                                                   COMP  6065
000277             END                                                      COMP  6066
000277            ELSE ERROR(145)                                          COMP  6067
000300          END                                                         COMP  6068
000301         ELSE BEGIN ERROR(55); SKIP(FSYS+[DOSY]) END;                 COMP  6069
000307         IF SY = DOSY THEN INSYMBOL ELSE ERROR(54);                   COMP  6070
000315         NOOP; LADDR := IC; NEEDX(0,7,K);                             COMP  6071
000324         IF LSY = TOSY THEN GEN15(37B,K,I,6) ELSE GEN15(37B,K,6,I);   COMP  6072
000341         GEN30(03B,3,K,0,2); LCIX := CIX; LPOS := CP; DECREFX(K);     COMP  6073
000353         IF LATTR.TYPTR <> NIL THEN                                   COMP  6074
000355          BEGIN                                                       COMP  6075
000355           IF DEBUG THEN                                              COMP  6076
000356            IF LATTR.TYPTR <> INTPTR THEN                             COMP  6077
000360             BEGIN GETBOUNDS(LATTR.TYPTR,LMIN,LMAX);                  COMP  6078
000362              CHECKBNDS(6,LMIN,LMAX,ASSERR)                           COMP  6079
000364             END;                                                     COMP  6080
000366           GEN30(51B,6,BRG[LATTR.VLEVEL],LATTR.CWDISPL,0);            COMP  6081
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000374          WITH ARGS[6] DO                                                 COMP    6082
000374           BEGIN ACONT := SIMPADDR; ALEV := LATTR.VLEVEL;                 COMP    6083
000402             AADDR := LATTR.CWDISPL                                       COMP    6084
000402           END                                                            COMP    6085
000404         END;                                                             COMP    6086
000404        DECREFX(6);                                                       COMP    6087
000407        STATEMENT(FSYS);                                                  COMP    6088
000412        LOAD(LATTR,J); NEEDX(0,7,K); GEN15(76B,K,1,0);                    COMP    6089
000425        IF LSY = TOSY THEN GEN15(36B,6,J,K) ELSE GEN15(37B,6,J,K);        COMP    6090
000440        DECREFX(J); DECREFX(K);                                           COMP    6091
000446        IF LIMIT.TYPTR <> NIL THEN                                        COMP    6092
000450         IF LIMIT.KIND = CST THEN                                         COMP    6093
000451           BEGIN NEEDX(I,I,I);                                            COMP    6094
000454            IF LIMIT.CVAL.IVAL = 1 THEN                                   COMP    6095
000456              GEN15(76B,I,1,0)                                            COMP    6096
000460            ELSE GEN30(71B,I,0,LIMIT.CVAL.IVAL,0)                         COMP    6097
000464           END                                                            COMP    6098
000466         ELSE                                                             COMP    6099
000467           BEGIN LOAD(LIMIT,J);                                           COMP    6100
000472            IF J <> I THEN BXIXJ(I,J);                                    COMP    6101
000476            LC := LC - 1                                                  COMP    6102
000476           END;                                                           COMP    6103
000500        GEN30(04B,0,0,LADDR,2); NOOP; INS(IC,LCIX,LPOS);                  COMP    6104
000512        CLEARREGS                                                         COMP    6105
000512      END (*FORSTATEMENT*) ;                                              COMP    6106
000525                                                                          COMP    6107
```

```
000003      PROCEDURE WITHSTATEMENT;                                            COMP    6108
000003       VAR LCP: CTP; OLDTOP: DISPRANGE; LLC: ADDRRANGE; LATTR: ATTR;      COMP    6109
000020         I: REGNR; EXITLOOP: BOOLEAN;                                      COMP    6110
000022      BEGIN OLDTOP := TOP; LLC := LC;                                     COMP    6111
000010       (*LOOP UNTIL SY <> COMMA:*)                                        COMP    6112
000010       REPEAT                                                             COMP    6113
000010         IF SY = IDENT THEN                                               COMP    6114
000012           BEGIN SEARCHID([VARS,FIELD],LCP); INSYMBOL END                 COMP    6115
000015         ELSE BEGIN ERROR(2); LCP := UVARPTR END;                         COMP    6116
000022         SELECTOR(FSYS+[COMMA,DOSY],LCP);                                 COMP    6117
000026         IF GATTR.TYPTR <> NIL THEN                                       COMP    6118
000031           IF GATTR.TYPTR↑.FORM = RECORDS THEN                            COMP    6119
000034             IF TOP < DISPLIMIT THEN                                      COMP    6120
000036               BEGIN TOP := TOP + 1;                                      COMP    6121
000037                 WITH DISPLAY[TOP], GATTR DO                              COMP    6122
000040                   BEGIN FNAME := TYPTR↑.FIELDS; OCCUR := REC;            COMP    6123
000045                     IF WORDACC = DRCT THEN                               COMP    6124
000046                       BEGIN WACC := DRCT;                                COMP    6125
000050                         LEV := VLEVEL; CWDSPL := CWDISPL                 COMP    6126
000052                       END                                                COMP    6127
000054                     ELSE                                                 COMP    6128
000055                       BEGIN                                              COMP    6129
000055                         LOADADDRESS(GATTR,I);                            COMP    6130
000057                         WITH LATTR DO                                    COMP    6131
000057                           BEGIN TYPTR := GATTR.TYPTR; KIND := VARBL;     COMP    6132
000062                             WORDACC := DRCT; VLEVEL := LEVEL;            COMP    6133
000064                             CWDISPL := LC; PCKD := FALSE                 COMP    6134
000065                           END;                                           COMP    6135
000066                         STORE(LATTR,I);                                  COMP    6136
000070                         LEV := LEVEL; CWDSPL := LC;                      COMP    6137
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000076              LC := LC + 1;                                          COMP    6138
000076              WACC := INDRCT                                         COMP    6139
000076             END;                                                    COMP    6140
000100            IF PCKD THEN                                             COMP    6141
000102             BEGIN PKD := TRUE;                                      COMP    6142
000105              IF BITREG = NONE THEN                                  COMP    6143
000107               BEGIN BACC := DRCT; BDSPL := CBDISPL END              COMP    6144
000112              ELSE                                                   COMP    6145
000113               BEGIN                                                 COMP    6146
000113                IF CBDISPL <> 0 THEN                                 COMP    6147
000115                 BEGIN NEEDX(0,7,I);                                 COMP    6148
000120                  IF CBDISPL = 1 THEN GEN15(76B,I,0,1)               COMP    6149
000125                  ELSE GEN30(71B,I,0,CBDISPL,0);                     COMP    6150
000133                  GEN15(36B,I,VBDISPL,I);                            COMP    6151
000137                  DECREFX(VWDISPL)                                   COMP    6152
000141                 END                                                 COMP    6153
000142                ELSE I := VBDISPL;                                   COMP    6154
000144                WITH LATTR DO                                        COMP    6155
000144                 BEGIN TYPTR := GATTR.TYPTR; KIND := VARBL;          COMP    6156
000147                  WORDACC := DRCT; VLEVEL := LEVEL;                  COMP    6157
000151                  CWDISPL := LC; PCKD := FALSE                       COMP    6158
000152                 END;                                                COMP    6159
000153                STORE(LATTR,I);                                      COMP    6160
000155                BACC := INDRCT; BDSPL := LC;                         COMP    6161
000162                LC := LC + 1;                                        COMP    6162
000163               END                                                   COMP    6163
000163             END (*PCKD*)                                            COMP    6164
000163              ELSE PKD := FALSE                                      COMP    6165
000165            END                                                      COMP    6166
000167           END                                                       COMP    6167
000167          ELSE ERROR(250)                                           COMP    6168
000171         ELSE ERROR(140);                                           COMP    6169
000175       EXITLOOP := SY <> COMMA;                                     COMP    6170
000200       IF NOT EXITLOOP THEN INSYMBOL                                COMP    6171
000201       UNTIL EXITLOOP;                                              COMP    6172
000203       IF LC > LCMAX THEN LCMAX := LC;                              COMP    6173
000207       IF SY = DOSY THEN INSYMBOL ELSE ERROR(54);                   COMP    6174
000215       STATEMENT(FSYS);                                             COMP    6175
000220       (*DISPOSE LOCALLY USED X-REGISTERS*)                         COMP    6176
000220       FOR I := 0 TO 7 DO                                           COMP    6177
000221        WITH XRGS[I] DO                                             COMP    6178
000226         IF XCONT = SIMPVAR THEN                                    COMP    6179
000230          IF (XLEV = LEVEL)AND (XADDR >= LLC) THEN XCONT := AVAIL;   COMP    6180
000241       FOR I := 0 TO 7 DO                                           COMP    6181
000242        WITH XRGS[I] DO                                             COMP    6182
000247         IF XCONT = INDVAR THEN                                     COMP    6183
000251          IF XRGS[XREG].XCONT = AVAIL THEN XCONT := AVAIL;           COMP    6184
000261       TOP := OLDTOP; LC := LLC                                     COMP    6185
000263     END (*WITHSTATEMENT*) ;                                        COMP    6186
000271                                                                    COMP    6187
000271    BEGIN (*STATEMENT*)                                             COMP    6188
000271     IF SY = INTCONST THEN (*LABEL*)                                COMP    6189
000010      BEGIN CLEARREGS; NOOP;                                        COMP    6190
000014       IF IVAL = TRAPLAB THEN INS( IC,LCIX2,LCP2 ) ;                UPDAT10   22
000023       LLP := FSTLABP;                                              COMP    6191
000025       WHILE LLP <> FLABP DO                                        COMP    6192
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A.

```
000030          WITH LLP↑ DO                                                    COMP    6193
000031           IF LABVAL = IVAL THEN                                          COMP    6194
000033            BEGIN                                                         COMP    6195
000033             IF DEFINED THEN ERROR(165)                                   COMP    6196
000035             ELSE                                                         COMP    6197
000037              BEGIN LOCP := FSTOCC;                                       COMP    6198
000040               WHILE LOCP <> NIL DO                                       COMP    6199
000042                WITH LOCP↑ DO                                             COMP    6200
000043                 BEGIN INS(IC,LOC,POS); LOCP := NXTREF END;               COMP    6201
000052               DEFINED := TRUE; LABADDR := IC                             COMP    6202
000055              END;                                                        COMP    6203
000057             GOTO 1                                                       COMP    6204
000060            END                                                           COMP    6205
000060           ELSE LLP := NEXTLAB;                                           COMP    6206
000063         ERROR(167);                                                      COMP    6207
000065       1: INSYMBOL;                                                       COMP    6208
000066         IF SY = COLON THEN INSYMBOL ELSE ERROR(5)                        COMP    6209
000073        END;                                                              COMP    6210
000074       IF NOT (SY IN FSYS+[IDENT]) THEN                                   COMP    6211
000077        BEGIN ERROR(6); SKIP(FSYS) END;                                   COMP    6212
000103       IF SY IN STATBEGSYS+[IDENT] THEN                                   COMP    6213
000106        BEGIN                                                             COMP    6214
000106         CASE SY OF                                                       COMP    6215
000107          IDENT:     BEGIN SEARCHID([VARS,FIELD,FUNC,PROC],LCP); INSYMBOL; COMP    6216
000112                      IF LCP↑.KLASS = PROC THEN CALL(FSYS,LCP)            COMP    6217
000116                      ELSE ASSIGNMENT(LCP)                                COMP    6218
000122                     END;                                                 COMP    6219
000124          BEGINSY:   BEGIN INSYMBOL; COMPOUNDSTATEMENT END;               COMP    6220
000130          GOTOSY:    BEGIN INSYMBOL; GOTOSTATEMENT END;                   COMP    6221
000134          IFSY:      BEGIN INSYMBOL; IFSTATEMENT END;                     COMP    6222
000140          CASESY:    BEGIN INSYMBOL; CASESTATEMENT END;                   COMP    6223
000144          WHILESY:   BEGIN INSYMBOL; WHILESTATEMENT END;                  COMP    6224
000150          REPEATSY:  BEGIN INSYMBOL; REPEATSTATEMENT END;                 COMP    6225
000154          FORSY:     BEGIN INSYMBOL; FORSTATEMENT END;                    COMP    6226
000160          WITHSY:    BEGIN INSYMBOL; WITHSTATEMENT END                    COMP    6227
000163         END;                                                            COMP    6228
000232         IF NOT (SY IN FSYS) THEN                                        COMP    6229
000235          BEGIN ERROR(6); SKIP(FSYS) END                                 COMP    6230
000240        END                                                               COMP    6231
000240       END (*STATEMENT*) ;                                               COMP    6232
000244                                                                         COMP    6233
000244      BEGIN (*BODY*)                                                     COMP    6234
000244         DP := FALSE;                                                    COMP    6235
000010        CIX := 0; CP := 4; RCIX := 0; RCP := 15;                         COMP    6236
000014        BONUS[SHRTCST] := 20; BONUS[LONGCST] := 10;                      COMP    6237
000016        BONUS[SIMPVAR] := 4; BONUS[INDVAR] := 3;                         COMP    6238
000017        BRG[LEVEL] := 5; BRG[1] := 2; BRG[0] := 6;                       COMP    6239
000023        IC := 0; CADDR := 0; EXT := 0; EXTIDX := 0; EXTRX := 0;          COMP    6240
000027        CLEARREGS;                                                       COMP    6241
000030        FCLSP := NIL;                                                    COMP    6242
000031       IF LEVEL = 1 THEN                                                 COMP    6243
000033       BEGIN                                                             COMP    6244
000033        IF PMD THEN SEARCHEXTID(=PASCPMD   =,EXT);                       COMP    6245
000036        GEN30(40B,0,0,0,0);                                              COMP    6246
000041        (*DISPL OF OUTPUT EFET[0]*)                                      COMP    6247
000041        GEN30(0,0,0,OUTPUTPTR↑.VADDR+CHEFET,0);                          COMP    6248
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A 22

```
000047    ORDALFA(PROGNAME,LCD); GEN60(LCD); GEN30(0,0,0,IC+1,0);                  COMP    6249
000050    GEN30(0,0,0,0,0); LCIX1 := CIX; LCP1 := CP;                              COMP    6250
000066    GEN30(61B,7,0,EXTFILS,0);                                                COMP    6251
000072    SEARCHEXTID(ΞP.INIT    Ξ,EXT); RJTOEXT;                                  COMP    6252
000077    GEN30(61B,6,5,0,0); LCIX := CIX; LCPT := CP;                             COMP    6253
000105    GEN30(61B,7,6,100,0); GEN30(51B,0,0,IC,0);                               COMP    6254
000114    GEN30(06B,7,4,STOVLERR,0);                                               COMP    6255
000120    (*OPEN GLOBAL FILES:*)                                                   COMP    6256
000120    OPENFILES(DISPLAY[1].FNAME);                                             COMP    6257
000123    (*SEARCH FOR UNDECLARED EXTERNAL FILES*)                                 COMP    6258
000123    EXFILP := FEXFILP;                                                       COMP    6259
000125    WHILE EXFILP <> NIL DO                                                   COMP    6260
000127     WITH EXFILP↑ DO                                                         COMP    6261
000130      BEGIN                                                                  COMP    6262
000130       IF NOT DECLARED AND (FILENAME<>ΞINPUT     Ξ)                         COMP    6263
000132        AND (FILENAME <> ΞOUTPUT    Ξ) THEN                                 COMP    6264
000136        BEGIN  ERROR( 172 ) ;  LINECOUNT := LINECOUNT+1 ;                    SPACERR    4
000141         WRITELN(Ξ FILE-ID Ξ,FILENAME);                                     COMP    6266
000154        END;                                                                COMP    6268
000154       EXFILP := NXTP                                                       COMP    6269
000154      END;                                                                  COMP    6270
000157    (*CALL P.RESET (NO EFFECT IF ACTUAL FILE = INPUT:*)                      COMP    6271
000157    IF INPUTPTR <> NIL THEN                                                  COMP    6272
000161     BEGIN GEN30(51B,1,BRG[1],INPUTPTR↑.VADDR+CHEFET,0);                     COMP    6273
000166      SEARCHEXTID(ΞP.RESET   Ξ,EXT); RJTOEXT;                               COMP    6274
000173     END;                                                                    COMP    6275
000173    (*SET DEFAULT VALUE FOR OUTPUT:*)                                        COMP    6276
000173     GEN30(71B,6,0,1000,0); GEN30(51B,6,BRG[1],OUTPUTPTR↑.VADDR,0)           COMP    6277
000203    END ELSE                                                                 COMP    6278
000205    BEGIN                                                                    COMP    6279
000205     ORDALFA(FPROCP↑.NAME,LCD); GEN60(LCD); GEN30(0,0,0,IC+1,0);             COMP    6280
000216     GEN30(0,0,0,0,0); LCIX1 := CIX; LCP1 := CP;                             COMP    6281
000224     IF LEVEL > 2 THEN GEN15(56B,6,5,0);                                     COMP    6282
000231     GEN15(76B,6,5,0); GEN15(20B,6,2,2); GEN15(12B,7,7,6);                   COMP    6283
000242     GEN15(66B,5,6,0); GEN15(56B,7,5,1);                                     COMP    6284
000250     GEN30(61B,6,6,0,0); LCIX := CIX; LCPT := CP;                            COMP    6285
000255     GEN30(61B,7,6,100,0); GEN30(51B,0,0,IC,0);                              COMP    6286
000264     GEN30(06B,7,4,STOVLERR,0);                                              COMP    6287
000270     LCP := FPROCP↑.NEXT;                                                    COMP    6288
000273     I := 0; LDISP := 3;                                                     COMP    6289
000275     WHILE (LCP <> NIL)AND (I < XPARMAX) DO   (*STORE X-I AT PARAM LOC*)     COMP    6290
000302      BEGIN                                                                  COMP    6291
000302       WITH GATTR DO                                                         COMP    6292
000302        BEGIN TYPTR := INTPTR; KIND := VARBL; WORDACC := DRCT;               COMP    6293
000304         VLEVEL := LEVEL; CWDISPL := LDISP; PCKD := FALSE                    COMP    6294
000306        END;                                                                  COMP    6295
000307       WITH XRGS[I] DO                                                       COMP    6296
000311        BEGIN XCONT := OTHER; REFNR := 1 END;                                COMP    6297
000315       STORE(GATTR,I);                                                       COMP    6298
000317       I := I + 1; LDISP := LDISP + 1; LCP := LCP↑.NEXT                      COMP    6299
000322      END;                                                                   COMP    6300
000324     LOCFILES := FALSE;                                                      COMP    6301
000325     (*OPEN LOCAL FILES:*)                                                   COMP    6302
000325     OPENFILES(DISPLAY[TOP].FNAME);                                          COMP    6303
000331     IF LOCFILES THEN CLEARREGS;   (*ASSUME REGISTER CONTENTS DESTROYED*)    COMP    6304
000334     (*COPY MULTIPLE VALUES INTO LOCAL CELLS:*)                              COMP    6305
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000334     LCP := FPROCP↑.NEXT; LDISP := 3;                                  COMP      6306
000340     WHILE LCP <> NIL DO                                               COMP      6307
000342      WITH LCP↑ DO                                                     COMP      6308
000343       BEGIN                                                           COMP      6309
000343        IF IDTYPE <> NIL THEN                                          COMP      6310
000344         BEGIN LSZ := FULLWORDS(IDTYPE↑.SIZE);                         COMP      6311
000350          IF (KLASS = VARS)AND (VKIND = DRCT)AND (LSZ <> 1) THEN       COMP      6312
000357           BEGIN GEN30(51B,4,5,LDISP,0); GEN30(71B,7,5,VADDR,0);       COMP      6313
000370            NEEDB(I); GEN30(61B,I,0,LSZ-1,0); NOOP;                    COMP      6314
000400            GEN15(53B,5,4,I); GEN15(10B,6,5,0);                        COMP      6315
000407            GEN15(53B,6,7,I); GEN15(67B,I,I,1);                        COMP      6316
000416            GEN30(06B,I,0,IC-1,2); NOOP;                               COMP      6317
000424            BRGS[I].BCONT := FREE;                                     COMP      6318
000430            FOR I := 4 TO 7 DO                                         COMP      6319
000431             BEGIN XRGS[I].XCONT := AVAIL;                             COMP      6320
000437              ARGS[I].ACONT := UNSPECADDR                              COMP      6321
000437             END                                                       COMP      6322
000442           END                                                         COMP      6323
000444         END;                                                          COMP      6324
000444         LDISP := LDISP + 1; LCP := NEXT                               COMP      6325
000446       END;                                                            COMP      6326
000451     END;                                                              COMP      6327
000451     IF TRAPSET THEN                                                   UPDAT10     23
000453      BEGIN  CLEARREGS ;  (*ASSUME REGISTER CONTENTS DESTROYED*)       UPDAT10     24
000454       GEN30( 51B,1,0,57B,0 ) ;  (*TRAP LOCATION*)                     UPDAT10     25
000457       GEN15( 10B,6,1,0 ) ;  GEN30( 51B,6,BRG[ LEVEL ],TRADD,0 ) ;     UPDAT10     26
000467       GEN30( 71B,6,0,0,2 ) ;  LCIX2 := CIX ;  LCP2 := CP ;            UPDAT10     27
000475       GEN15( 76B,1,BRG[ LEVEL ],0 ) ;  GEN15( 20B,1,2,2 ) ;           UPDAT10     28
000504       GEN15( 12B,6,6,1 ) ;  GEN15( 54B,6,1,0 ) ;                      UPDAT10     29
000512      END ;                                                            UPDAT10     30
000512     LCMAX := LC;                                                      COMP      6328
000514     (*LOOP UNTIL SY <> SEMICOLON;*)                                   COMP      6329
000514     REPEAT STATEMENT(FSYS+[SEMICOLON,ENDSY]);                         COMP      6330
000517      IF SY IN STATBEGSYS THEN ERROR(14)                               COMP      6331
000522     UNTIL NOT (SY IN STATBEGSYS);                                     COMP      6332
000526     WHILE SY = SEMICOLON DO                                           COMP      6333
000530      BEGIN INSYMBOL;                                                  COMP      6334
000531       REPEAT STATEMENT(FSYS+[SEMICOLON,ENDSY]);                       COMP      6335
000534        IF SY IN STATBEGSYS THEN ERROR(14)                             COMP      6336
000537       UNTIL NOT (SY IN STATBEGSYS);                                   COMP      6337
000543      END;                                                             COMP      6338
000543     INS(LCMAX,LCIX,LCPT);                                             COMP      6339
000547     IF SY = ENDSY THEN INSYMBOL ELSE ERROR(13);                       COMP      6340
000555     (*CLOSE LOCAL FILES;*)                                            COMP      6341
000555     CLSP := FCLSP;                                                    COMP      6342
000556     IF CLSP <> NIL THEN CLEARREGS ;                                   UPDATE8     15
000561     WHILE CLSP <> NIL DO                                              COMP      6343
000563      WITH CLSP↑ DO                                                    COMP      6344
000564       BEGIN GEN30(51B,1,BRG[LEVEL],EFETADDR,0);                       COMP      6345
000571        SEARCHEXTID(ΞP.CLOSE    Ξ,EXT); RJTOEXT;                       COMP      6346
000575        CLSP := NXTP                                                   COMP      6347
000576       END;                                                            COMP      6348
000601     IF LEVEL = 1 THEN                                                 COMP      6349
000603      BEGIN LGOHEAD(PROGNAME);                                         COMP      6350
000605       (*CLOSE,OUTPUT*)                                                COMP      6351
000605       CLEARREGS ;                                                     UPDATE9     61
```

FORM 1413 PRINTED IN U.S.A. MOORE BUSINESS FORMS, INC. F

```
000607      GEN30(51B,1,BRG[1],OUTPUTPTR↑.VADDR+CHEFET,0);                    COMP    6352
000615      SEARCHEXTID(ΞP.PUTS    Ξ,EXT); RJTOEXT;                           COMP    6353
000622      (*END REQUEST:*)                                                  COMP    6354
000622      GEN30(71B,6,0,051604B,0); GEN15(20B,6,5,2);                       COMP    6355
000631      GEN15(56B,6,1,0); NOOP; GEN30(00B,0,0,0,0)                        COMP    6356
000640     END                                                                COMP    6357
000641    ELSE                                                                COMP    6358
000642     BEGIN                                                              COMP    6359
000642      WITH FPROCP↑ DO                                                   COMP    6360
000644       IF PFCNT = 0 THEN LGOHEAD(NAME)                                  COMP    6361
000647       ELSE                                                             COMP    6362
000652        BEGIN PFNAME(PFCNT); LGOHEAD(PNAME) END;                        COMP    6363
000656      IF FPROCP↑.KLASS = FUNC THEN (*LOAD FUNCTION RESULT:*)            COMP    6364
000662       BEGIN                                                            COMP    6365
000662        WITH GATTR DO                                                   COMP    6366
000662         BEGIN TYPTR := FPROCP↑.IDTYPE; KIND := VARBL; WORDACC := DRCT; COMP    6367
000664          VLEVEL := LEVEL; CWDISPL := 2; PCKD := FALSE                  COMP    6368
000666         END;                                                           COMP    6369
000667        LOAD(GATTR,I);                                                  COMP    6370
000671        IF I <> 6 THEN BXIXJ(6,I)                                       COMP    6371
000673       END;                                                             COMP    6372
000674      CLEARREGS ;                                                       UPDATE9   62
000676      IF TRAPSET THEN                                                   UPDAT10   31
000700       BEGIN  GEN30( 51B,1,BRG[ LEVEL ],TRADD,0 ) ;                     UPDAT10   32
000704        GEN15( 10B,7,1,0 ) ;  GEN30( 51B,7,0,57B,0 ) ; (*TRAP LOCATION*) UPDAT10   33
000713       END ;                                                            UPDAT10   34
000713      GEN15(56B,1,5,1); GEN15(66B,6,5,0); GEN15(63B,7,1,0);             COMP    6373
000724       GEN15(20B,1,5,2); GEN15(63B,5,1,0); GEN30(02B,7,0,0,0)           COMP    6374
000734      END;                                                              COMP    6375
000735     (*APPEND CONSTANTS TO CODE, FIXUP THEIR REFERENCES*)               COMP    6376
000735     LCIX := CIX;                                                       COMP    6377
000736     LCSP := FSTCSP;                                                    COMP    6378
000740     WHILE LCSP <> NIL DO                                               COMP    6379
000742      WITH LCSP↑ DO                                                     COMP    6380
000743       BEGIN LOCP := CREF;                                              COMP    6381
000744        WHILE LOCP <> NIL DO                                            COMP    6382
000746         WITH LOCP↑ DO                                                  COMP    6383
000747          BEGIN INS(IC,LOC,POS);                                        COMP    6384
000753           LOCP := NXTREF                                               COMP    6385
000753          END;                                                          COMP    6386
000756        CREF := NIL;                                                    COMP    6387
000761        LP := CSTP;                                                     COMP    6388
000762        WHILE LP <> NIL DO                                              COMP    6389
000764         WITH LP↑ DO                                                    COMP    6390
000765          BEGIN GEN60(CSVAL); LP := NXTCSP END;                         COMP    6391
000771        LCSP := NXTCSP;                                                 COMP    6392
000773       END;                                                             COMP    6393
000774     FSTCSP := LFSTCSP;                                                 COMP    6394
000776     INS(LCIX*100000B+CIX,LCIX1,LCP1);                                  COMP    6395
001002     IF PMD THEN PMDINFO(DISPLAY[LEVEL].FNAME);                         COMP    6396
001007     GEN60(0);                                                          COMP    6397
001011     LGOTEXT; LGOEND;                                                   COMP    6398
001015    END (*BODY*) ;                                                      COMP    6399
001032                                                                        COMP    6400
001032   BEGIN (*BLOCK*)                                                      COMP    6401
001032    FLABP := FSTLABP;  LFSTCSP := FSTCSP;                               COMP    6402
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000012   TRAPSET := FALSE ;                                                  UPDAT10    35
000013   REPEAT                                                              COMP     6403
000013    IF SY = LABELSY THEN                                               COMP     6404
000015     BEGIN INSYMBOL; LABELDECLARATION END;                             COMP     6405
000020    IF SY = CONSTSY THEN                                               COMP     6406
000022     BEGIN INSYMBOL; CONSTDECLARATION END;                             COMP     6407
000025    IF SY = TYPESY THEN                                                COMP     6408
000027     BEGIN INSYMBOL; TYPEDECLARATION END;                              COMP     6409
000032    IF SY = VARSY THEN                                                 COMP     6410
000034     BEGIN INSYMBOL; VARDECLARATION END;                               COMP     6411
000037    WHILE SY IN [PROCEDURESY,FUNCTIONSY] DO                            COMP     6412
000042     BEGIN LSY := SY; INSYMBOL; PROCEDUREDECLARATION(LSY) END;         COMP     6413
000047    IF SY <> BEGINSY THEN                                              COMP     6414
000051     BEGIN ERROR(18); SKIP(FSYS) END                                   COMP     6415
000054   UNTIL SY IN STATBEGSYS;                                             COMP     6416
000057   IF SY = BEGINSY THEN INSYMBOL ELSE ERROR(17);                       COMP     6417
000064   REPEAT BODY(FSYS+[CASESY]);                                         COMP     6418
000067    IF SY <> FSY THEN                                                  COMP     6419
000071     BEGIN ERROR(6); SKIP(FSYS) END                                    COMP     6420
000075   UNTIL (SY = FSY)OR (SY IN BLOCKBEGSYS);                             COMP     6421
000101 END (*BLOCK*) ;                                                       COMP     6422
000107                                                                       COMP     6423
000107                                                                       COMP     6424
000003 PROCEDURE PROGRAMME(FSYS: SETOFSYS);                                  COMP     6425
000004  (*SHOULD PREFERABLY SURROUND PROC BLOCK WHICH WAS NOT POSSIBLE*)      COMP     6426
000004  (*BEFORE BOOTSTRAP DUE TO COMPILER RESTRICTION ON PROC NESTING*)     COMP     6427
000004  LABEL 1;                                                            COMP     6428
000004  VAR EXFILP: EXTFILEP; LCP: CTP;                                     COMP     6429
000006 BEGIN PROGNAME := ΞP.MAIN    Ξ; FEXFILP := NIL; EXTFILS := 0;        COMP     6430
000012  (*DECLARE OUTPUT WITH FIX ADDRESS B2 + 6*)                          COMP     6431
000012  NEW(LCP,VARS);                                                      COMP     6432
000014  WITH LCP↑ DO                                                        COMP     6433
000015   BEGIN NAME := ΞOUTPUT    Ξ; IDTYPE := TEXTPTR;                     COMP     6434
000020    KLASS := VARS; VKIND := DRCT; NEXT := NIL;                        COMP     6435
000027    VLEV := 1; VADDR := LC                                            COMP     6436
000031   END;                                                               COMP     6437
000034  ENTERID(LCP);                                                       COMP     6438
000035  LC := LC + TEXTPTR↑.SIZE.WORDS;                                     COMP     6439
000040  IF SY = PROGRAMSY THEN                                              COMP     6440
000042   BEGIN INSYMBOL;                                                    COMP     6441
000043    IF SY = IDENT THEN                                                COMP     6442
000045      BEGIN  PROGNAME := ID ;  LISTNAME := ID ;  INSYMBOL ;           LISTCTRL   51
000050      IF NOT (SY IN [SEMICOLON,LPARENT]) THEN                         COMP     6444
000053       BEGIN ERROR(7); SKIP(FSYS+[SEMICOLON,LPARENT]) END;            COMP     6445
000057      IF SY = LPARENT THEN                                            COMP     6446
000061       BEGIN                                                          COMP     6447
000061        REPEAT INSYMBOL;                                              COMP     6448
000062         IF SY = IDENT THEN                                           COMP     6449
000064          BEGIN                                                       COMP     6450
000064           IF ID = ΞINPUT     Ξ THEN                                  COMP     6451
000066            BEGIN NEW(INPUTPTR,VARS);                                 COMP     6452
000067             WITH INPUTPTR↑ DO                                        COMP     6453
000071              BEGIN NAME := ΞINPUT     Ξ; IDTYPE := TEXTPTR;          COMP     6454
000074               KLASS := VARS; VKIND := DRCT; NEXT := NIL;             COMP     6455
000103               VLEV := 1; VADDR := LC                                 COMP     6456
000105              END;                                                    COMP     6457
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000107                ENTERID(INPUTPTR);                                                COMP     6458
000110                LC := LC + TEXTPTR↑.SIZE.WORDS;                                   COMP     6459
000113               END                                                                COMP     6460
000113              ELSE                                                                COMP     6461
000113               IF ID = ΞOUTPUT    Ξ THEN OUTPUTPTR := LCP;                        COMP     6462
000116              EXTFILS := EXTFILS + 1;                                            COMP     6463
000120              EXFILP := FEXFILP;                                                 COMP     6464
000121              WHILE EXFILP <> NIL DO                                             COMP     6465
000123               WITH EXFILP↑ DO                                                   COMP     6466
000124                BEGIN                                                            COMP     6467
000124                 IF FILENAME = ID THEN                                           COMP     6468
000126                  BEGIN ERROR(101); GOTO 1 END;                                  COMP     6469
000130                 EXFILP := NXTP                                                  COMP     6470
000130                END;                                                             COMP     6471
000133  1:            NEW(EXFILP);                                                      COMP     6472
000135                WITH EXFILP↑ DO                                                   COMP     6473
000137                 BEGIN FILENAME := ID; NXTP := FEXFILP;                          COMP     6474
000142                  DECLARED := FALSE;                                             COMP     6475
000144                  INSYMBOL;                                                      COMP     6476
000145                  IF OP = MUL THEN                                               UPDATE9    63
000147                   BEGIN READONLY := TRUE; INSYMBOL END                          COMP     6478
000153                  ELSE READONLY := FALSE;                                        COMP     6479
000157                  IF FILENAME = ΞINPUT     Ξ THEN READONLY := TRUE ;             UPDATE9    64
000164                  IF FILENAME = ΞOUTPUT    Ξ THEN READONLY := FALSE ;            UPDATE9    65
000171                  IF OP = PLUS THEN                                              UPDATE9    66
000173                   BEGIN IF FILENAME = ΞINPUT     Ξ THEN                         UPDATE9    67
000176                           INPUTPTR↑.IDTYPE := STEXTPTR                          UPDATE9    68
000176                         ELSE IF FILENAME = ΞOUTPUT    Ξ THEN                   UPDATE9    69
000203                           OUTPUTPTR↑.IDTYPE := STEXTPTR                         UPDATE9    70
000203                         ELSE ERROR( 6 ) ;                                       UPDATE9    71
000210                  INSYMBOL ;                                                     UPDATE9    72
000211                END ;                                                            UPDATE9    73
000211               SYSLOC := EXTFILS + 1                                             COMP     6480
000212              END;                                                               COMP     6481
000215             FEXFILP := EXFILP                                                   COMP     6482
000215            END                                                                  COMP     6483
000216           ELSE ERROR(2);                                                       COMP     6484
000221           IF NOT (SY IN [COMMA,RPARENT]) THEN                                  COMP     6485
000224            BEGIN ERROR(6); SKIP(FSYS+[COMMA,RPARENT]) END                      COMP     6486
000230          UNTIL SY <> COMMA;                                                   COMP     6487
000232          IF SY = RPARENT THEN INSYMBOL ELSE ERROR(4)                          COMP     6488
000237         END;                                                                  COMP     6489
000240        IF SY = SEMICOLON THEN INSYMBOL ELSE ERROR(14)                         COMP     6490
000244       END                                                                     COMP     6491
000245      ELSE BEGIN ERROR(2); SKIP(FSYS) END                                      COMP     6492
000252     END                                                                       COMP     6493
000252    ELSE BEGIN ERROR(3); SKIP(FSYS) END;                                       COMP     6494
000257    IF OUTPUTPTR = NIL THEN                                                    COMP     6495
000261     BEGIN ERROR(176); OUTPUTPTR := LCP END;                                   COMP     6496
000265    REPEAT BLOCK(FSYS,PERIOD,NIL)                                              COMP     6497
000267    UNTIL SY = PERIOD                                                          COMP     6498
000270  END (*PROGRAMME*) ;                                                          COMP     6499
000301                                                                               COMP     6500
000003  PROCEDURE STANDARDNAMES;                                                     COMP     6501
000003  BEGIN                                                                        COMP     6502
000003   NA[ 1] := ΞFALSE     Ξ;NA[ 2] := ΞTRUE      Ξ;NA[ 3] := ΞOUTPUT    Ξ;     COMP     6503
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000011   NA[ 4] := ΞINPUT     Ξ;NA[ 5] := ΞGET       Ξ;NA[ 6] := ΞPUT       Ξ;  COMP  6504
000014   NA[ 7] := ΞRESET     Ξ;NA[ 8] := ΞREWRITE   Ξ;NA[ 9] := ΞGETSEG    Ξ;  COMP  6505
000017   NA[10] := ΞPUTSEG    Ξ;NA[11] := ΞLINELIMIT Ξ;NA[12] := ΞREAD      Ξ;  COMP  6506
000022   NA[13] := ΞREADLN    Ξ;NA[14] := ΞWRITE     Ξ;NA[15] := ΞWRITELN   Ξ;  COMP  6507
000025   NA[16] := ΞMESSAGE   Ξ;NA[17] := ΞPAGE      Ξ;NA[18] := ΞTIME      Ξ;  COMP  6508
000030   NA[19] := ΞDATE      Ξ;NA[20] := ΞHALT      Ξ;NA[21] := ΞPACK      Ξ;  COMP  6509
000033   NA[22] := ΞUNPACK    Ξ;NA[23] := ΞNEW       Ξ;NA[24] := ΞDISPOSE   Ξ;  COMP  6510
000036   NA[25] := ΞRELEASE   Ξ;NA[26] := Ξ*DUMMY2   Ξ;NA[27] := Ξ*DUMMY3   Ξ;  COMP  6511
000041   NA[28] := ΞEOF       Ξ;NA[29] := ΞEOS       Ξ;NA[30] := ΞEOLN      Ξ;  COMP  6512
000044   NA[31] := ΞODD       Ξ;NA[32] := ΞUNDEFINED Ξ;NA[33] := ΞROUND     Ξ;  COMP  6513
000047   NA[34] := ΞTRUNC     Ξ;NA[35] := ΞEXPO      Ξ;NA[36] := ΞABS       Ξ;  COMP  6514
000052   NA[37] := ΞSQR       Ξ;NA[38] := ΞORD       Ξ;NA[39] := ΞCHR       Ξ;  COMP  6515
000055   NA[40] := ΞPRED      Ξ;NA[41] := ΞSUCC      Ξ;NA[42] := ΞCARD      Ξ;  COMP  6516
000060   NA[43] := ΞCLOCK     Ξ;NA[44] := Ξ*DUMMY4   Ξ;NA[45] := Ξ*DUMMY5   Ξ;  COMP  6517
000063   NA[46] := ΞSIN       Ξ;NA[47] := ΞCOS       Ξ;NA[48] := ΞEXP       Ξ;  COMP  6518
000066   NA[49] := ΞSQRT      Ξ;NA[50] := ΞLN        Ξ;NA[51] := ΞARCTAN    Ξ;  COMP  6519
000071  END (*STANDARDNAMES*) ;                                                  COMP  6520
000160                                                                          COMP  6521
000003  PROCEDURE STDTYPENTRIES;                                                COMP  6522
000003   VAR SP: STP;                                                           COMP  6523
000004  BEGIN                                                (*TYPE*)          COMP  6524
000004                                                       (******)          COMP  6525
000004                                                                         COMP  6526
000004   NEW(INTPTR,SCALAR,STANDARD);                         (*INTEGER*)      COMP  6527
000007   WITH INTPTR↑ DO                                                       COMP  6528
000011    BEGIN FORM := SCALAR; SCALKIND := STANDARD; FTYPE := FALSE;          COMP  6529
000017     WITH SIZE DO                                                        COMP  6530
000017      BEGIN WORDS := 1; BITS := 0 END                                    COMP  6531
000023    END;                                                                 COMP  6532
000023   NEW(REALPTR,SCALAR,STANDARD);                        (*REAL*)         COMP  6533
000025   WITH REALPTR↑ DO                                                      COMP  6534
000027    BEGIN FORM := SCALAR; SCALKIND := STANDARD; FTYPE := FALSE;          COMP  6535
000035     WITH SIZE DO                                                        COMP  6536
000035      BEGIN WORDS := 1; BITS := 0 END                                    COMP  6537
000041    END;                                                                 COMP  6538
000041   NEW(CHARPTR,SCALAR,STANDARD);                        (*CHAR*)         COMP  6539
000043   WITH CHARPTR↑ DO                                                      COMP  6540
000045    BEGIN FORM := SCALAR; SCALKIND := STANDARD; FTYPE := FALSE;          COMP  6541
000053     WITH SIZE DO                                                        COMP  6542
000053      BEGIN WORDS := 0; BITS := 6 END                                    COMP  6543
000060    END;                                                                 COMP  6544
000060   NEW(BOOLPTR,SCALAR,DECLARED);                        (*BOOLEAN*)      COMP  6545
000061   WITH BOOLPTR↑ DO                                                      COMP  6546
000063    BEGIN FORM := SCALAR; SCALKIND := DECLARED; FTYPE := FALSE;          COMP  6547
000071     WITH SIZE DO                                                        COMP  6548
000072      BEGIN WORDS := 0; BITS := 1 END;                                   COMP  6549
000076    END;                                                                 COMP  6550
000076   NEW(NILPTR,POINTER);                                 (*NIL*)          COMP  6551
000077   WITH NILPTR↑ DO                                                       COMP  6552
000101    BEGIN ELTYPE := NIL; FORM := POINTER; FTYPE := FALSE;                COMP  6553
000107    WITH SIZE DO                                                         COMP  6554
000110     BEGIN WORDS := 0; BITS := 18 END                                    COMP  6555
000115    END;                                                                 COMP  6556
000115   NEW(TEXTPTR,FILES);                                  (*TEXT*)         COMP  6557
000116   WITH TEXTPTR↑ DO                                                      COMP  6558
000120    BEGIN FILTYPE := CHARPTR; PCKDFIL := TRUE; FORM := FILES;            COMP  6559
```

MOORE BUSINESS FORMS, INC. FORM 141 PRINTED IN U.S.A.

```
000127     TEXTFILE := TRUE; SEGFILE := FALSE; FTYPE := TRUE;                          COMP     6560
000134     WITH SIZE DO                                                                COMP     6561
000135      BEGIN WORDS := 128*BUFFAC + 1 + CHEFETSZ; BITS := 0 END                    COMP     6562
000143    END;                                                                         COMP     6563
000143   NEW( STEXTPTR,FILES ) ;                                                       UPDATE9    74
000145   WITH STEXTPTR↑ DO                                                             UPDATE9    75
000147    BEGIN  FILTYPE := CHARPTR ;  PCKDFIL := TRUE ;  FORM := FILES ;             UPDATE9    76
000156     TEXTFILE := TRUE ;  SEGFILE := TRUE ;  FTYPE := TRUE ;                     UPDATE9    77
000163     WITH SIZE DO                                                              UPDATE9    78
000164      BEGIN  WORDS := 128*BUFFAC+1+CHEFETSZ ;  BITS := 0 ;  END ;               UPDATE9    79
000172    END ;                                                                      UPDATE9    80
000172   NEW(SP,SUBRANGE);                                          (*ALFA*)        COMP     6564
000174   WITH SP↑ DO                                                                 COMP     6565
000175    BEGIN FORM := SUBRANGE; RANGETYPE := INTPTR; FTYPE := FALSE;               COMP     6566
000203     MIN.IVAL := 1; MAX.IVAL := ALFALENG;                                      COMP     6567
000205     WITH SIZE DO                                                              COMP     6568
000206      BEGIN WORDS := 0; BITS := 4 END;                                         COMP     6569
000213    END;                                                                       COMP     6570
000213   NEW(ALFAPTR,ARRAYS);                                                        COMP     6571
000214   WITH ALFAPTR↑ DO                                                            COMP     6572
000216    BEGIN FORM := ARRAYS; FTYPE := FALSE;                                      COMP     6573
000222     AELTYPE := CHARPTR; INXTYPE := SP;                                        COMP     6574
000227     PCKDARR := TRUE; PARTWORDELS := TRUE; ELSPERWORD := ALFALENG;             COMP     6575
000235     WITH SIZE DO                                                              COMP     6576
000236      BEGIN WORDS := 1; BITS := 0 END                                          COMP     6577
000242    END                                                                        COMP     6578
000242  END (*STDTYPENTRIES*);                                                       COMP     6579
000246                                                                               COMP     6580
000003  PROCEDURE STDNAMENTRIES;                                                     COMP     6581
000003   VAR CP,CP1: CTP; I,LCNT: INTEGER;                                           COMP     6582
000007  BEGIN                                                  (*NAME:*)            COMP     6583
000007                                                         (*******)            COMP     6584
000007                                                                              COMP     6585
000007   NEW(CP,TYPES);                                        (*INTEGER*)          COMP     6586
000007   WITH CP↑ DO                                                                COMP     6587
000011    BEGIN NAME := ΞINTEGER    Ξ; IDTYPE := INTPTR; KLASS := TYPES END;        COMP     6588
000017   ENTERID(CP);                                                               COMP     6589
000020   NEW(CP,TYPES);                                        (*REAL*)             COMP     6590
000022   WITH CP↑ DO                                                                COMP     6591
000024    BEGIN NAME := ΞREAL       Ξ; IDTYPE := REALPTR; KLASS := TYPES END;       COMP     6592
000032   ENTERID(CP);                                                               COMP     6593
000033   NEW(CP,TYPES);                                        (*CHAR*)             COMP     6594
000035   WITH CP↑ DO                                                                COMP     6595
000037    BEGIN NAME := ΞCHAR       Ξ; IDTYPE := CHARPTR; KLASS := TYPES END;       COMP     6596
000045   ENTERID(CP);                                                               COMP     6597
000046   NEW(CP,TYPES);                                        (*BOOLEAN*)          COMP     6598
000050   WITH CP↑ DO                                                                COMP     6599
000052    BEGIN NAME := ΞBOOLEAN    Ξ; IDTYPE := BOOLPTR; KLASS := TYPES END;       COMP     6600
000060   ENTERID(CP);                                                               COMP     6601
000061   NEW(CP,TYPES);                                        (*TEXT*)             COMP     6602
000063   WITH CP↑ DO                                                                COMP     6603
000065    BEGIN NAME := ΞTEXT       Ξ; IDTYPE := TEXTPTR; KLASS := TYPES END;       COMP     6604
000073   ENTERID(CP);                                                               COMP     6605
000074   NEW(CP,TYPES);                                        (*ALFA*)             COMP     6606
000076   WITH CP↑ DO                                                                COMP     6607
000100    BEGIN NAME := ΞALFA       Ξ; IDTYPE := ALFAPTR; KLASS := TYPES END;       COMP     6608
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000106   ENTERID(CP);                                                          COMP     6609
000107   NEW(CP,KONST);                                        (*NIL*)         COMP     6610
000111   WITH CP↑ DO                                                           COMP     6611
000113    BEGIN NAME := ΞNIL       Ξ; IDTYPE := NILPTR; KLASS := KONST;        COMP     6612
000121     NEXT := NIL; VALUES.IVAL := 377777B;                                COMP     6613
000124    END;                                                                 COMP     6614
000124   ENTERID(CP);                                                          COMP     6615
000125   NEW(CP,KONST);                                        (*COLON*)       COMP     6616
000127   WITH CP↑ DO                                                           COMP     6617
000131    BEGIN NAME := ΞCOL       Ξ; IDTYPE := CHARPTR; KLASS := KONST;       COMP     6618
000137     NEXT := NIL; VALUES.IVAL := 0                                       COMP     6619
000141    END;                                                                 COMP     6620
000142   ENTERID(CP);                                                          COMP     6621
000143   NEW(CP,KONST);                                        (*PERCENT*)     PERCENTA    1
000145   WITH CP↑ DO                                                           PERCENTA    2
000147    BEGIN  NAME := ΞPER       Ξ; IDTYPE := CHARPTR; KLASS := KONST;      PERCENTA    3
000155     NEXT := NIL; VALUES.IVAL := 633;                                    PERCENTA    4
000160    END;                                                                 PERCENTA    5
000160   ENTERID(CP);                                                          PERCENTA    6
000161   NEW(CP,KONST);                                        (*MAXINT*)      COMP     6622
000163   WITH CP↑ DO                                                           COMP     6623
000165    BEGIN NAME := ΞMAXINT    Ξ; IDTYPE := INTPTR; KLASS := KONST;        COMP     6624
000173     NEXT := NIL; VALUES.IVAL := 7777777777777777B;                      COMP     6625
000176    END;                                                                 COMP     6626
000176   ENTERID(CP);                                                          COMP     6627
000177   CP1 := NIL; LCNT := 0;                                                COMP     6628
000201   FOR I := 1 TO 2 DO                                                    COMP     6629
000202    BEGIN NEW(CP,KONST);                                 (*FALSE,TRUE*)  COMP     6630
000205     WITH CP↑ DO                                                         COMP     6631
000207      BEGIN NAME := NA[LCNT+I]; IDTYPE := BOOLPTR; KLASS := KONST;       COMP     6632
000216       NEXT := CP1; VALUES.IVAL := I - 1                                 COMP     6633
000220      END;                                                               COMP     6634
000220     ENTERID(CP); CP1 := CP                                              COMP     6635
000221    END;                                                                 COMP     6636
000224   LCNT := LCNT + 4;                                                     COMP     6637
000226   BOOLPTR↑.FCONST := CP;                                                COMP     6638
000231   FOR I := 1 TO NRSTDPROC DO                                            COMP     6639
000231    BEGIN NEW(CP,PROC,STANDARD);              (*GET,PUT,RESET*)         COMP     6640
000234     WITH CP↑ DO                              (*REWRITE,GETSEG*)        COMP     6641
000236      BEGIN NAME := NA[LCNT+I]; IDTYPE := NIL; (*PUTSEG,LINELIMIT*)      COMP     6642
000242       NEXT := NIL; KEY := I;                 (*READ,READLN*)           COMP     6643
000246       KLASS := PROC; PFDECKIND := STANDARD   (*WRITE,WRITELN*)         COMP     6644
000250      END;                                    (*MESSAGE,PAGE*)          COMP     6645
000252     ENTERID(CP)                              (*TIME,DATE,HALT*)        COMP     6646
000252    END;                                      (*PACK,UNPACK,NEW*)       COMP     6647
000255   LCNT := LCNT + NRSTDPROC;                  (*DISPOSE,RELEASE*)       COMP     6648
000257   FOR I := 1 TO NRSTDFUNC DO                 (*EOF,EOS,EOLN,ODD*)      COMP     6649
000260    BEGIN NEW(CP,FUNC,STANDARD);              (*UNDEFINED,ROUND*)       COMP     6650
000263     WITH CP↑ DO                              (*TRUNC,EXPO,ABS*)        COMP     6651
000265      BEGIN NAME := NA[LCNT+I]; IDTYPE := NIL; (*SQR,ORD,CHR,PRED*)      COMP     6652
000271       KLASS := FUNC; PFDECKIND := STANDARD;  (*SUCC,CARD,CLOCK*)       COMP     6653
000276        NEXT := NIL; KEY := I;                                           COMP     6654
000302     END;                                                                COMP     6655
000302     ENTERID(CP)                                                         COMP     6656
000302    END;                                                                 COMP     6657
000305   LCNT := LCNT + NRSTDFUNC;                                             COMP     6658
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000307   FOR I := 1 TO NREXFUNC DO                          (*SIN,COS,EXP*)
000310    BEGIN NEW(CP,FUNC,STANDARD);                      (*SQRT,LN,ARCTAN*)
000313     WITH CP↑ DO
000315      BEGIN NAME := NA[LCNT+I]; IDTYPE := REALPTR;
000321       KLASS := FUNC; PFDECKIND := STANDARD;
000326       NEXT := NIL; KEY := NRSTDFUNC + I
000330      END;
000332     ENTERID(CP)
000332    END;
000335  END (*STDNAMENTRIES*);
000354
000003  PROCEDURE ENTERUNDECL;
000003  BEGIN
000003   NEW(UTYPPTR,TYPES);
000007   WITH UTYPPTR↑ DO
000011    BEGIN NAME := Ξ          Ξ; IDTYPE := NIL; KLASS := TYPES END;
000017   NEW(UCSTPTR,KONST);
000020   WITH UCSTPTR↑ DO
000022    BEGIN NAME := Ξ          Ξ; IDTYPE := NIL; NEXT := NIL;
000027     VALUES.IVAL := 0; KLASS := KONST;
000032    END;
000032   NEW(UVARPTR,VARS);
000034   WITH UVARPTR↑ DO
000036    BEGIN NAME := Ξ          Ξ; IDTYPE := NIL; VKIND := DRCT;
000044     NEXT := NIL; VLEV := 0; VADDR := 0; KLASS := VARS
000051    END;
000053   NEW(UFLDPTR,FIELD);
000054   WITH UFLDPTR↑ DO
000056    BEGIN NAME := Ξ          Ξ; IDTYPE := NIL; NEXT := NIL; FLDADDR := 0;
000065     KLASS := FIELD
000065    END;
000070   NEW(UPRCPTR,PROC,DECLARED,ACTUAL);
000071   WITH UPRCPTR↑ DO
000073    BEGIN NAME := Ξ          Ξ; IDTYPE := NIL;
000076     KLASS := PROC; PFDECKIND := DECLARED; PFKIND := ACTUAL;
000105     NEXT := NIL; PFDECL := DECL; PFLEV := 0; PFADDR := 0
000112    END;
000114   NEW(UFCTPTR,FUNC,DECLARED,ACTUAL);
000115   WITH UFCTPTR↑ DO
000117    BEGIN NAME := Ξ          Ξ; IDTYPE := NIL; NEXT := NIL;
000124     KLASS := FUNC; PFDECKIND := DECLARED; PFKIND := ACTUAL;
000133     PFDECL := DECL; PFLEV := 0; PFADDR := 0
000136    END
000140  END (*ENTERUNDECL*) ;
000145
000003  PROCEDURE INITSCALARS;
000003  BEGIN FWPTR := NIL; FSTLABP := NIL; FSTCSP := NIL; FILECNT := 0;
000012   INPUTPTR := NIL; OUTPUTPTR := NIL;
000014   LABCNT := 0; ERRORS := FALSE; B6DPL := 3;
000016   DEBUG := TRUE; EXTON := FALSE; LISTON := TRUE; PMD := TRUE;
000021   RUNLINKAGE := FALSE;
000021   DP := TRUE; PRTERR := TRUE; ERRINX := 0;
000023   KK := ALFALENG; XPARMAX := 4 ;
000025   BUFFAC := 1 ;  (*WILL NOT ALLOW INPUT AND OUTPUT TO BE TAPE FILES*)
000025                  (*ANY MORE, BUT WILL REDUCE PROGRAM SIZE BY 1400B*)
000025   IC := 0; CODEADDR := 0; PCNT := 0;
```

```
COMP      6659
COMP      6660
COMP      6661
COMP      6662
COMP      6663
COMP      6664
COMP      6665
COMP      6666
COMP      6667
COMP      6668
COMP      6669
COMP      6670
COMP      6671
COMP      6672
COMP      6673
COMP      6674
COMP      6675
COMP      6676
COMP      6677
COMP      6678
COMP      6679
COMP      6680
COMP      6681
COMP      6682
COMP      6683
COMP      6684
COMP      6685
COMP      6686
COMP      6687
COMP      6688
COMP      6689
COMP      6690
COMP      6691
COMP      6692
COMP      6693
COMP      6694
COMP      6695
COMP      6696
COMP      6697
COMP      6698
COMP      6699
COMP      6700
COMP      6701
COMP      6702
COMP      6703
COMP      6704
COMP      6705
COMP      6706
COMP      6707
COMP      6708
RUNMODS     37
COMP      6709
UPDATE22    27
UPDATE22    28
UPDATE22    29
COMP      6711
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22

```
000030   LC := 6; (*NEVER TOUCH THIS INITIALISATION; 3-6 ARE RESERVED FOR          COMP     6712
000031         FUTURE SYSTEM USE*)                                                 COMP     6713
000031   CH := Ξ Ξ; EOL := FALSE ; CHCNT := 0; MAXCHCNT := 72 ; EOP := FALSE ;    LISTCTRL   52
000035   PAGECOUNT := 0 ;  LINEADDR := LC ;  LISTNAME := Ξ           Ξ ;          LISTCTRL   53
000041   PAGEEJECT := TRUE ;  LINECOUNT := 100 ;  (*FORCE INITIAL PAGE EJECT*)    LISTCTRL   54
000043   LSTMAP := Ξ 0  1          2 3 45  6  Ξ ;                                 LISTCTRL   55
000046   LSTSTR := Ξ   B0 E- P- R- T- U- X0   Ξ ;                                 LISTCTRL   56
000052   LSTSTR[ 5 ] := CHR( BUFFAC+ORD( Ξ0Ξ ) ) ;                                LISTCTRL   57
000056   IF EXTON THEN LSTSTR[ 8 ] := Ξ+Ξ ;                                       LISTCTRL   58
000061   IF PMD   THEN LSTSTR[ 11 ] := Ξ+Ξ ;                                      LISTCTRL   59
000065   IF RUNLINKAGE THEN LSTSTR[ 14 ] := Ξ+Ξ ;                                 LISTCTRL   60
000071   IF DEBUG THEN LSTSTR[ 17 ] := Ξ+Ξ ;                                      LISTCTRL   61
000075   IF MAXCHCNT = 72 THEN LSTSTR[ 20 ] := Ξ+Ξ ;                              LISTCTRL   62
000102   LSTSTR[ 23 ] := CHR( XPARMAX+ORD( Ξ0Ξ ) ) ;                              LISTCTRL   63
000106   LINELIMIT(OUTPUT,-1)   (*UNLIMITED OUTPUT*)                               COMP     6715
000107 END (*INITSCALARS*) ;                                                       COMP     6716
000122                                                                             COMP     6717
000003 PROCEDURE INITSETS;                                                         COMP     6718
000003 BEGIN                                                                       COMP     6719
000003   DIGITS := [Ξ0Ξ..Ξ9Ξ]; LETTERS := [ΞAΞ..ΞZΞ];                             COMP     6720
000010   CONSTBEGSYS := [ADDOP,INTCONST,REALCONST,CHARCONST,STRINGCONST,IDENT];    COMP     6721
000011   SIMPTYPEBEGSYS := [LPARENT]+CONSTBEGSYS;                                  COMP     6722
000012   TYPEBEGSYS := [ARROW,PACKEDSY,SEGMENTEDSY,ARRAYSY,RECORDSY,SETSY,         COMP     6723
000012         FILESY]+SIMPTYPEBEGSYS;                                             COMP     6724
000013   TYPEDELS := [ARRAYSY,RECORDSY,SETSY,FILESY];                              COMP     6725
000015   BLOCKBEGSYS := [LABELSY,CONSTSY,TYPESY,VARSY,PROCEDURESY,FUNCTIONSY,      COMP     6726
000015          BEGINSY];                                                          COMP     6727
000016   SELECTSYS := [ARROW,PERIOD,LBRACK];                                       COMP     6728
000017   FACBEGSYS := [INTCONST,REALCONST,CHARCONST,STRINGCONST,IDENT,LPARENT,     COMP     6729
000017         LBRACK,NOTSY];                                                      COMP     6730
000020   STATBEGSYS := [BEGINSY,GOTOSY,IFSY,WHILESY,REPEATSY,FORSY,WITHSY,         COMP     6731
000020         CASESY];                                                            COMP     6732
000021 END (*INITSETS*) ;                                                          COMP     6733
000033                                                                             COMP     6734
000003 PROCEDURE INITTABLES;                                                       COMP     6735
000003   VAR I: 1..ALFALENG; CH1: CHAR;                                            COMP     6736
000003   PROCEDURE RESERVEDWORDS;                                                  COMP     6737
000003   BEGIN                                                                     COMP     6738
000003     RW[ 1] := ΞIF       Ξ;RW[ 2] := ΞDO       Ξ;RW[ 3] := ΞOF       Ξ;  COMP     6739
000011     RW[ 4] := ΞTO       Ξ;RW[ 5] := ΞIN       Ξ;RW[ 6] := ΞOR       Ξ;  COMP     6740
000014     RW[ 7] := ΞEND      Ξ;RW[ 8] := ΞFOR      Ξ;RW[ 9] := ΞVAR      Ξ;  COMP     6741
000017     RW[10] := ΞDIV      Ξ;RW[11] := ΞMOD      Ξ;RW[12] := ΞSET      Ξ;  COMP     6742
000022     RW[13] := ΞAND      Ξ;RW[14] := ΞNOT      Ξ;RW[15] := ΞTHEN     Ξ;  COMP     6743
000025     RW[16] := ΞELSE     Ξ;RW[17] := ΞWITH     Ξ;RW[18] := ΞGOTO     Ξ;  COMP     6744
000030     RW[19] := ΞCASE     Ξ;RW[20] := ΞTYPE     Ξ;RW[21] := ΞFILE     Ξ;  COMP     6745
000033     RW[22] := ΞBEGIN    Ξ;RW[23] := ΞUNTIL    Ξ;RW[24] := ΞWHILE    Ξ;  COMP     6746
000036     RW[25] := ΞARRAY    Ξ;RW[26] := ΞCONST    Ξ;RW[27] := ΞLABEL    Ξ;  COMP     6747
000041     RW[28] := ΞREPEAT   Ξ;RW[29] := ΞRECORD   Ξ;RW[30] := ΞDOWNTO   Ξ;  COMP     6748
000044     RW[31] := ΞPACKED   Ξ;RW[32] := ΞPROGRAM  Ξ;RW[33] := ΞFUNCTION Ξ;  COMP     6749
000047     RW[34] := ΞPROCEDURE Ξ;RW[35] := ΞSEGMENTED Ξ;                          COMP     6750
000051     LRW[0] :=  0; LRW[1] :=  0; LRW[2] :=  6; LRW[3] := 14; LRW[4] := 21;  COMP     6751
000055     LRW[5] := 27; LRW[6] := 31; LRW[7] := 32; LRW[8] := 33; LRW[9] := 35;  COMP     6752
000062     LRW[10] := 35;                                                          COMP     6753
000063   END (*RESWORDS*) ;                                                        COMP     6754
000132                                                                             COMP     6755
000003   PROCEDURE SYMBOLS;                                                        COMP     6756
```

MOORE BUSINESS FORMS, INC. F
FORM 213
PRINTED IN U.S.A.
22

```
000003   BEGIN                                                                  COMP     6757
000003    RSY[1] := IFSY; RSY[2] := DOSY; RSY[3] := OFSY; RSY[4] := TOSY;       COMP     6758
000011    RSY[5] := RELOP; RSY[6] := ADDOP; RSY[7] := ENDSY; RSY[8] := FORSY;   COMP     6759
000015    RSY[9] := VARSY; RSY[10] := MULOP;RSY[11] := MULOP; RSY[12] := SETSY; COMP     6760
000021    RSY[13] := MULOP; RSY[14] := NOTSY; RSY[15] := THENSY;                COMP     6761
000024    RSY[16] := ELSESY; RSY[17] := WITHSY; RSY[18] := GOTOSY;              COMP     6762
000027    RSY[19] := CASESY; RSY[20] := TYPESY; RSY[21] := FILESY;              COMP     6763
000032    RSY[22] := BEGINSY; RSY[23] := UNTILSY; RSY[24] := WHILESY;           COMP     6764
000035    RSY[25] := ARRAYSY; RSY[26] := CONSTSY; RSY[27] := LABELSY;           COMP     6765
000040    RSY[28] := REPEATSY; RSY[29] := RECORDSY; RSY[30] := DOWNTOSY;        COMP     6766
000043    RSY[31] := PACKEDSY; RSY[32] := PROGRAMSY; RSY[33] := FUNCTIONSY;     COMP     6767
000046    RSY[34] := PROCEDURESY; RSY[35] := SEGMENTEDSY;                       COMP     6768
000050    SSY[Ξ+Ξ] := ADDOP; SSY[Ξ-Ξ] := ADDOP; SSY[Ξ*Ξ] := MULOP;             COMP     6769
000053    SSY[Ξ/Ξ] := MULOP; SSY[Ξ(Ξ] := LPARENT; SSY[Ξ)Ξ] := RPARENT;         COMP     6770
000056    SSY[Ξ$Ξ] := OTHERSY; SSY[Ξ=Ξ] := RELOP; SSY[Ξ Ξ] := OTHERSY;         COMP     6771
000061    SSY[Ξ,Ξ] := COMMA; SSY[Ξ.Ξ] := PERIOD; SSY[ΞΞΞΞ] := OTHERSY;         COMP     6772
000064    SSY[Ξ[Ξ] := LBRACK; SSY[Ξ]Ξ] := RBRACK; SSY[PER] := COLON;           PERCENTB    2
000067    SSY[Ξ≠Ξ] := RELOP; SSY[Ξ↑Ξ] := OTHERSY;                              COMP     6774
000071    SSY[Ξ∨Ξ] := ADDOP; SSY[Ξ∧Ξ] := MULOP;                                COMP     6775
000073    SSY[Ξ↑Ξ] := ARROW; SSY[Ξ↓Ξ] := OTHERSY;                              COMP     6776
000075    SSY[Ξ<Ξ] := RELOP; SSY[Ξ>Ξ] := RELOP;                                COMP     6777
000077    SSY[Ξ≤Ξ] := RELOP; SSY[Ξ≥Ξ] := RELOP;                                COMP     6778
000101    SSY[Ξ¬Ξ] := NOTSY; SSY[Ξ;Ξ] := SEMICOLON;                            COMP     6779
000103   END (*SYMBOLS*) ;                                                      COMP     6780
000107                                                                          COMP     6781
000003   PROCEDURE OPERATORS;                                                   COMP     6782
000003    VAR I: INTEGER; CH: CHAR;                                             COMP     6783
000005   BEGIN                                                                  COMP     6784
000005     FOR I := 1 TO RESWORDS DO ROP[I] := NOOP;                           COMP     6785
000014     ROP[5] := INOP; ROP[10] := IDIV; ROP[11] := IMOD;                   COMP     6786
000017     ROP[6] := OROP; ROP[13] := ANDOP;                                   COMP     6787
000021     FOR CH := Ξ+Ξ TO Ξ;Ξ DO SOP[CH] := NOOP;                            COMP     6788
000027     SOP[Ξ+Ξ] := PLUS;SOP[Ξ-Ξ] := MINUS;SOP[Ξ*Ξ] := MUL;SOP[Ξ/Ξ] := RDIV;  COMP     6789
000032     SOP[Ξ=Ξ] := EQOP;SOP[Ξ≠Ξ] := NEOP;SOP[Ξ∨Ξ] := OROP;SOP[Ξ∧Ξ] := ANDOP; COMP     6790
000036     SOP[Ξ<Ξ] := LTOP;SOP[Ξ>Ξ] := GTOP;SOP[Ξ≤Ξ] := LEOP;SOP[Ξ≥Ξ] := GEOP;  COMP     6791
000041    END (*OPERATORS*) ;                                                   COMP     6792
000045                                                                          COMP     6793
000045   BEGIN (*INITTABLES*)                                                   COMP     6794
000045    RESERVEDWORDS; SYMBOLS; OPERATORS;                                    COMP     6795
000013    PASCF := ΞPASCFL    Ξ;                                                COMP     6796
000015    FOR I := 7 TO ALFALENG DO PASCF[I] := CHR(0);                         COMP     6797
000026    PASCL := PASCF; PASCL[5] := ΞLΞ; PASCL[6] := Ξ.Ξ;                     COMP     6798
000034    PNAME := ΞPRC       Ξ;                                                COMP     6799
000035    FOR I:= 8 TO ALFALENG DO PNAME[I] := CHR(0)                           COMP     6800
000041   END (*INITTABLES*) ;                                                   COMP     6801
000054                                                                          OVERLAY     2
000003   PROCEDURE OVL ;  EXTERN ;                                              OVERLAY     3
000003                                                                          CTIME       2
000003   PROCEDURE CTIMEOUT ;                                                   CTIME       3
000003    VAR  CTIMEMSG : PACKED ARRAY[ 1..40 ] OF CHAR ;  I,J : INTEGER ;      CTIME       4
000011   BEGIN  CTIMEMSG := Ξ-    S.MMM CP SECONDS COMPILATION TIME  Ξ ;        CTIME       5
000012    FOR I := 0 TO 2 DO                                                    CTIME       6
000012     BEGIN  J := CTIME ;  CTIME := CTIME DIV 10 ;                         CTIME       7
000020      CTIMEMSG[ 10-I ] := CHR( ORD( Ξ0Ξ )+J-10*CTIME ) ;                  CTIME       8
000032     END ;                                                               CTIME       9
000033    I := 4 ;                                                             CTIME      10
```

MOORE BUSINESS FORMS, INC. FORM 413 PRINTED IN U.S.A. 22

```
000034   REPEAT  J := CTIME ;  CTIME := CTIME DIV 10 ;                    CTIME     11
000041    CTIMEMSG[ 10-I ] := CHR( ORD( '0' )+J-10*CTIME ) ;               CTIME     12
000054    I := I+1 ;                                                       CTIME     13
000055   UNTIL ( CTIME = 0 ) OR ( I = 9 ) ;                               CTIME     14
000060   IF I = 9 THEN CTIMEMSG[ 2 ] := '*' ;                             CTIME     15
000064   MESSAGE( CTIMEMSG ) ;                                            CTIME     16
000066  END ;                                                             CTIME     17
000076                                                                    COMP    6802
000076 BEGIN                                                              COMP    6803
000076                                                                    COMP    6804
000076  (*INITIALIZE:*)                                                   COMP    6805
000076  (*************)                                                   COMP    6806
000076  REWRITE( XREFFIL ) ;                                              CROSSREF  26
000035  CTIME := CLOCK ;                                                  CTIME     18
000037  DATE( LDATE ) ;  TIME( LTIME ) ;                                 LISTCTRL  64
000041  PAGE( OUTPUT ) ;                                                  LISTCTRL  65
000045  INITSCALARS; INITSETS; INITTABLES;                                COMP    6807
000050                                                                    COMP    6808
000050  (*ENTER STANDARD NAMES AND STANDARD TYPES:*)                      COMP    6809
000050  (******************************************)                      COMP    6810
000050                                                                    COMP    6811
000050  LEVEL := 0; TOP := 0;                                             COMP    6812
000052  WITH DISPLAY[0] DO                                                COMP    6813
000052   BEGIN FNAME := NIL; OCCUR := BLCK END;                           COMP    6814
000057  STDTYPENTRIES; STANDARDNAMES; STDNAMENTRIES; ENTERUNDECL;         COMP    6815
000063  TOP := 1; LEVEL := 1;                                             COMP    6816
000065  WITH DISPLAY[1] DO                                                COMP    6817
000065   BEGIN FNAME := NIL; OCCUR := BLCK END;                           COMP    6818
000072                                                                    COMP    6819
000072  (*COMPILE:*)                                                      COMP    6820
000072  (**********)                                                      COMP    6821
000072                                                                    COMP    6822
000072  INSYMBOL;                                                         COMP    6824
000073  PROGRAMME(BLOCKBEGSYS+STATBEGSYS-[CASESY]);                       COMP    6825
000076  WHILE NOT( EOF( INPUT ) OR EOLN( INPUT ) ) DO                     UPDATE9   81
000100   BEGIN  CHCNT := CHCNT+1 ;  READ( LINEBUFFER[ CHCNT ] ) ;  END ;  OVLAP9     1
000105                                                                    COMP    6826
000105  EOP := TRUE ;  ENDOFLINE ;                                        LISTCTRL  66
000107 13:                                                                COMP    6829
000107  PUTSEG( XREFFIL ) ;                                               CROSSREF  27
000110  WITH XREFFIL^ DO                                                  CROSSREF  28
000112   BEGIN  AVAL := LISTNAME ;  LNO := 0 ;  PNO := PAGECOUNT ;        CROSSREF  29
000120   END ;                                                            CROSSREF  30
000120  PUT( XREFFIL ) ;                                                  CROSSREF  31
000121  XREFFIL^.AVAL := LDATE ;  PUT( XREFFIL ) ;                        CROSSREF  32
000124  XREFFIL^.AVAL := LTIME ;  PUT( XREFFIL ) ;                        CROSSREF  33
000127  PUTSEG( XREFFIL ) ;                                               CROSSREF  34
000130  CTIME := CLOCK-CTIME ;  CTIMEOUT ;                                CTIME     19
000133  PUTSEG( OUTPUT ) ;  OVL ;                                         OVERLAY    4
000135 END .                                                              COMP    6833
```

MOORE BUSINESS FORMS, INC. FORM 1413 PRINTED IN U.S.A. 22