

0001

ASMB,R,B,L

** NO ERRORS PASS#1 **R1E ASMB 92067-16011**

```
0001          ASMB,R,B,L
0002 00000      NAM JBINC,3
0003          EXT EXEC
0004*
0005* CLEARS JOB BINARY AREA
0006* (FRITZ JOERN )
0007*
0008 00000 177755 RCODE DEC -19
0009 00001 000006 CCODE DEC 6
0010 00002 000000 JBINC NOP
0011 00003 065020R      LDB =B101
0012 00004 002400      CLA
0013 00005 015001X      JSB EXEC      CLEAR JBINS
0014 00006 000010R      DEF **2
0015 00007 000000R      DEF RCODE
0016 00010 065021R      LDB =B102
0017 00011 002400      CLA
0018 00012 015001X      JSB EXEC      CLEAR JBINC
0019 00013 000015R      DEF **2
0020 00014 000000R      DEF RCODE
0021 00015 015001X      JSB EXEC
0022 00016 000020R      DEF **2
0023 00017 000001R      DEF CCODE
      00020 000101
      00021 000102
0024          END JBINC
** NO ERRORS *TOTAL **RTE ASMB 92067-16011**
```

0001

ASMB,R,B,L

** NO ERRORS PASS#1 **RTE ASMB 92067-16011**

```

C001          ASMB,R,B,L
C002*
C003 00000      NAM BASIC
C004*
C005**
C006*****DUSM SUPER BASIC*****
C007**
C008*
C009**
C010***** SPECIAL SWITCH REGISTER FUNCTIONS
C011**
C012**
C013*      SW 15      1      ALL OUTPUT ON LINE PRINTER
C014*              0      ALL OUTPUT ON SYSTEM TTY
C015*      SW 14      1      TERMINATE EXECUTION OR LISTING
C016*      SW 13      1      DELETE ALL REMARKS
C017*
C018              COM PGPTR,PGPT1,PGPT2,PGPT,BGNPG
C019              COM PG1,PG2,PBUFF,PBPTR,FTRAK
C020              COM BFBL
C021*
C022*
C023*      PGPTR      TEMP VIRTUAL ADDRESS POINTER
C024*      PGPT1      PAGE 1 VIRTUAL ADDRESS POINTER
C025*      PGPT2      PAGE 2 VIRTUAL ADDRESS POINTER
C026*      PGPT      ADDRESS OF CURRENT PAGE POINTER
C027*      BGNPG      TEMP POINTER TO START OF CURRENT CORE RES. PAGE
C028*      PG1      START OF CORE PAGE #1
C029*      PG2      START OF CORE PAGE #2
C030*      PBUFF      FIRST WORD OF USERS PROGRAM
C031*      PBPTR      LAST WORD+1 OF USERS PROGRAM
C032*      FTRAK      START OF WORK AREA USED FOR IM CODE STORAGE
C033*      BFBL      CORE BUFFER LENGTH (MULTIPLE # OF SECTORS)
C034 00000 014607R LWBM DEF LWAMA
C035 00001 000000      BSS 1
C036 00002 000002 FWAM ABS 2      FIRST WORD PROGRAM MEMORY
C037 00003 077777 LWAM ABS 32767      LAST WORD PROGRAM MEMORY
C038 00004 000001C PT1 DEF PGPT1      ADDR OF V.A. POINTER FOR PAGE 1
C039 00005 000002C PT2 DEF PGPT2      ADDR OF V.A. POINTER FOR PAGE 2
C040 00006 000000 FCORE BSS 1      START OF FREE CORE
C041 00007 000000 FDISK BSS 1      START OF FREE DISK STORAGE
C042 00010 000000 SYMTB BSS 1      START OF SYMBOL TABLE
C043 00011 014751R SYMTA DEF SYMTB      SYMBOL TABLE END
C044 00012 000000 LSTAK BSS 1      LOW-CORE STACK ADDRESS
C045 00013 013313R ASBTB DEF FINIS
C046 00014 013313R SBIBB DEF FINIS
C047 00015 000000 OLDLN BSS 1      PREVIOUS LINE #
C048 00016 014751R .BUFA DEF SYMTB
C049 00017 000000 BADDR BSS 1      I/O BUFFER
C050 00020 000000 CCNT BSS 1      POINTERS
C051 00021 014610R SBUFA DEF SBUFB
C052 00022 000000 SBPTR BSS 1      SYNTAX BUFFER POINTER
C053 00023 000000 TFLAG BSS 1
C054 00024 000000 TTYFL BSS 1
C055 00025 000000 ENFLG BSS 1

```

C056	00026	000000	TSIPT	BSS	1	TEMPORARY STACK POINTER
C057	00027	000000	LSTPT	BSS	1	LOW-CORE STACK POINTER
C058	00030	000000	HSTPT	BSS	1	HIGH-CORE STACK POINTER
C059	00031	000000	PRA00	BSS	1	PROGRAM EXECUTION
C060	00032	000000	NXTST	BSS	1	SEQUENCING INFORMATION
C061	00033	000000	.LNUM	BSS	1	CURRENT LINE NUMBER
C062	00034	000000	TYPE	BSS	1	CURRENT STATEMENT TYPE
C063	00035	000000	DSTRT	BSS	1	DATA
C064	00036	000000	NXTDI	BSS	1	STATEMENT
C065	00037	000000	DCCNT	BSS	1	POINTERS
C066	00040	000000	RSYM	BSS	1	
C067	00041	000000	SIGN	BSS	1	
C068	00042	000000	EXP	BSS	1	
C069	00043	000000	XH	BSS	1	RANDOM
C070	00044	000000	XL	BSS	1	VARIABLE
C071	00045	000000	TEMPS	BSS	12	TEMPORARIES
C072	00057		MLBX1	EQU	TEMPS+10	
C073	00061	000000	B1	BSS	2	
C074	00063	000000	B2	BSS	2	
C075	00065	000000	B3	BSS	2	
C076	00067	011077R	.LOGA	DEF	.LOG	
C077	00070	015274R	EOF	JSB	ERROR	
C078	00071	015274R	NOEOF	JSB	ERROR	
C079	00072	010007R	EBUFA	DEF	EBUFF	
C080	00073	010012R	EBFA	DEF	EBFF-1	
C081	00074	010015R	LBUFA	DEF	LBUFF	
C082	00075	010021R	LNBFA	DEF	LNBFF-1	
C083	00076	007673R	ERBS	DEF	ERR-1	
C084	00077	000100	RECER	DEF	RCERR-ERR	
C085	00100	002225R	FOPBS	DEF	QUOTE-2	
C086	00101	103041R	STBAS	DEF	SYNTB-26,I	
C087	00102	106114R	XECBR	DEF	XECTB-26,I	
C088	00103	107111R	ARBAS	DEF	AROTB-6,I	
C089	00104	010024R	PDFBS	DEF	PDFT-1	
C090	00105	003765R	TBLAD	DEF	SYCMD	
C091	00106	004025R	STTYP	DEF	LET	
C092	00107	004106R	MATIO	DEF	READ	
C093	00110	004144R	MCBOP	DEF	AND	
C094	00111	004162R	PDFNS	DEF	SIN	
C095	00112	004223R	MATHN	DEF	ZER	
C096	00113	004054R	ANEXT	DEF	NEXT	
C097	00114	004067R	AEND	DEF	END	
C098	00115	004103R	ADATA	DEF	DATA	
C099	00116	004131R	ATHEN	DEF	THEN	
C100	00117	004134R	ATO	DEF	TO	
C101	00120	004136R	ASTEP	DEF	STEP	
C102	00121	004141R	ANOT	DEF	NOT	
C103	00122	004157R	ATAB	DEF	TAB	
C104	00123	000057R	MBXL	DEF	MLBX1	
C105	00124	013313R	BGNIB	DEF	FINIS	START OF CALL LINKAGE TABLE
C106	00125	013313R	ENDIB	DEF	FINIS	LAST WORD +1 OF CALL TABLE
C107	00000		A	EQU	0	
C108	00001		B	EQU	1	
C109	00053		.0	EQU	533	
C110	00054		.1	EQU	.0+1	

C111	00055	.2	EQU	.0+2
C112	00056	.3	EQU	.0+3
C113	00057	.4	EQU	.0+4
C114	00060	.5	EQU	.0+5
C115	00061	.6	EQU	.0+6
C116	00062	.7	EQU	.0+7
C117	00063	.8	EQU	.0+8
C118	00064	.9	EQU	.0+9
C119	00065	.10	EQU	.0+10
C120	00126	000013	.11	DEC 11
C121	00127	000014	.12	DEC 12
C122	00130	000016	.14	DEC 14
C123	00070		.15	EQU .0+13
C124	00131	000022	.18	DEC 18
C125	00132	000025	.21	DEC 21
C126	00133	000026	.22	DEC 22
C127	00134	000027	.23	DEC 23
C128	00135	000032	.26	DEC 26
C129	00136	000033	.27	DEC 27
C130	00137	000034	.28	DEC 28
C131	00140	000036	.30	DEC 30
C132	00071		.31	EQU .0+14
C133	00141	000040	.32	DEC 32
C134	00142	000041	.33	DEC 33
C135	00143	000042	.34	DEC 34
C136	00144	000045	.37	DEC 37
C137	00145	000050	.40	DEC 40
C138	00146	000051	.41	DEC 41
C139	00147	000053	.43	DEC 43
C140	00150	000055	.45	DEC 45
C141	00151	000054	.44	DEC 44
C142	00152	000056	.46	DEC 46
C143	00153	000057	.47	DEC 47
C144	00154	000060	.48	DEC 48
C145	00155	000061	.49	DEC 49
C146	00156	000062	.50	DEC 50 ***:
C147	00157	000072	.58	DEC 58
C148	00072		.63	EQU .0+15
C149	00160	000100	B100	OCT 100
C150	00161	000105	E	OCT 105
C151	00162	000106	F	OCT 106
C152	00163	000110	.72	DEC 72
C153	00164	000112	.74	DEC 74
C154	00165	000113	.75	DEC 75
C155	00166	000116	N	OCT 116
C156	00167	000123	S	OCT 123
C157	00072		B77	EQU .63
C158	00170	000133	B133	OCT 133
C159	00171	000137	B137	OCT 137
C160	00073		B177	EQU .0+16
C161	00172	000200	B200	OCT 200
C162	00173	000777	MSK1A	OCT 777 :::
C163	00074		MSK0	EQU .0+17
C164	00174	000400	B400	OCT 400
C165	00175	000776	B776	OCT 776

C166	00176	000377	MSK1	OCT	377	::
C167	00177	001000	B1000	OCT	1000	::
C168	00200	001400	B1400	OCT	1400	::
C169	00201	003002	SCCNT	OCT	3002	
C170	00202	002000	B2000	OCT	2000	::
C171	00203	002400	B2400	OCT	2400	::
C172	00204	005000	B6000	OCT	6000	::
C173	00205	010400	UNMNC	OCT	10400	::
C174	00206	011000	B1100	OCT	11000	::
C175	00207	011400	B1140	OCT	11400	::
C176	00210	020000	B20K	OCT	20000	
C177	00211	016400	DEFOP	OCT	16400	::
C178	00212	017000	REMOP	OCT	17000	::
C179	00213	025000	R00P	OCT	25000	::
C180	00214	031000	WROP	OCT	31000	
C181	00215	025400	PRNOP	OCT	25400	
C182	00216	005000	LF	OCT	5000	
C183	00217	053146	TENTH	OCT	53146	
C184	00220	077400	OPMSK	OCT	77400	::
C185	00221	077600	MSK4	OCT	77600	
C186	00222	077777	INF	OCT	77777	
C187	00223	100007	TYPEFL	OCT	100007	
C188	00224	100017	TABCN	OCT	100017	
C189	00225	100377	OPDMK	OCT	100377	::
C190	00226	130000	RMODE	OCT	130000	
C191	00227	140000	UNNRM	OCT	140000	
C192	00230	174000	HIMSK	OCT	174000	
C193	00231	100000	N1	DEC	-1.0	
	00232	000000				
C194	00052		M1	EQU	.0-1	
C195	00051		M2	EQU	.0-2	
C196	00050		M3	EQU	.0-3	
C197	00047		M4	EQU	.0-4	
C198	00046		M5	EQU	.0-5	
C199	00045		M6	EQU	.0-6	
C200	00044		M7	EQU	.0-7	
C201	00043		M8	EQU	.0-8	
C202	00042		M9	EQU	.0-9	
C203	00041		M10	EQU	.0-10	
C204	00233	177765	M11	DEC	-11	
C205	00234	177764	M12	DEC	-12	
C206	00235	177761	M15	DEC	-15	
C207	00236	177760	M16	DEC	-16	
C208	00237	177753	M21	DEC	-21	
C209	00240	177740	M32	DEC	-32	
C210	00241	177725	D53	OCT	-53	
C211	00242	177716	D62	OCT	-62	
C212	00243	177715	D63	OCT	-63	
C213	00244	177770	SBCUN	OCT	177770	
C214	00245	100001	STCON	OCT	100001	
C215	00246	177706	D72	OCT	-72	
C216	00077		D100	EQU	.0+20	
C217	00247	177670	M72	DEC	-72	
C218	00250	177667	M73	DEC	-73	
C219	00251	177664	M76	DEC	-76	

0220	00252	177645	D133	OCT	-153	
0221	00253	177600	D200	OCT	-200	
0222	00075		M256	EQU	.0+18	
0223	00254	177312	M310	DEC	-310	
0224	00255	176030	M1000	DEC	-1000	
0225	00256	100000	MAXSN	DEC	-32768	
0226	00044		MSK3	EQU	M7	
0227	00257	043116	FN	ASC	1, FN	
0228	00260	037440	QMARK	ASC	1, ?	
0229	00261	040000	HALF	OCT	40000	
0230	00262	000000		OCT	0	
0231	00261		HONE	EQU	HALF	
0232	00263	100000	MNEG	OCT	100000	MAXIMUM NEGATIVE FLOATING
0233	00264	000376		OCT	376	POINT NUMBER
0234	00263		FLGBT	EQU	MNEG	
0235	00265	102756	MAXFX	DEC	-999999.5	
	00266	002050				
0236	00267	114631	MINFX	DEC	-0.099999959	
	00270	116373				
0237	00271	000040	BLANK	OCT	40	
0238	00272	000000	PARM	BSS	1	
0239	00273	177753	TBLX	DEC	-21	

BASE PAGE SUBROUTINES

```

0241**
0242***  EMIT ERROR MESSAGE  **
0243**
0244 00274 000000  ERROR NOP
0245 00275 063741R      LDA PRNTL          SHIFT TO
0246 00276 073131R      STA LISTR          COMMAND MODE
0247 00277 062020R      LDA CCNT          SAVE
0248 00300 073266R      STA OPCHK          OUTPUT
0249 00301 062017R      LDA BADDR          BUFFER
0250 00302 073321R      STA RSCHK          POINTERS
0251 00303 062073R      LDA EBFA          SET BUFFER
0252 00304 072017R      STA BADDR          POINTER
0253 00305 060063      LDA .8             SET CHARACTER
0254 00306 072020R      STA CCNT          COUNT
0255 00307 065274R      LDB ERROR          ERROR SOURCE IN (B)
0256 00310 062076R      LDA ERBS          ERROR ADDRESS IN (A)
0257 00311 002004      INA              MOVE TO NEXT ERROR
0258 00312 154000      CPB 0,1           SAME AS ACTUAL ERROR?
0259 00313 003005      CMA,INA,RSS        YES
0260 00314 025311R      JMP *-3           NO
0261 00315 042076R      ADA ERBS          COMPUTE ERROR
0262 00316 073426R      STA ENOUT          SAVE NEGATIVE OF ERROR
0263 00317 003004      CMA,INA           NUMBER
0264 00320 017132R      JSB OUTIN          NUMBER TO BUFFER
0265 00321 065072R      LDB EBUFA          LOAD BUFFER ADDRESS
0266 00322 062020R      LDA CCNT          LOAD NEGATIVE OF
0267 00323 017161R      JSB EXECW          OUTPUT ERROR MESSAGE
0268 00324 062075R      LDA LNBFA          OUTPUT
0269 00325 072017R      STA BADDR
0270 00326 060065      LDA .10
0271 00327 072020R      STA CCNT          LINE
0272 00330 062033R      LDA .LNUM
0273 00331 017132R      JSB OUTIN
0274 00332 065074R      LDB LBUFA          NUMBER
0275 00333 062020R      LDA CCNT
0276 00334 017161R      JSB EXECW
0277 00335 063426R      LDA ENOUT          RETRIEVE NEGATIVE OF ERROR
0278 00336 042077R      ADA RECER          RECOVERABLE
0279 00337 002021      SSA,RSS
0280 00340 027611R      JMP PEXMK        NO, RETURN TO SYNTAX MODE
0281 00341 040061      ADA .6
0282 00342 002020      SSA
0283 00343 025340R      JMP *-3
0284 00344 062226R      LDA RMODE          RETURN TO
0285 00345 073131R      STA LISTR          RUN MODE
0286 00346 063266R      LDA OPCHK          RESTORE
0287 00347 072020R      STA CCNT          OUTPUT
0288 00350 063321R      LDA RSCHK          BUFFER
0289 00351 072017R      STA BADDR          POINTERS
0290 00352 126274R      JMP ERROR,I      RETURN TO PROGRAM

```

BASE PAGE SUBROUTINES

C292**

C293*** MOVE WORDS TO HIGHER DISK

C294**

C295	00353	000000	MVTOH	NOP	
C296	00354	066050R	LDB	TEMP2	FETCH SOURCE ADDRESS
C297	00355	056051R	MVTO1	CPB	TEMP3
					ALL RELOCATION DONE?
C298	00356	126353R	JMP	MVTOH,I	YES, EXIT
C299	00357	003400	CCA		BACK UP
C300	00360	042052R	ADA	TEMP4	SOURCE AND
C301	00361	072052R	STA	TEMP4	.DESTINATION
C302	00362	044052	ADB	M1	ADDRESSES
C303	00363	017051R	JSB	LD6I	MOVE WORD
C304	00364	017151R	JSB	DSTOR	WORD
C305	00365	000052R	DEF	TEMP4	
C306	00366	026355R	JMP	MVTO1	

C307**

C308*** MOVE WORDS TO HIGHER CORE

C309**

C310	00367	000000	MVTOC	NOP
C311	00370	066050R	LDB	TEMP2
C312	00371	056051R	MVT1	CPB
				TEMP3
C313	00372	126367R	JMP	MVTOC,I
C314	00373	003400	CCA	
C315	00374	042052R	ADA	TEMP4
C316	00375	072052R	STA	TEMP4
C317	00376	044052	ADB	M1
C318	00377	160001	LDA	1,I
C319	00400	172052R	STA	TEMP4,I
C320	00401	026371R	JMP	MVT1

C321**

C322*** INPUT A CONSTANT **

C323**

C324	00402	000000	CONST	NOP	
C325	00403	017377R	JSB	GETCR	
C326	00404	126402R	JMP	CONST,I	
C327	00405	005400	CLB		SET SIGN
C328	00406	076041R	STB	SIGN	POSITIVE
C329	00407	006004	INB		
C330	00410	052147R	CPA	.43	'+' ?
C331	00411	026415R	JMP	CONS1	YES
C332	00412	052150R	CPA	.45	NO, '-' ?
C333	00413	007401	CCB,RSS		YES
C334	00414	026420R	JMP	CONS2	NO
C335	00415	076041R	CONS1	STB	SIGN
					RECORD SIGN
C336	00416	017377R	JSB	GETCR	FETCH NEXT
C337	00417	026426R	JMP	SYE12-1	CHARACTER
C338	00420	016430R	CONS2	JSB	NUMCK
					FETCH CONSTANT
C339	00421	026424R	JMP	CONS3	NONE FOUND
C340	00422	036402R	ISZ	CONST	SUCCESSFULLY FOUND,
C341	00423	126402R	JMP	CONST,I	EXIT VIA (P+2)
C342	00424	056041R	CONS3	CPB	SIGN
					SIGN FOUND? ((B) = 0)
C343	00425	003401	CCA,RSS		NO
C344	00426	016274R	JSB	ERROR	YES, SOLITARY SIGN
C345	00427	126402R	SYE12	JMP	CONST,I
					EXIT VIA (P+1)

C346**

```

C347***  FETCH NUMBER AND CONVERT TO BINARY  **
C348**
C349  00430  000000  NUMCK  NOP          CHARACTER IN (A), SIGN SET
C350  00431  005400          CLB
C351  00432  075042R          STB EXP      ZERO
C352  00433  077066R          STB MANT1    ALL
C353  00434  077130R          STB MANT2    COMPONENTS
C354  00435  077366R          STB EXPON    OF NUMBER
C355  00436  075051R          STB TEMP3    SET 'NUMBER' FLAG FALSE
C356  00437  007400          CCB          SET 'DECIMAL POINT'
C357  00440  077416R          STB DPFLG    FLAG FALSE
C358  00441  052152R  NUMC1  CPA .46      DECIMAL POINT?
C359  00442  037416R          ISZ DPFLG    YES, SET FLAG TRUE
C360  00443  026447R          JMP NUMC2    NO
C361  00444  002400          CLA          INITIALIZE POST-DECIMAL DIGIT
C362  00445  073366R          STA EXPON    DIGIT COUNTER TO ZERO
C363  00446  026466R          JMP NUMC3+1  FETCH A CHARACTER
C364  00447  017353R  NUMC2  JSB DIGCK    DIGIT?
C365  00450  025521R          JMP NUMC7    NO
C366  00451  037366R          ISZ EXPON    YES, COUNT DIGIT
C367  00452  001727          ALF,ALF      LEFT-JUSTIFY
C368  00453  001723          ALF,RAR      DIGIT AND
C369  00454  072052R          STA TEMP4    SAVE IT
C370  00455  016777R          JSB MBY10   MULTIPLY PREVIOUS NUMBER BY 10
C371  00456  065042R          LDB EXP
C372  00457  005002          SZB          ZERO EXPONENT?
C373  00460  026472R          JMP NUMC4    NO
C374  00461  060057          LDA .4        YES, SET
C375  00462  072042R          STA EXP      EXPONENT TO 4
C376  00463  062052R          LDA TEMP4    LOAD
C377  00464  005400          CLB          NUMBER
C378  00465  016742R  NUMC3  JSB NORML    NORMALIZE THE NUMBER
C379  00466  035051R          ISZ TEMP3    SET 'NUMBER OCCURRED' FLAG
C380  00467  017377R          JSB GETCR    ANOTHER CHARACTER?
C381  00470  025570R          JMP NUM12    NO
C382  00471  026441R          JMP NUMC1    YES
C383  00472  044047  NUMC4  ADB M4        COMPUTE
C384  00473  007000          CMB          EXPONENT
C385  00474  062052R          LDA TEMP4    BIAS AND
C386  00475  075052R          STB TEMP4    SAVE IT
C387  00476  005400          CLB
C388  00477  035052R  NUMC5  ISZ TEMP4    DIGIT POSITIONED?
C389  00500  025516R          JMP NUMC6    NO
C390  00501  000040          CLE          YES, ADD IN
C391  00502  047130R          ADB MANT2    LOW PART
C392  00503  103101          CLO          OF NUMBER
C393  00504  002040          SEZ          OVERFLOW?
C394  00505  002004          INA          YES, BUMP (A)
C395  00506  043066R          ADA MANT1    ADD IN HIGH PART OF NUMBER
C396  00507  102301          SOS          OVERFLOW?
C397  00510  026465R          JMP NUMC3    NO
C398  00511  000065          CLE,ERA      YES, ROTATE
C399  00512  005500          ERB          DOWN AND
C400  00513  035042R          ISZ EXP      BUMP
C401  00514  000000          NOP          EXPONENT

```

BASE PAGE SUBROUTINES

0402	00515	026465R	JMP NUMC3	
0403	00516	000065	NUMC6 CLE,ERA	SHIFT
0404	00517	005500	ERB	DIGIT
0405	00520	026477R	JMP NUMC5	RIGHT
0406	00521	006400	NUMC7 CLB	DECIMAL POINT
0407	00522	076052R	STB TEMP4	
0408	00523	056051R	CPB TEMP3	OR DIGIT FOUND?
0409	00524	126430R	JMP NUMCK,I	NO, EXIT VIA (P+1)
0410	00525	052161R	CPA E	YES, 'E' ?
0411	00526	002001	RSS	YES
0412	00527	026570R	JMP NUM12	NO, NO EXPONENT PART
0413	00530	017377R	JSB GETCR	
0414	00531	016274R	NUMER JSB ERROR	
0415	00532	052147R	CPA .43	'+' ?
0416	00533	026540R	JMP NUMC8	YES
0417	00534	052150R	CPA .45	NO, '-' ?
0418	00535	003401	CCA,RSS	YES
0419	00536	026542R	JMP NUMC9	NO
0420	00537	072052R	STA TEMP4	NOTE MINUS SIGN
0421	00540	017377R	NUMC8 JSB GETCR	
0422	00541	026531R	JMP NUMER	
0423	00542	017353R	NUMC9 JSB DIGCK	DIGIT?
0424	00543	026531R	JMP NUMER	NO
0425	00544	072051R	STA TEMP3	YES, SAVE IT
0426	00545	017377R	JSB GETCR	
0427	00546	026564R	JMP NUM10	SECOND
0428	00547	017353R	JSB DIGCK	DIGIT?
0429	00550	026564R	JMP NUM10	NO
0430	00551	066051R	LDB TEMP3	YES
0431	00552	005020	BLS,BLS	MULTIPLY
0432	00553	046051R	ADB TEMP3	PRIOR DIGIT
0433	00554	005000	BLS	BY 10
0434	00555	040001	ADA 1	ADD NEW DIGIT
0435	00556	072051R	STA TEMP3	SAVE EXPONENT
0436	00557	017377R	JSB GETCR	
0437	00560	026564R	JMP NUM10	THIRD
0438	00561	017353R	JSB DIGCK	DIGIT?
0439	00562	002001	RSS	NO
0440	00563	026531R	JMP NUMER	YES
0441	00564	062051R	NUM10 LDA TEMP3	LOAD EXPONENT
0442	00565	036052R	ISZ TEMP4	POSITIVE?
0443	00566	003004	CMA,INA	YES, COMPLEMENT IT
0444	00567	002001	RSS	NO
0445	00570	002400	NUM12 CLA	CLEAR IF NO EXPONENT PART
0446	00571	037416R	ISZ DPFLG	DECIMAL POINT?
0447	00572	043366R	ADA EXPON	YES, CORRECT EXPONENT
0448	00573	002003	SZA,RSS	ZERO EXPONENT?
0449	00574	026611R	JMP NUM14	YES
0450	00575	002020	SSA	NO, NEGATIVE EXPONENT?
0451	00576	026605R	JMP NUM13	NO
0452	00577	003004	CMA,INA	YES, SET
0453	00600	073366R	STA EXPON	COUNTER
0454	00601	017030R	JSB DBY10	DIVIDE NUMBER BY 10
0455	00602	037366R	ISZ EXPON	DONE?
0456	00603	026601R	JMP *-2	NO

BASE PAGE SUBROUTINES

```

C457 00604 025611R      JMP NUM14      YES
C458 00605 073366R NUM13 STA EXPON     SET COUNTER
C459 00606 015777R      JSB MBY10      MULTIPLY BY 10
C460 00607 037366R      ISZ EXPON      DONE?
C461 00610 025606R      JMP *-2        NO
C462 00611 063066R NUM14 LDA MANT1      YES, LOAD
C463 00612 067130R      LDB MANT2      NUMBER
C464 00613 035041R      ISZ SIGN       POSITIVE?
C465 00614 025620R      JMP NUM15      YES
C466 00615 003000      CMA             NO,
C467 00616 007007      CMB,INB,SZB,RSS COMPLEMENT
C468 00617 002004      INA             IT
C469 00620 015647R NUM15 JSB .PACK      PACK NUMBER INTO (A) AND (B)
C470 00621 104400      DST SVOP1
      00622 015316R
C471 00623 062022R      LDA SBPTR
C472 00624 002020      SSA
C473 00625 025642R      JMP NUM17
C474 00626 104200      DLD SVOP1
      00627 015316R
C475 00630 035022R      ISZ SBPTR
C476 00631 172022R      STA SBPTR,I     STORE
C477 00632 035022R      ISZ SBPTR      NUMBER IN
C478 00633 175022R      STB SBPTR,I     PROPER
C479 00634 035022R      ISZ SBPTR      LOCATION
C480 00635 017416R NUM16 JSB BCKSP      FETCH
C481 00636 017377R      JSB GEICR      FIRST
C482 00637 060065      LDA .10        UNUSED CHARACTER
C483 00640 035430R      ISZ NUMCK      RETURN
C484 00641 125430R      JMP NUMCK,I     VIA (P+2)
C485 00642 104200 NUM17 DLD SVOP1
      00643 015316R
C486 00644 017342R      JSB DDST
C487 00645 000022R      DEF SBPTR
C488 00646 025635R      JMP NUM16
C489**
C490*** NORMALIZE AND PACK FLOATING POINT NUMBER **
C491**
C492 00647 000000 .PACK NOP      MANTISSA IN (A) AND (B),
C493 00650 015742R      JSB NORML      EXPONENT IN EXP, (E) CLEARED
C494 00651 002103      CLE,SZA,RSS     ZERO RESULT?
C495 00652 125647R      JMP .PACK,I     YES
C496 00653 044073      ADB B177      NO, ROUND
C497 00654 002021      SSA,RSS        POSITIVE NUMBER?
C498 00655 005004      INB           YES, FINISH ROUND
C499 00656 103101      CLO
C500 00657 002040      SEZ           OVERFLOW FROM (B)?
C501 00660 002104      CLE,INA        YES, BUMP (A)
C502 00661 102301      SOS           OVERFLOW? (A=100000, B=0)
C503 00662 001200      RAL
C504 00663 002031      SSA,SLA,RSS     TWO HIGH BITS 1'S? (A=140000)
C505 00664 025667R      JMP PACK1     NO
C506 00665 002300      CCE           YES
C507 00666 001130      APS,SLA,ALS     SET (A) =100000 AND SKIP
C508 00667 001300 PACK1 RAR          COUNTERPART TO *-5

```

BASE PAGE SUBROUTINES

C509	00670	072777R	STA MBY10	SAVE (A)
C510	00671	060001	LDA 1	DELETE 8 LOW
C511	00672	010075	AND M256	ORDER BITS OF MANTISSA
C512	00673	070001	STA 1	SAVE LOWER MANTISSA
C513	00674	062042R	LDA EXP	FETCH EXPONENT
C514	00675	002040	SEZ	DECREMENT EXPONENT?
C515	00676	040052	ADA M1	YES
C516	00677	102201	SUC	NO, PRIOR OVERFLOW?
C517	00700	002004	INA	YES, INCREMENT EXPONENT
C518	00701	042172R	ADA B200	NO, EXPONENT
C519	00702	002020	SSA	UNDERFLOW?
C520	00703	025722R	JMP PACK3	YES
C521	00704	040075	ADA M256	NO, EXPONENT
C522	00705	002021	SSA, RSS	OVERFLOW?
C523	00706	025726R	JMP PACK4	YES
C524	00707	042172R	ADA B200	NO, RESTORE EXPONENT,
C525	00710	001200	RAL	POSITION SIGN,
C526	00711	010074	AND MSK0	MASK TO 8 BITS, AND
C527	00712	044000	ADB 0	COMBINE WITH LOW MANTISSA
C528	00713	062777R	LDA MBY10	RETRIEVE HIGH MANTISSA
C529	00714	052263R	CFA MNEG	
C530	00715	002001	RSS	NEGATIVE
C531	00716	125647R	JMP .PACK, I	
C532	00717	055264R	CPB MNEG+1	OVERFLOW?
C533	00720	025726R	JMP PACK4	YES
C534	00721	125647R	JMP .PACK, I	NO
C535	00722	015274R	PACK3 JSB ERROR	
C536	00723	002400	UNDER CLA	ZERO RESULT
C537	00724	005400	CLB	ON UNDERFLOW
C538	00725	125647R	JMP .PACK, I	
C539	00726	015274R	PACK4 JSB ERROR	
C540	00727	062777R	OVRER LDA MBY10	
C541	00730	015732R	JSB OVFLW	
C542	00731	125647R	JMP .PACK, I	
C543**				
C544*** LOAD INFINITY ON OVERFLOW **				
C545**				
C546	00732	000000	OVFLW NOP	
C547	00733	064051	LDB M2	LOAD
C548	00734	002020	SSA	APPROPRIATE
C549	00735	065175R	LDB B776	LOW MANTISSA
C550	00736	032222R	IOR INF	LOAD
C551	00737	002020	SSA	APPROPRIATE
C552	00740	062263R	LDA MNEG	HIGH MANTISSA
C553	00741	125732R	JMP OVFLW, I	
C554**				
C555*** NORMALIZE (A), (B), AND EXP **				
C556**				
C557	00742	000000	NORML NOP	SET
C558	00743	072777R	STA MBY10	LEFT-SHIFT
C559	00744	002400	CLA	COUNTER
C560	00745	072776R	SIA MPY	TO ZERO
C561	00746	062777R	LDA MBY10	
C562	00747	002003	SZA, RSS	ON
C563	00750	005002	SZB	ZERO

BASE PAGE SUBROUTINES

C564	00751	025757R	JMP NORM3	CLEAR
C565	00752	072042R	STA EXP	EVERYTHING
C566	00753	073066R	STA MANT1	STORE
C567	00754	077130R	STB MANT2	MANTISSA
C568	00755	125742R	JMP NORM1,I	AND RETURN
C569	00756	035776R	NORM2 ISZ MPY	COUNT LEFT SHIFTS
C570	00757	004066	NORM3 CLE,ELB	ROTATE (A) AND
C571	00760	001600	ELA	(B) LEFT INTO (E)
C572	00761	002061	SEZ,SSA,RSS	TWO HIGHEST BITS 0?
C573	00762	025756R	JMP NORM2	YES, + UNNORMALIZED
C574	00763	002060	SEZ,SSA	NO, TWO HIGHEST BITS 1?
C575	00764	025756R	JMP NORM2	YES, - UNNORMALIZED
C576	00765	001500	ERA	SHIFT TO
C577	00766	005540	ERB,CLE	NORMALIZE MANTISSA
C578	00767	073066R	STA MANT1	NO,
C579	00770	062776R	LDA MPY	COMPUTE
C580	00771	003004	CMA,INA	CORRECTED
C581	00772	042042R	ADA EXP	EXPONENT
C582	00773	072042R	STA EXP	VALUE
C583	00774	063066R	LDA MANT1	
C584	00775	025754R	JMP NORM1	
C585	00776	000000	MPY BSS 1	
C586**				
C587*** MULTIPLY UNPACKED NUMBER BY 10 **				
C588**				
C589	00777	000000	MBY10 NOP	
C590	01000	063066R	LDA MANT1	RETURN ON
C591	01001	002003	SZA,RSS	ZERO
C592	01002	125777R	JMP MBY10,I	MANTISSA
C593	01003	065042R	LDB EXP	MULTIPLY
C594	01004	044056	ADB .5	BY
C595	01005	075042R	STB EXP	8
C596	01006	067130R	LDB MANT2	LOAD MANTISSA
C597	01007	000065	CLE,ERA	DIVIDE
C598	01010	005500	ERB	BY
C599	01011	000065	CLE,ERA	4
C600	01012	005540	ERB,CLE	
C601	01013	047130R	ADB MANT2	DOUBLE
C602	01014	002040	SEZ	ADD TO
C603	01015	002004	INA	PRODUCE
C604	01016	043066R	ADA MANT1	1.25 * MANTISSA
C605	01017	002021	SSA,RSS	CORRECT
C606	01020	027025R	JMP *+5	
C607	01021	000065	CLE,ERA	ON
C608	01022	005500	ERB	
C609	01023	035042R	ISZ EXP	OVERFLOW
C610	01024	000000	NOP	
C611	01025	073066R	STA MANT1	
C612	01026	077130R	STB MANT2	
C613	01027	125777R	JMP MBY10,I	
C614**				
C615*** DIVIDE UNPACKED NUMBER BY 10 **				
C616**				
C617	01030	000000	DBY10 NOP	MULTIPLY BY DOUBLE-LENGTH TENTH
C618	01031	063066R	LDA MANT1	RETURN

BASE PAGE SUBROUTINES

C619	01032	002005	SZA,RSS	ON ZERO
C620	01033	127030R	JMP DBY10,I	MANTISSA
C621	01034	064051	LDB M2	ADD EXPONENT OF
C622	01035	046042R	ADB EXP	'TENTH' TO
C623	01036	076042R	STB EXP	MANTISSA EXPONENT
C624	01037	063130R	LDA MANT2	JUSTIFY
C625	01040	000065	CLE,ERA	LOWER MANTISSA
C626	01041	100200	MPY TENTH	MULTIPLY BY 63146 (ONE-TENTH)
	01042	000217R		
C627	01043	000065	CLE,ELA	SHIFT
C628	01044	005640	ELB,CLE	BACK
C629	01045	040001	ADA 1	ADD IN LOWER MANTISSA*
C630	01046	002040	SEZ	TENTH*(2)-16
C631	01047	006004	INB	AND ROUND
C632	01050	077130R	STB MANT2	TO 16 BITS
C633	01051	063066R	LDA MANT1	DO
C634	01052	100200	MPY TENTH	SAVE FOR
	01053	000217R		
C635	01054	000040	CLE	HIGH
C636	01055	040001	ADA 1	MANTISSA
C637	01056	043130R	ADA MANT2	(EFFECTIVELY) SUM
C638	01057	002040	SEZ	DOUBLE-LENGTH
C639	01060	006004	INB	PRODUCTS
C640	01061	077066R	STB MANT1	EXCHANGE
C641	01062	070001	STA 1	(A) AND (B)
C642	01063	063066R	LDA MANT1	REGISTERS
C643	01064	016742R	JSB NORML	NORMALIZE RESULT
C644	01065	127030R	JMP DBY10,I	

BASE PAGE SUBROUTINES

C646**

C647*** FIND AND STORE ONE-CHARACTER OPERATORS **

C648**

C649	01066	000000	SYMCK	NOP	CHARACTER IN (A)
C650	01067	076053R		STB COUNT	-(ENTRIES TO BE SEARCHED)
C651	01070	001727		ALF,ALF	POSITION
C652	01071	032141R		IOR .32	CHARACTER
C653	01072	167066R		LDB SYMCK,I	STARTING TABLE ENTRY - 2
C654	01073	037066R		ISZ SYMCK	SET RETURN ADDRESS
C655	01074	044055	SYMC1	ADB .2	UPDATE TABLE POINTER
C656	01075	150001		CPA 1,1	MATCH?
C657	01076	027104R		JNP SYMC2	
C658	01077	036053R		ISZ COUNT	NO, CONTINUE SEARCH?
C659	01100	027074R		JMP SYMC1	YES
C660	01101	001727		ALF,ALF	NO, RESTORE
C661	01102	010073		AND B177	CHARACTER
C662	01103	127066R		JMP SYMCK,I	AND EXIT
C663	01104	003400	SYMC2	CCA	GET
C664	01105	040001		ADA 1	INFORMATION
C665	01106	160000		LDA 0,1	WORD
C666	01107	012220R		AND OPMSK	AND
C667	01110	172022R		STA SBPTR,I	STORE IT
C668	01111	052204R		CPA B6000	::
C669	01112	027323R		JMP FSC14	
C670	01113	037066R		ISZ SYMCK	RETURN VIA
C671	01114	127066R		JMP SYMCK,I	(P+2)

C672**

C673*** FIND CALLED SUBROUTINE **

C674**

C675	01115	000000	FND5B	NOP	
C676	01116	076050R		STB TEMP2	SAVE SUBROUTINE NUMBER
C677	01117	066124R		LDB BGNTB	LOAD (B) WITH SUBROUTINE TABLE
C678	01120	056125R	FND51	CPB ENDTB	END OF TABLE?
C679	01121	016274R		JSB ERROR	YES
C680	01122	160001	CALER	LDA 1,1	NO, EXTRACT
C681	01123	010072		AND .65	SUBROUTINE NUMBER
C682	01124	052050R		CPA TEMP2	DESIRED ONE?
C683	01125	127115R		JMP FND5B,I	YES
C684	01126	044055		ADB .2	NO, MOVE TO
C685	01127	027120R		JMP FND51	NEXT TABLE ENTRY

C687*

C688*

C689*

C690*

C691*

C692*

C693*

C694*

C695*

C696 01130 000000 MDIM NOP

SUBROUTINE TO COMPUTE THE STORAGE REQUIRED BY AN
ARRAY WHOSE PACKED DIMENSIONS ARE IN A UPON ENTRY

THE SUBROUTINE RETURNS IN A THE NUMBER OF LOCATIONS
REQUIRED FOR THE SPECIFIED DIMENSIONS

= 2*DIM1*DIM2

BASE PAGE SUBROUTINES

0697	01131	070001	STA 1	STORE PACKED DIMS. TEMPORALILY
0698	01132	010074	AND MSK0	
0699	01133	073223R	STA .FLUN	STORE # OF COLUMNS
0700	01134	060001	LDA 1	
0701	01135	001727	ALF,ALF	
0702	01136	010074	AND MSK0	A = # OF ROWS
0703	01137	001000	ALS	DOUBLE FOR FLOATING POINT
0704	01140	100200	MPY .FLUN	COMPUTE 2*ROW*COLUMN
	01141	001223R		
0705	01142	002020	SSA	RESULT < 32768 ?
0706	01143	016274R	JSB ERROR	NO, ERROR DIMENSIONS TOO LARGE
0707	01144	006002 MER9	SZB	RESULT < 64K?
0708	01145	027143R	JMP *-2	NO, ERROR
0709	01146	127130R	JMP MDIM,1	YES, RETURN

BASE PAGE SUBROUTINES

C711**

C712*** ROUND A SUBSCRIPT TO AN INTEGER **

C713**

C714*

C715* RETURNS INTEGER IN [1,32767] (BIASED BY -1)

C716* OR EXITS TO ERROR.

C717*

C718	01147	000000	SBFIX	NOP	SUBSCRIPT IN (A) AND (B)
C719	01150	017160R	JSB	IFIX	24-BIT INTEGER?
C720	01151	027221R	JMP	E6-1	NO
C721	01152	002041	SEZ,RSS		YES, ROUND AND
C722	01153	044052	ADB M1		BIAS BY -1
C723	01154	002003	SZA,RSS		15-BIT
C724	01155	006020	SSB		POSITIVE INTEGER?
C725	01156	027221R	JMP	E6-1	NO
C726	01157	127147R	JMP	SBFIX,I	YES

C727**

C728*** INTEGERIZE FLOATING POINT NUMBER **

C729**

C730	01160	000000	IFIX	NOP	
C731	01161	102101	STO		
C732	01162	073377R	STA	GETCR	SAVE (A)
C733	01163	017223R	JSB	.FLUN	EXPONENT
C734	01164	002020	SSA		NON-NEGATIVE?
C735	01165	027210R	JMP	IFIX3	NO
C736	01166	042236R	ADA M16		YES, EXPONENT
C737	01167	002020	SSA		<= 15?
C738	01170	103101	CLD		YES
C739	01171	040043	ADA M8		EXPONENT
C740	01172	002021	SSA,RSS		<= 23?
C741	01173	127160R	JMP	IFIX,I	NO, ALL SIGNIFICANCE IS INTEGER
C742	01174	040043	ADA M8		MOVE BINARY POINT TO END OF (B)
C743	01175	073223R	STA	.FLUN	SAVE SHIFT COUNT
C744	01176	063377R	LDA	GETCR	RETRIEVE (A)
C745	01177	027204R	JMP	IFIX2	
C746	01200	000071	IFIX1	CLE,SLA,ARS	SHIFT (A) RIGHT
C747	01201	002200	CME		SHIFT (B)
C748	01202	004035	SLB,ERB		RIGHT
C749	01203	102101	STO		NOTE IF A 1 IS LOST
C750	01204	037223R	IFIX2	ISZ .FLUN	DONE?
C751	01205	027200R	JMP	IFIX1	NO
C752	01206	037160R	ISZ	IFIX	YES
C753	01207	127160R	JMP	IFIX,I	
C754	01210	063377R	IFIX3	LDA GETCR	RETRIEVE (A)
C755	01211	002120	CLE,SSA		TRUNCATE
C756	01212	003401	CCA,RSS		TO
C757	01213	002401	CLA,RSS		-1
C758	01214	007401	CCB,RSS		OR
C759	01215	006400	CLB		0
C760	01216	027206R	JMP	IFIX2+2	

```

C762**
C763*** TAKE ARITHMETIC INVERSE **
C764**
C765 01217 000000 ARINV NOP      NUMBER IN (A) AND (B)
C766 01220 015001X      FMP N1      MULTIPLY BY MINUS 1 TO TAKE INVERSE
      01221 000231R
C767 01222 127217R      JMP ARINV,I
C768**
C769*** UNPACK LOW WORD OF NUMBER **
C770**
C771 01223 000000 .FLUN NOP      WORD IN (B)
C001 01224 060001      LDA 1      (A) = (B)
C002 01225 010074      AND MSK0    EXTRACT EXPONENT IN (A)
C003 01226 007000      CMB      SUBTRACT OFF
C004 01227 044000      ADB 0      EXPONENT FROM
C005 01230 007000      CMB      MANTISSA IN (B)
C006 01231 000033      SLA,RAR     NEGATIVE EXPONENT?
C007 01232 032221R     IOR MSK4    YES, FILL IN LEADING BITS
C008 01233 127223R     JMP .FLUN,I NO
C009**
C010*** STACK (B) ON LOW-CORE STACK **
C011**
C012 01234 000000 SLWST NOP
C013 01235 035027R     ISZ LSTPT    ADVANCE 'LOW
C014 01236 062027R     LDA LSTPT    STACK' POINTER
C015 01237 052030R     CPA HSIPT    STACK OVERFLOW?
C016 01240 015274R E1   JSB ERROR   YES
C017 01241 175027R     STB LSTPT,I  NO, STACK (B)
C018 01242 127234R     JMP SLWST,I

```

BASE PAGE SUBROUTINES

```

C020**
C021*** BUMP HIGH STACK POINTER **
C022**
C023 01243 000000 BHSTP NOP
C024 01244 007400 CCB ADVANCE
C025 01245 045030R ADB HSTPT
C026 01246 075030R STB HSTPT POINTER
C027 01247 055027R CPB LSTPT OVERFLOW?
C028 01250 027240R JMP E1 YES
C029 01251 127243R JMP BHSTP,I NO
C030**
C031*** FETCH TOP OF STACK **
C032**
C033 01252 000000 STTOP NOP
C034 01253 017266R JSB UPCHK VALIDATE
C035 01254 017321R JSB RSCHK OPERAND
C036 01255 165030R LDB HSTPT,I SAVE
C037 01256 005020 SSB DATA ON DISK?
C038 01257 027263R JMP STT01 YES
C039 01260 104200 DLD 1,I LOAD NUMBER
      01261 100001
C040 01262 127252R JMP STTOP,I
C041 01263 060001 STT01 LDA B
C042 01264 017505R JSB FTHOP LOAD NUMBER
C043 01265 127252R JMP STTOP,I
C044**
C045*** VERIFY LEGITIMACY OF OPERAND **
C046**
C047 01266 000000 UPCHK NOP
C048 01267 165030R LDB HSTPT,I OPERAND ADDRESS TO (B)
C049 01270 005020 SSB DATA ON DISK?
C050 01271 017313R JSB GETOP YES, GET HIGH PART OF OPERAND
C051 01272 160001 LDA 1,I HIGH PART OF
C052 01273 052263R CPA MNEG OPERAND 100000B?
C053 01274 005005 INB,RSS YES
C054 01275 027304R JMP UPCH1 NO
C055 01276 005020 SSB DATA ON DISK?
C056 01277 017313R JSB GETOP YES, GET LOW PART OF OPERAND
C057 01300 160001 LDA 1,I LOW PART
C058 01301 052264R CPA MNEG+1 776B?
C059 01302 015274R .E8 JSB ERROR YES
C060 01303 044052 E8 ADB M1
C061 01304 055026R OPCH1 CPB TSTPT TEMPORARY OPERAND?
C062 01305 002001 RSS YES
C063 01306 127266R JMP UPCHK,I NO
C064 01307 062026R LDA TSTPT UNSTACK
C065 01310 040051 ADA M2 THE TEMPORARY
C066 01311 072026R STA TSTPT OPERAND
C067 01312 127266R JMP UPCHK,I EXIT WITH ADDRESS IN (B)
C068 01313 000000 GETOP NOP
C069 01314 060001 LDA B
C070 01315 012222R AND INF
C071 01316 017204R JSB DREAD
C072 01317 037313R ISZ GETOP
C073 01320 127313R JMP GETOP,I

```

BASE PAGE SUBROUTINES

C074**

C075*** ALLOT SPACE FOR INTERMEDIATE RESULT **

C076**

C077	01321	000000	RSCHK	NOP	
C078	01322	062026R	LDA	TSTPT	ALLOT
C079	01323	040055	ADA	.2	
C080	01324	072026R	STA	TSTPT	SPACE
C081	01325	040052	ADA	M1	OVERFLOW INTO
C082	01326	052012R	CPA	LSTAK	LOW-CORE STACK?
C083	01327	002001	RSS		YES
C084	01330	127321R	JMP	RSCHK,I	NO
C085	01331	062012R	LDA	LSTAK	SAVE
C086	01332	002004	INA		LOWER
C087	01333	072051R	STA	TEMP3	STACK BOUND
C088	01334	040064	ADA	.9	UPDATE
C089	01335	072012R	STA	LSTAK	STACK BOTTOM
C090	01336	062027R	LDA	LSTPT	SET
C091	01337	002004	INA		SOURCE
C092	01340	072050R	STA	TEMP2	ADDRESS
C093	01341	040064	ADA	.9	UPDATE
C094	01342	072027R	STA	LSTPT	STACK TOP
C095	01343	002004	INA		SET .DESTINATION
C096	01344	072052R	STA	TEMP4	ADDRESS
C097	01345	003004	CMA,INA		OVERFLOW
C098	01346	042030R	ADA	HSTPT	INTO
C099	01347	002020	SSA		HIGH-CORE STACK?
C100	01350	027240R	JMP	E1	YES
C101	01351	015367R	JSB	MVTOC	NO, MOVE
C102	01352	127321R	JMP	RSCHK,I	LOW-CORE STACK

C103**

C104*** CHECK FOR DIGIT **

C105**

C106	01353	000000	DIGCK	NOP	CHARACTER IN (A)
C107	01354	064000	LDB	0	
C108	01355	046246R	ADB	D72	ASCII 72B
C109	01356	005021	SSB,RSS		OR GREATER?
C110	01357	127353R	JMP	DIGCK,I	YES, RETURN WITH CHARACTER
C111	01360	044065	ADB	.10	NO, ASCII 60B
C112	01361	005020	SSB		OR GREATER?
C113	01362	127353R	JMP	DIGCK,I	NO
C114	01363	037353R	ISZ	DIGCK	YES, SET 'SUCCESS' EXIT,
C115	01364	060001	LDA	1	LOAD DIGIT INTO (A),
C116	01365	127353R	JMP	DIGCK,I	AND RETURN

C117**

C118*** CHECK FOR LETTER **

C119**

C120	01366	000000	LETCK	NOP	CHARACTER IN (A)
C121	01367	064000	LDB	0	
C122	01370	046252R	ADB	D133	ASCII 133B
C123	01371	005021	SSB,RSS		OR GREATER?
C124	01372	127366R	JMP	LETCK,I	YES, EXIT WITH CHARACTER IN (A)
C125	01373	046135R	ADB	.26	NO, ASCII 101B
C126	01374	005021	SSB,RSS		OR GREATER?
C127	01375	037366R	ISZ	LETCK	YES
C128	01376	127366R	JMP	LETCK,I	NO

```

C129*
C130*  ON END-OF-FILE CONDITION RETURN TO P+1 ELSE
C131*  RETURN TO P+2 WITH NON-BLANK CHARACTER IN (A)
C132*
C133  01377 000000  GETCR NOP
C134  01400 035020R  ISZ CCNT      ANY CHARACTERS LEFT?
C135  01401 002001  RSS
C136  01402 127377R  JMP GETCR,I  NO, END-OF-FILE EXIT
C137  01403 065017R  LDB BADDR  LOAD BUFFER ADDRESS
C138  01404 035017R  ISZ BADDR  UPDATE FOR NEXT TIME
C139  01405 004065  CLE,ERB   SET CHARACTER FLAG
C140  01406 160001  LDA 1,1   LOAD CURRENT BUFFER WORD
C141  01407 002041  SEZ,RSS   FIRST CHARACTER?
C142  01410 001727  ALF,ALF   YES, POSITION IT
C143  01411 010073  AND B177  MASK EXTRANEIOUS BITS
C144  01412 052271R  CPA BLANK  BLANK?
C145  01413 027400R  JMP GETCR+1 YES, FETCH NEXT CHARACTER
C146  01414 037377R  ISZ GETCR  UPDATE RETURN ADDRESS
C147  01415 127377R  JMP GETCR,I  AND EXIT
C148**
C149***  BACKSPACE OVER ONE CHARACTER **
C150**
C151  01416 000000  BCKSP NOP
C152  01417 003400  CCA          BACKSPACE
C153  01420 042020R  ADA CCNT      OVER
C154  01421 072020R  STA CCNT      LAST
C155  01422 003400  CCA          CHARACTER IN
C156  01423 042017R  ADA BADDR      INPUT
C157  01424 072017R  STA BADDR      BUFFER
C158  01425 127416R  JMP BCKSP,I
C159**
C160***  PRINT A NUMBER **
C161**
C162  01426 000000  ENOUT NOP
C163  01427 002300  CCE          SET SIGN FLAG TRUE
C164  01430 016041R  JSB NUMOT     OUTPUT THE NUMBER
C165  01431 062141R  LDA .32      OUTPUT
C166  01432 017477R  JSB OUTCR      A BLANK
C167  01433 065060R  LDB MLBX1+1  FIELD
C168  01434 045020R  ADB CCNT
C169  01435 005002  SZB          FULL?
C170  01436 027431R  JMP *-5      NO
C171  01437 127426R  JMP ENOUT,1
C172**
C173***  SPACE FOR A COMMA **
C174**
C175  01440 000000  EDELM NOP
C176  01441 065020R  LDB CCNT      NO, LOAD CHARACTER COUNT
C177  01442 005003  EDEL1 SZB,RSS  ZERO?
C178  01443 127440R  JMP EDELM,1  YES
C179  01444 046235R  ADB M15      NO, SUBTRACT ZONE WIDTH
C180  01445 005021  SSB,RSS      NEGATIVE RESULT?
C181  01446 027442R  JMP EDEL1    NO
C182  01447 077461R  STB OUTLN    YES, SAVE BLANK COUNT
C183  01450 062141R  LDA .32      FETCH BLANK

```

BASE PAGE SUBROUTINES

C184	01451	017477R	JSB OUTCR	OUTPUT
C185	01452	037461R	ISZ OUTLN	
C186	01453	027450R	JMP *-3	BLANKS
C187	01454	065020R	LDB CCNT	LINE
C188	01455	046251R	ADB M76	
C189	01456	005021	SSB,RSS	FULL?
C190	01457	017461R	JSB OUTLN	YES
C191	01460	127440R	JMP EDELM,1	

BASE PAGE SUBROUTINES

C193**

C194*** OUTPUT A COMPLETED LINE **

C195**

C196	01461	000000	OUTLN NOP	
C197	01462	062034R	LDA TYPE	FETCH 'CHARACTERS PRINTED' COUNTER
C198	01463	000010	SLA	CORRECT FOR START ON
C199	01464	002004	INA	ODD PRINT POSITION
C200	01465	042020R	ADA CCNT	OUTPUT
C201	01466	065016R	LDB .BUFA	A
C202	01467	017161R	JSB EXECW	LINE
C203	01470	065060R	LDB MLBX1+1	CORRECT
C204	01471	045020R	ADB CCNT	
C205	01472	075060R	STB MLBX1+1	MARKER
C206	01473	002400	CLA	RESET COUNT OF
C207	01474	072034R	STA TYPE	CHARACTERS PRINTED
C208	01475	016671R	JSB PRNIN	CLEAN UP
C209	01476	127461R	JMP OUTLN,1	

C210**

C211*** ADD A CHARACTER TO OUTPUT BUFFER **

C212**

C213	01477	000000	OUTCR NOP	CHARACTER IN (A)
C214	01500	073160R	STA IFIX	SAVE CHARACTER
C215	01501	035020R	ISZ CCNT	COUNT IT
C216	01502	065020R	LDB CCNT	FIRST CHARACTER
C217	01503	004010	SLB	OF BUFFER WORD?
C218	01504	035017R	ISZ BADDR	YES, MOVE TO FRESH WORD
C219	01505	162017R	LDA BADDR,I	LOAD BUFFER WORD
C220	01506	004010	SLB	SAVE
C221	01507	001727	ALF,ALF	OTHER
C222	01510	010075	AND M256	CHARACTER
C223	01511	033160R	IOR IFIX	ADD NEW CHARACTER
C224	01512	004010	SLB	POSITION
C225	01513	001727	ALF,ALF	WORD AND
C226	01514	172017R	STA BADDR,1	STORE IT
C227	01515	127477R	JMP OUTCR,I	

C228*

C229*

C230	00046	TEMP EQU TEMPS+1
C231	00047	TEMP1 EQU TEMPS+2
C232	00050	TEMP2 EQU TEMPS+3
C233	00051	TEMP3 EQU TEMPS+4
C234	00052	TEMP4 EQU TEMPS+5
C235	00053	COUNT EQU TEMPS+6
C236	00051	STEMP EQU TEMPS+4
C237	01066	MANT1 EQU SYMCK
C238	01130	MANT2 EQU MDIM
C239	01366	EXPON EQU LETCK
C240	01416	DPFLG EQU BCKSP
C241	01477	ARYAD EQU OUTCR
C242	00402	EOL EQU CONST

BASIC INTERPRETER CONTROL

C244*

C245***** BASIC INTERPRETER CONTROL *****

C246*

C247* THIS PROGRAM INTERPRETS THE SYSTEM COMMANDS AND PROVIDES

C248* I/O CONTROL FOR THE BASIC INTERPRETER. ALL USER

C249* COMMUNICATION IS DONE THRU THIS PROGRAM. USER RESPONSES ARE

C250* CHECKED FOR SYSTEM COMMANDS AND IF A VALID COMMAND IS

C251* DETECTED THIS PROGRAM INITIATES APPROPRIATE ACTION.

C252*

C253*

C254* DATA LOCAL TO MONITOR

C255*

C256 01516 001517R RDYA DEF READY

C257 01517 003412 READY OCT 3412

C258 01520 051105 ASC 2,READ

01521 040504

C259 01522 054415 OCT 54415

C260 01523 000216R LFEED DEF LF

C261 01524 000260R QMRKA DEF QMARK

C262 01525 004006R STOPA DEF STCMD

C263 01526 001715R CMNDA DEF CMNDS

C264 01527 000000 LTRAK BSS 1

C265 01530 000021 RC17 DEC 17

C266 01531 000000 ENTRY NOP

C267 01532 060147 LDA 147B

(LWA OF THIS PROG)

C268 01533 002004 INA

FIND START OF FREE CORE

C269 01534 072004C STA BGNPG

AND SAVE

C270 01535 072005C STA PG1

AS START OF BUFFER #1

C271 01536 003004 CMA,INA

C272 01537 040100 ADA 100B

(LWAM)

C273 01540 001100 ARS

FREE CORE / 2

C274 01541 005400 CLB

C275 01542 100400 DIV .128

DETERMINE # AVAIL SECTORS

01543 000172R

C276 01544 002003 SZA,RSS

ZERO SECTORS AVAIL ?

C277 01545 015274R JSB ERROR

YES, ERROR

C278 01546 100200 NOCOR MPY .128

DETERMINE BUFFER LENGTH

01547 000172R

FG 000

LN 0279 STA BFRL AND SAVE

C279 01550 072000 STA BFRL AND SAVE

C280 01551 042005C ADA PG1 DETERMINE START ADR OF BUFFER #2

C281 01552 072006C STA PG2 AND SAVE

C282 01553 015005X JSB EXEC

WORK AREA LIMITS

C283 01554 001561R DEF *+5

C284 01555 001530R DEF RC17

C285 01556 000011C DEF FTRAK

C286 01557 001527R DEF LTRAK

C287 01560 015135R DEF SIZE

C288 01561 002400 CLA

C289 01562 072001C STA PGPT1

C290 01563 072002C STA PGPT2

C291 01564 072000C STA PGPTR

C292*

BASIC INTERPRETER CONTROL

0293	01565	062002R	FLUSH	LDA	FWAM	
0294	01566	072007C		STA	PBUFF	SET PROGRAM BUFFER ADDRESS
0295	01567	072010C		STA	PBPTR	SET PROGRAM BUFFER POINTER
0296	01570	062141R		LDA	.32	INITIALIZE
0297	01571	072271R		STA	BLANK	DELETE CHARACTER FOR GETCR
0298	01572	002400		CLA		SET LINE NUMBER
0299	01573	072033R		STA	.LNUM	TO 0 INITIALLY
0300	01574	072015R		STA	OLDLN	
0301	01575	063741R	RDYPT	LDA	PRNTL	SET TO
0302	01576	073131R		STA	LISTR	COMMAND MODE
0303	01577	002400		CLA		
0304	01600	073667R		STA	DRQST	CLEAR DATA REQUEST FLAG
0305	01601	072023R		STA	TFLAG	CLEAR PHOTO READER INPUT FLAG
0306	01602	072024R		STA	TTYFL	CLEAR TTY TAPE FLAG
0307	01603	072150R		STA	.DFLG	CLEAR DISK INPUT FLAG
0308	01604	072025R		STA	ENFLG	
0309	01605	073264R		STA	XII	RESET EXECUTION EXIT FLAG
0310	01606	060063		LDA	.8	
0311	01607	067516R		LDB	RDYA	
0312	01610	017161R		JSB	EXECW	
0313	01611	063741R	PEXMK	LDA	PRNTL	SHIFT TO
0314	01612	073131R		STA	LISTR	COMMAND MODE
0315	01613	062023R		LDA	TFLAG	
0316	01614	002002		SZA		IS TAPE FLAG SET?
0317	01615	027744R		JMP	PTAPE+1	YES, GET RECORD FROM PHOTO RDR
0318	01616	062150R		LDA	.DFLG	IS DISK FLAG SET?
0319	01617	002002		SZA		
0320	01620	026052R		JMP	NXTRC	GET RECORD FROM DISK
0321	01621	027633R		JMP	GTICD	THIS ADDED (6-APR=71)

BASIC INTERPRETER CONTROL

```

C323 01622 067523R DATAI LDB LFEEED      LOAD ADDRESS OF LINE FEED
C324 01623 075040R      STB RSYM      STORE ADDRESS OF READY SYMBOL
C325 01624 062024R      LDA TTYFL      TTY TAPE
C326 01625 002002      SZA      INPUT?
C327 01626 027636R      JMP GTRCD+3      YES, SUPPRESS LINE FEED
C328 01627 003400      CCA      NO
C329 01630 065040R      LDB RSYM      LOAD LF OR '?' ADDRESS
C330 01631 017161R      JSB EXECW      PRINT LF OR '?', NO CR-LF
C331 01632 027636R      JMP *+4
C332 01633 060050      GTRCD LDA M3      PRINT ! TO SAY BASIC
C333 01634 067712R      LDB XPNT      IS READY
C334 01635 017161R      JSB EXECW
C335 01636 062247R      LDA M72
C336 01637 065016R      LDB .BUFA
C337 01640 017124R      JSB EXECR      GET RECORD FROM TTY
C338*
C339 01641 003021      RPRCS CMA,SSA,RSS      SET A=-1-# CHARS AND CHECK FOR
C340 01642 015274R      JSB ERROR      RECORD TOO LONG
C341 01643 072020R      RTLE STA CCNT      -1-# CHARACTERS < 0, SET CCNT
C342 01644 062016R      LDA .BUFA      LOAD BUFFER ADDRESS
C343 01645 000066      CLE,ELA      SHIFT LEFT, LEAST BIT USED AS
C344 01646 072017R      STA BADDR      ODD/EVEN FLAG
C345 01647 017377R      JSB GETCR      FETCH FIRST CHARACTER
C346 01650 027622R      JMP DATAI      NULL RECORD, INPUT AGAIN
C347 01651 067667R      LDB DRQST
C348 01652 005003      SZB,RSS      DATA REQUEST?
C349 01653 027674R      JMP CKRCD      NO DATA REQUEST, GO CHECK RECORD
C350 01654 052167R      CPA S      ASCII S FIRST CHARACTER?
C351 01655 027760R      JMP STOP      ASSUME STOP REQUESTED
C352 01656 002400      CLA      LINE
C353 01657 017161R      JSB EXECW      FEED
C354 01660 017416R      JSB BCKSP      BACKSPACE
C355 01661 062226R      LDA RMODE      RETURN TO
C356 01662 073131R      STA LISTR      RUN MODE
C357 01663 067667R      LDB DRQST
C358 01664 002400      CLA
C359 01665 073667R      STA DRQST      CLEAR DATA REQUEST FLAG
C360 01666 124001      JMP 1,I      GO TO DATA REQUEST CALLING POINT
C361*
C362*
C363* THIS SECTION REQUESTS DATA INPUT
C364*
C365 01667 000900      DRQST NOP      EXIT/ENTRY AND FLAG
C366 01670 067741R      LDB PRNTL      SHIFT TO
C367 01671 077131R      STB LISTR      COMMAND MODE
C368 01672 067524R      LDB QMRKA
C369 01673 027623R      JMP DATAI+1      PRINT '?' AND WAIT

```

BASIC INTERPRETER CONTROL

C371*
C372* THIS SECTION CHECKS RECORD FOR SYSTEM COMMANDS.
C373*
C374 01674 066021R CKRCD LDB SBUFA
C375 01675 076022R STB SBPTR INITIALIZE SYNTAX BUFFER POINTER
C376 01676 172022R STA SBPTR,I PUT FIRST CHAR IN SYNTAX BUFFER
C377 01677 017366R JSB LETCK IS CHARACTER A LETTER
C378 01700 025155R JMP SYNTAX NO, TRY SYNTAX
C379*
C380 01701 062105R LDA TBLAD LOAD SYS CMND TABLE START POINT
C381 01702 064041 LDB M10 LOOK FOR A
C382 01703 015242R JSB IBSRH SYSTEM COMMAND
C383 01704 015274R JSB ERROR NOT A VALID COMMAND
C384*
C385 01705 INVSC EQU * INVALID CMND ERROR REFERENCE
C386 01705 060065 LDA .10
C387 01706 042053R ADA COUNT
C388 01707 043526R ADA CMNDA ADD START ADDR. OF CMND ROUTINES
C389 01710 164000 LDB 0,I
C390 01711 124001 JMP 1,I
C391*
C392 01712 001713R XPNT DEF *+1
C393 01713 020440 ASC 2,1 _
01714 057440

BASIC INTERPRETER CONTROL

```

C395*
C396* THIS SETS UP AND EXECUTES THE SYSTEM COMMANDS
C397*
C398 01715 CMNDS EQU * COMMAND LIST REFERENCE
C399*
C400*****
C401*** THE FOLLOWING TABLE ADDED 13-APR-71 ***
C402*****
C403*
C404 01715 001727R DEF RUN
C405 01716 001730R DEF SCRTH
C406 01717 001731R DEF TLIST
C407 01720 001733R DEF PLIST
C408 01721 001743R DEF PTAPE
C409 01722 001760R DEF STOP
C410 01723 001772R DEF TAPE
C411 01724 001774R DEF BYEC
C412 01725 001777R DEF PUNC
C413 01726 002003R DEF LOAD
C414*
C415*****
C416 01727 027256R RUN JMP MFASE GO TO RUN ENTRY POINT
C417*
C418 01730 027565R SCRTH JMP FLUSH SCRATCH CURRENT PROGRAM
C419 01731 063741R TLIST LDA PRNTL LIST PROGRAM, TFLAG = 0
C420 01732 005401 CLB,RSS
C421 01733 063740R PLIST LDA PUNCL PUNCH PROGRAM TFLAG # 0
C422 01734 073131R STA LISTR SET DRIVER ADDRESS
C423 01735 005400 CLB
C424 01736 076023R LPUNF STB TFLAG
C425 01737 025641R JMP LIST GO TO LIST ENTRY POINT
C426 01740 013242R PUNCL DEF EXECPT
C427 01741 013161R PRNTL DEF EXECW
C428 01742 013211R TLPUN DEF LPUN
C429 01743 000000 PTAPE NOP
C430 01744 062025R LDA ENFLG
C431 01745 002003 SZA,RSS
C432 01746 027751R JMP *+3
C433 01747 017152R JSB EXECS
C434 01750 027575R JMP RDYPT
C435 01751 062163R LDA .72
C436 01752 065016R LDB .BUFA
C437 01753 017137R JSB EXECT GET RECORD FROM PHOTO READER
C438 01754 002003 PRERR SZA,RSS YES
C439 01755 027744R JMP PTAPE+1 NULL RECORD
C440 01756 072023R STA TFLAG SET FLAG # 0
C441 01757 027641R JMP RPRCS GO PROCESS RECORD
C442*
C443* STOP COMMAND SERVICE
C444*
C445 01760 067741R STOP LDB PRNTL SHIFT TO
C446 01761 077131R SIB LISTR COMMAND MODE
C447 01762 062263R LDA MNeg
C448 01763 002006 INA,SZA
C449 01764 027763R JMP *-1 DELAY FOR 100 MILLISECONDS

```

C450	01765	017161R	JSB EXECW	C-R AND L-F
C451	01766	060057	LDA .4	
C452	01767	067525R	LDB STOPA	
C453	01770	017161R	JSB EXECW	PRINT 'STOP'
C454	01771	027575R	JMP RDIPT	
C455*				
C456**	SET	LINE FEED SUPPRESSION		
C457*				
C458	01772	072024R	TAPE STA TTYFL	SET TO 'TAPE' MODE
C459	01773	027633R	JMP GTRCD	
C460	01774	015005X	BYEC JSB EXEC	
C461	01775	001777R	DEF *+2	
C462	01776	000061	DEF .6	
C463	01777	063742R	PUNC LDA TLPUN	
C464	02000	073131R	STA LISTR	
C465	02001	005404	CLB,INB	
C466	02002	027736R	JMP LPUNF	
C467**				
C468***		LOAD PROGRAM FROM DISK FILE		
C469**				
C470	02003	000000	LOAD NOP	
C471	02004	062152R	LDA SPSP	SET FILE NAME
C472	02005	072137R	STA FNAME	TO
C473	02006	072140R	STA FNAME+1	ALL
C474	02007	072141R	STA FNAME+2	BLANKS
C475	02010	062136R	LDA FADR	LOAD FILE NAME ADDRESS
C476	02011	001000	ALS	DOUBLE TO USE AS BYTE POINTER
C477	02012	073160R	STA PNTR	
C478	02013	017377R	LD1 JSB GETCR	FETCH FILE NAME CHARACTER
C479	02014	025032R	JMP LD2	END OF INPUT FILE, EXIT
C480	02015	072045R	STA .TEMP	SAVE CHARACTER
C481	02016	067160R	LDB PNTR	GET BYTE POINTER
C482	02017	037160R	ISZ PNTR	BUMP POINTER
C483	02020	004065	CLE,ERB	MAKE WORD ADDRESS AND SET
C484	02021	160001	LDA B,I	L/R FLAG THEN GET WORD
C485	02022	002041	SEZ,RSS	(E)=0 FOR LEFT CHAR.
C486	02023	001727	ALF,ALF	POSITION CHARACTER TO LEFT BYTE
C487	02024	010075	AND M256	MASK RIGHT BYTE AND ADD
C488	02025	032045R	IOR .TEMP	NEW CHARACTER
C489	02026	002041	SEZ,RSS	(E)=0 SWAP HALVES
C490	02027	001727	ALF,ALF	OF A
C491	02030	170001	STA B,I	AND SAVE
C492	02031	025013R	JMP LD1	DO IT AGAIN
C493	02032	015005X	LD2 JSB EXEC	SEARCH DISK FOR REQUESTED
C494	02033	002037R	DEF *+4	FILE
C495	02034	000131R	DEF .18	
C496	02035	002137R	DEF FNAME	
C497	02036	002142R	DEF NSECT	
C498	02037	062142R	LDA NSECT	IF FILE NOT FOUND
C499	02040	002003	SZA,RSS	PRINT ERROR MESSAGE
C500	02041	016274R	JSB ERKOR	
C501	02042	062142R	NOFIL LDA NSECT	
C502	02043	002004	INA	
C503	02044	003004	CMA,INA	MAKE -(SECTORS+1)
C504	02045	072145R	STA ENTLN	

BASIC INTERPRETER CONTROL

C505	02046	062146R	LDA DNBUF	FORCE READ OF FIRST SECTOR
C506	02047	072144R	STA MLUCN	
C507	02050	002400	CLA	SET RELATIVE SECTOR
C508	02051	072143R	STA RLSEC	TO ZERO
C509	02052	062025R	NXTRC LDA ENFLG	END STATEMENT ENCOUNTERED?
C510	02053	002002	SZA	
C511	02054	027575R	JMP RDYPT	YES, RETURN TO READY
C512	02055	015061R	JSB DISCR	NO, READ NEXT SOURCE STATEMENT
C513	02056	027575R	JMP RDYPT	END OF FILE, RETURN TO READY
C514	02057	072150R	STA .DPLG	SET DISK INPUT FLAG
C515	02060	027641R	JMP RPRCS	GO PROCESS RECORD
C516**				
C517**				
C518***			READ A SOURCE STATEMENT FROM THE DISC OUTPUT BUFFER	
C519***			WHEN THE BUFFER IS EXHAUSTED ANOTHER SECTOR IS READ	
C520***			FROM THE DISK.	
C521**				
C522**				
C523	02061	000000	DISCR NOP	
C524	02062	016105R	JSB DRORD	GO GET THE 1ST WORD(= # OF WORDS)
C525	02063	001727	ALF, ALF	
C526	02064	010072	AND B77	
C527	02065	072153R	STA STLEN	SAVE STATEMENT LENGTH
C528	02066	002003	SZA, RSS	IS IT=0
C529	02067	125061R	JMP DISCR, I	YES, END OF FILE EXIT
C530	02070	003004	CMA, INA	
C531	02071	072154R	STA TERMS	SET WORD COUNTER
C532	02072	062016R	LDA .BUFA	SET START
C533	02073	072151R	STA SBUFF	OF 'BUFFER'
C534	02074	016105R	DSCR2 JSB DRORD	
C535	02075	172151R	STA SBUFF, I	
C536	02076	035151R	ISZ SBUFF	
C537	02077	035154R	ISZ TERMS	HAVE WE FINISHED THE STATEMENT?
C538	02100	025074R	JMP DSCR2	NO
C539	02101	035061R	ISZ DISCR	YES- RETURN
C540	02102	062153R	LDA SILEN	GET STATEMENT LENGTH
C541	02103	001000	ALS	MAKE CHARACTERS
C542	02104	125061R	JMP DISCR, I	
C543**				
C544***			READ A WORD FROM DISK INPUT BUFFER	
C545***			WHEN THE BUFFER IS EXHAUSTED A NEW SECTOR	
C546***			IS READ FROM THE DISK	
C547**				
C548	02105	000000	DRORD NOP	
C549	02106	062144R	LDA MLUCN	
C550	02107	052146R	CPA DNBUF	POINTER AT END OF BUFFER
C551	02110	002001	RSS	YES
C552	02111	025120R	JMP DWDS	NO- GET THE NEXT WORD
C553	02112	062147R	LDA DBUFF	
C554	02113	072144R	STA MLUCN	SET POINTER TO START OF BUFFER
C555	02114	035145R	ISZ ENTLN	ARE WE AT THE END?
C556	02115	002001	RSS	
C557	02116	125061R	JMP DISCR, I	YES, TAKE P+1 EXIT FROM DISCR
C558	02117	016123R	JSB RSECT	
C559	02120	162144R	DWDS LDA MLUCN, I	GET NEXT WORD FROM BUFFER

BASIC INTERPRETER CONTROL

```

C560 02121 036144R      ISZ MLOCN      BUMP POINTER
C561 02122 126105R      JMP DRORD,I    EXIT TO DISCR
C562**
C563***      READ A SECTOR FROM DISK FILE
C564**
C565 02123 000000      RSECT NOP
C566 02124 015005X      JSB EXEC
C567 02125 002134R      DEF **7      READ NEXT SECTOR
C568 02126 000130R      DEF .14
C569 02127 000055      DEF .2
C570 02130 013457R      DEF FINIS+100      OF
C571 02131 000172R      DEF .128
C572 02132 002137R      DEF FNAME      DISK FILE
C573 02133 002143R      DEF RLSEC
C574 02134 036143R      ISZ RLSEC      INCREMENT RELATIVE SECTOR NUMBER
C575 02135 126123R      JMP RSECT,I
C576*
C577*
C578 01160      PNTR EQU IFIX      BYTE POINTER
C579 00045      .TEMP EQU TEMPS      TEMPORARY CHARACTER STORAGE
C580 02136 002137R      FADR DEF **1      FILE NAME ADDRESS
C581 02137 000000      FNAME BSS 3      HOLDS FILE NAME
C582 02142 000000      NSECT BSS 1      NUMBER OF SECTORS IN FILE
C583 02143 000000      RLSEC BSS 1      RELATIVE FILE SECTOR TO BE READ
C584 02144 000000      MLOCN BSS 1      SECTOR BUFFER POINTER
C585 02145 000000      ENTLN BSS 1      SECTOR COUNTER
C586 02146 013657R      DNBUF DEF FINIS+228      END OF SECTOR BUFFER
C587 02147 013457R      DBUFF DEF FINIS+100      START OF SECTOR BUFFER
C588 02150 000000      .DFLG BSS 1      DISK I/O FLAG
C589 02151 000000      SBUFF BSS 1      INPUT BUFFER POINTER
C590 02152 020040      SPSP OCT 20040      (BLANK-BLANK)
C591 02153 000000      STLEN BSS 1      HOLDS STATEMENT WORD COUNT
C592 02154 000000      TERMS BSS 1      WORD COUNTER

```

CHECK SYNTAX AND TRANSLITERATE

```

C594*
C595* *****
C596***
C597*** CHECK SYNTAX OF STATEMENT ***
C598***
C599* *****
C600*
C601**
C602*** DETERMINE SEQUENCE NUMBER **
C603**
C604 02155 065033R SYNTAX LDB .LNUM GET OLD SEQUENCE #
C605 02156 007004 CMB,INB MAKE NEGATIVE
C606 02157 046015R ADB OLDLN IS IT HIGHEST OLD SEQ #
C607 02160 006020 SSB
C608 02161 016223R JSB SAVIT YES, LDB .LNUM
C609 02162 065015R LDB OLDLN NO
C610 02163 075015R STB OLDLN AND SAVE
C611 02164 016400R JSB INTCK RECORD
C612 02165 000256R DEF MAXSN SEQUENCE NUMBER
C613 02166 036022R ISZ SBPTR SAVE SPACE FOR LENGTH WORD
C614 02167 075033R STB .LNUM SAVE LINE NUMBER
C615 02170 065021R LDB SBUFA SET
C616 02171 006004 INB TEMP TO
C617 02172 075046R STB TEMP (SBUFA)+1
C618**
C619*** DETERMINE STATEMENT TYPE **
C620**
C621 02173 050065 CPA .10 NULL STATEMENT?
C622 02174 026466R JMP DLSTM YES, DELETE IT
C623 02175 172022R STA SBPTR,I NO, RECORD NEXT CHARACTER
C624 02176 062106R LDA STIYP PRINT-TABLE ADDRESS
C625 02177 065273R LDB TBLX -(NUMBER OF ENTRIES)
C626 02200 016242R JSB TBSRH FIND STATEMENT TYPE
C627 02201 016274R JSB ERROR NOT FOUND
C628 02202 064042 SYNE1 LDB M9 SET MULTIPLE STORE
C629 02203 075300R STB MSFLG TO FALSE
C630 02204 065010C LDB PBPTR NULL
C631 02205 056007C CPB PBUFF PROGRAM?
C632 02206 002001 RSS YES
C633 02207 026213R JMP SYNT1 NO
C634 02210 065002R LDB FWAM INSURE NO
C635 02211 075007C STB PBUFF SPURIOUS COMMON
C636 02212 076010C STB PBPTR EXISTS
C637 02213 065013R SYNT1 LDB ASBTB
C638 02214 075045R STB TEMPS SET S-STACK POINTER
C639 02215 006400 CLB SET DEFINE FLAG
C640 02216 075302R STB DFLAG TO FALSE
C641 02217 075304R STB PFLAG SET PARAMETER FLAG TO FALSE
C642 02220 SYNT EQU *
C643 02220 001727 ALF,ALF COMPUTE
C644 02221 042101R ADA STBAS SYNTAX ROUTINE AND
C645 02222 124000 JMP 0,I BRANCH TO IT
C646 02223 000000 SAVIT NOP
C647 02224 065033R LDB .LNUM
C648 02225 036223R ISZ SAVIT

```

CHECK SYNTAX AND TRANSLITERATE

```

C649 02226 125223R      JMP SAVIT,I
C650**
C651*** SINGLE CHARACTER AND/OR FORMULA OPERATORS **
C652**
C653 02227 000400  QUOTE OCT 400  ::
C654 02230 021040      ASC 1,"
C655 02231 001000  COMMA OCT 1000 ::
C656 02232 025040      ASC 1,,
C657 02233 001400  SMCLN OCT 1400 ::
C658 02234 035440      ASC 1,;
C659 02235 002001  RPARN OCT 2001 ::
C660 02236 024440      ASC 1,)
C661 02237 002401  RBRAC OCT 2401
C662 02240 055440      ASC 1,]
C663 02241 003002  SCMMA OCT 3002 ::
C664 02242 025040      ASC 1,,
C665 02243 003402  ASSUP OCT 3402 ::
C666 02244 035440      ASC 1,=
C667 02245 004007  PLUS  OCT 4007 ::
C668 02246 025440      ASC 1,+
C669 02247 004407  MINUS OCT 4407 ::
C670 02250 025440      ASC 1,-
C671 02251 005010  TIMES OCT 5010 ::
C672 02252 025040      ASC 1,*
C673 02253 005410  DIV   OCT 5410 ::
C674 02254 027440      ASC 1,/
C675 02255 006012  EXPS  OCT 6012 ::
C676 02256 057040      ASC 1,^
C677 02257 006405  GTR   OCT 6405 ::
C678 02260 037040      ASC 1,>
C679 02261 007005  LSS   OCT 7005
C680 02262 035040      ASC 1,<
C681 02263 007405  UNEQL OCT 7405 ::
C682 02264 021440      ASC 1,#
C683 02265 010005  EQUAL OCT 10005 ::
C684 02266 035440      ASC 1,=
C685 02267 010411  UNMIN OCT 10411 ::
C686 02270 025440      ASC 1,-
C687 02271 011020  LBRAC OCT 11020 ::
C688 02272 055440      ASC 1,[
C689 02273 011420  LPARN OCT 11420 ::
C690 02274 024040      ASC 1,(
C691 02275 012011  UPLUS OCT 12011 ::
C692 02276 025440      ASC 1,+
C693 02277 012403  GRUP  OCT 12403 ::
C694 02300 000000  MSFLG NOP
C695 02301 013004  ANDOP OCT 13004 ::
C696 02302 000000  DFLAG NOP
C697 02303 013411  NOTOP OCT 13411 ::
C698 02304 000000  PFLAG NOP
C699 02305 014005  GTREQ OCT 14005 ::
C700 02306 000000  UFLAG NOP
C701 02307 014405  LSSEQ OCT 14405
C702*
C703***

```

CHECK SYNTAX AND TRANSLITERATE

C704** LET STATEMENT SYNTAX **

C705*** ***

C706*

C707	02310	005400	LETS	CLB	SET "NO STORE" FLAG (B)=0
C708	02311	077534R		S1B	SFLAG
C709	02312	069045		LDA	M8
C710	02313	072300R		S1A	MSFLG
C711	02314	017120R		JSB	FSC
C712	02315	057534R		CFB	SFLAG
C713	02316	015274R		JSB	ERROR
C714	02317		SYNE2	EQU	*

C715**

C716*** CHECK FOR END OF STATEMENT **

C717**

C718	02317	050065	EOST	CPA	.10
C719	02320	025476R		JMP	ACTST
C720	02321	025071R		JMP	NOEOF

					END-OF-STATEMENT?
					YES ACCEPT STATEMENT
					NO, ILLEGAL CHARACTER

CHECK SYNTAX AND TRANSLITERATE

```

C722*
C723***          ***
C724** DIM STATEMENT SYNTAX **
C725***          ***
C726*
C727 02322 036302R DIMS ISZ DFLAG SET DFLAG TO TRUE
C728 02323 017534R JSB ARRYS CHECK AN ARRAY
C729 02324 026476R JMP ACTST DONE
C730 02325 026323R JMP DIMS+1 WAS A COMMA, CONTINUE
C731*
C732***          ***
C733** COM STATEMENT SYNTAX **
C734***          ***
C735*
C736 02326 065010C COMS LDB PBPTR HAS PROGRAM BUFFER
C737 02327 056002R CPB FWAM BEEN MOVED?
C738 02330 002001 RSS NO
C739 02331 016274R JSB ERROR YES, ILLEGAL COM
C740 02332 076054R SYNE3 STB TEMPS+7 SET ARRAY POINTER
C741 02333 036302R ISZ DFLAG SET DEFINE FLAG TO TRUE
C742 02334 003400 COMS1 CCA SET COMMON FLAG
C743 02335 072304R STA PFLAG TO TRUE
C744 02336 017534R JSB ARRYS CHECK AN ARRAY
C745 02337 002001 RSS DONE
C746 02340 026334R JMP COMS1 MORE ARRAYS
C747 02341 065054R LDB TEMPS+7 FETCH UPDATED POINTER
C748 02342 076007C STB PBUFF SET PROGRAM BUFFER ADDRESS
C749 02343 076010C STB PBPTR SET PROGRAM BUFFER POINTER
C750 02344 026476R JMP ACTST EXIT
C751*
C752***          ***
C753** DEF STATEMENT SYNTAX **
C754***          ***
C755*
C756 02345 017645R DEFS JSB LTR
C757 02346 026356R JMP SYNE4 FIRST
C758 02347 062047R LDA TEMP1
C759 02350 001727 ALF,ALF TWO CHARACTERS
C760 02351 032050R IOR TEMP2
C761 02352 052257R CPA FN 'FN'?
C762 02353 002001 RSS YES
C763 02354 026356R JMP SYNE4 NO
C001 02355 017645R JSB LTR LETTER FOLLOWS?
C002 02356 016274R SYNE4 JSB ERROR NO
C003 02357 062047R LDA TEMP1 YES, RECORD A
C004 02360 065156R LDB .50 **:FUNCTION
C005 02361 017660R JSB STRUP NAME
C006 02362 062050R LDA TEMP2 RETRIEVE CHARACTER
C007 02363 017671R JSB LPCK LEFT PARENTHESIS?
C008 02364 032263R IOR FLGHT YES, SET FORMAL
C009 02365 172022R STA SBPTR,I PARAMETER BIT
C010 02366 017562R JSB VARUP FETCH SIMPLE VARIABLE
C011 02367 000000 NOP NONE FOUND
C012 02370 016274R JSB ERROR SUBSCRIPTED VARIABLE FOUND
C013 02371 017701R SYNE5 JSB RPCK RECORD A RIGHT PARENTHESIS

```

CHECK SYNTAX AND TRANSLITERATE

C014	02372	007400	CCB	ASSIGNMENT
C015	02373	017066R	JSB SYMCK	
C016	02374	002242R	DEF ASSOP-1	OPERATOR?
C017	02375	015274R	SYNE6 JSB ERROR	NO
C018	02376	060051	LDA M2	YES,
C019	02377	042022R	ADA SBPTR	RETRIEVE
C020	02400	160000	LDA 0,I	PARAMETER
C021	02401	012176R	AND MSK1	AND
C022	02402	072304R	STA PFLAG	SAVE IT
C023	02403	017120R	JSB FSC	FETCH DEFINING FORMULA
C024	02404	025317R	JMP EOST	END-OF-STATEMENT TEST
C025*				
C026***			***	
C027**	REM	STATEMENT SYNTAX	**	
C028	02405	102501	REMS LIA 1	IS SW 13 SET?
C029	02406	012210R	AND B20K	
C030	02407	002002	SZA	
C031	02410	027511R	JMP PEXMK	YES, DELETE THE REM
C032	02411	060055	LDA .10	
C033	02412	015437R	JSB CHRST	
C034	02413	025476R	JMP ACTST	
C035	02414	027511R	JMP PEXMK	
C036*				
C037***			***	
C038**	IF	STATEMENT SYNTAX	**	
C039***			***	
C040*				
C041	02415	017120R	IFS JSB FSC	GET DECISION FORMULA
C042	02416	172022R	STA SBPTR,I	TABLE
C043	02417	062116R	LDA ATHEN	SEARCH
C044	02420	007400	CCB	FOR
C045	02421	015242R	JSB TBSRH	'THEN'
C046	02422	015274R	JSB ERROR	NOT FOUND
C047	02423		SYNE7 EQU *	GET STATEMENT LABEL NUMBER
C048	02423	035022R	ISZ SBPTR	
C049	02424	017377R	JSB GETCR	
C050	02425	025070R	JMP EOF	
C051	02426	017366R	JSB LETCK	IS NEXT CHAR. A LETTER?
C052	02427	025436R	JMP IFS1	NO ADD GOTO CODE TO BUFFER
C053	02430	172022R	STA SBPTR,I	YES, SAVE CHAR TEMPORARILY
C054	02431	062106R	LDA STIYP	FIND 'CONDITIONAL EXEC'
C055	02432	065273R	LDB TBLX	STATEMENT TYPE
C056	02433	015242R	JSB TBSRH	
C057	02434	015274R	JSB ERROR	NOT FOUND, ERROR!
C058	02435	025220R	SYE1 JMP SYNT	FOUND, BRANCH TO SYNTAX ROUTINE
C059	02436	017416R	IFS1 JSB BCKSP	
C060	02437	062044R	LDA GOTO	REPLACE SEQUENCE NUMBER BY
C061	02440	040047	ADA M4	GO TO SEQUENCE NUMBER
C062	02441	172022R	STA SBPTR,I	
C063*				
C064***			***	
C065**	GO TO AND GOSUB	STATEMENT SYNTAX	**	
C066***			***	
C067	02442	017120R	GOTOS JSB FSC	FETCH AND RECORD FORMULA
C068	02443	025317R	JMP EOST	END-OF-STATEMENT TEST

CHECK SYNTAX AND TRANSLITERATE

```

C069*
C070*
C071***          ***
C072**  FOR STATEMENT SYNTAX **
C073***          ***
C074*
C075  02444 017562R FORS JSB VAROP    FETCH SIMPLE VARIABLE
C076  02445 000000      NOP           NONE FOUND
C077  02446 015274R SYNE8 JSB ERROR    SUBSCRIPTED VARIABLE FOUND
C078  02447 007400      CCB
C079  02450 017066R      JSB SYMCK     ASSIGNMENT
C080  02451 002242R      DEF ASSOP-1  OPERATOR?
C081  02452 025375R      JMP SYNE6     NO
C082  02453 017120R      JSB FSC       YES, FETCH INITIAL VALUE FORMULA
C083  02454 172022R      STA SBPTR,1   LOOK
C084  02455 062117R      LDA ATU       FOR
C085  02456 007400      CCB           THE
C086  02457 015242R      JSB TBSRH     'TO'
C087  02460 015274R      JSB ERROR     MISSING
C088  02461 017120R SYNE9 JSB FSC       GET LIMIT FORMULA
C089  02462 050065      CPA .10        END-OF-STATEMENT?
C090  02463 025476R      JMP ACTST     YES
C091  02464 007400      CCB           NO, ERASE
C092  02465 045022R      ADB SBPTR     ZERO
C093  02466 075022R      STB SBPTR     WORD
C094  02467 172022R      STA SBPTR,1   NOW
C095  02470 062120R      LDA ASTEP     LOOK
C096  02471 007400      CCB           FOR
C097  02472 015242R      JSB TBSRH     THE 'STEP'
C098  02473 015274R      JSB ERROR     MISSING
C099  02474 017120R SYE10 JSB FSC       GET STEP SIZE FORMULA
C100  02475 025317R      JMP EOST     END-OF-STATEMENT TEST
C101*
C102***          ***
C103**  NEXT STATEMENT SYNTAX **
C104***          ***
C105*
C106  02476 017562R NXTS JSB VAROP    FETCH SIMPLE VARIABLE
C107  02477 000000      NOP           NONE FOUND
C108  02500 025446R      JMP SYNE8     SUBSCRIPTED VARIABLE FOUND
C109  02501 025317R      JMP EOST     END-OF-STATEMENT TEST
C110*
C111***          ***
C112**  END, STOP, RESTORE, RETURN STATEMENT SYNTAX **
C113***          ***
C114*
C115  02502 062023R ENDS  LDA TFLAG
C116  02503 002002      SZA
C117  02504 035025R      ISZ ENFLG
C118  02505 062150R      LDA .DPLG     IS INPUT FROM DISK?
C119  02506 002002      SZA
C120  02507 035025R      ISZ ENFLG     YES, SET FLAG ON END STATEMENT
C121  02510 035022R      ISZ SBPTR
C122  02511 017377R      JSB GETCR     END-OF-STATEMENT?
C123  02512 025476R      JMP ACTST     YES

```

CHECK SYNTAX AND TRANSLITERATE

C124	02513	025071R	JMP NOEOF	NO
C125*				
C126***			***	
C127**	WAIT	STATEMENT SYNTAX	**	
C128***			***	
C129*				
C130	02514	017744R	WAITS JSB GETPF	
C131	02515	025317R	JMP EOST	END-OF-STATEMENT TEST
C132***				
C133**	CALL	STATEMENT SYNTAX		
C134***				
C135	02516	017377R	CALLS JSB GETCR	FETCH AND
C136	02517	025070R	JMP EOF	RECORD
C137	02520	035022R	ISZ SBPTR	LEFT
C138	02521	017671R	JSB LPCK	PARENTHESIS
C139	02522	015362R	JSB PRGIN	FETCH AND RECORD
C140	02523	000077	DEF D100	SUBROUTINE NUMBER
C141	02524	072047R	STA TEMP1	
C142	02525	017115R	JSB FND5B	FIND
C143	02526	160001	LDA 1,1	NUMBER
C144	02527	001727	ALF,ALF	OF
C145	02530	010071	AND .31	PARAMETERS
C146	02531	003000	CMA	RECORD
C147	02532	072054R	STA TEMPS+7	COMPLEMENT - 1
C148	02533	062047R	LDA TEMP1	RETRIEVE CHARACTER
C149	02534	007400	CALL2 CCB	
C150	02535	017066R	JSB SYMCK	COMMA?
C151	02536	002230R	DEF COMMA-1	
C152	02537	025545R	JMP CALL3	NO
C153	02540	035054R	CALL5 ISZ TEMPS+7	YES, MORE
C154	02541	002001	RSS	PARAMETERS PERMITTED?
C155	02542	015274R	SYE11 JSB ERROR	NO
C156	02543	017120R	JSB FSC	YES, FETCH
C157	02544	025534R	JMP CALL2	PARAMETER FORMULA
C158	02545	035054R	CALL3 ISZ TEMPS+7	ALL PARAMETERS PRESENT?
C159	02546	025542R	JMP SYE11	NO
C160	02547	065272R	LDB PARM	
C161	02550	005002	SZB	
C162	02551	017701R	JSB RPCK	YES, FETCH RIGHT PARENTHESIS
C163	02552	025317R	JMP EOST	END-OF-STATEMENT TEST
C164*				
C165***			***	
C166**	DATA	STATEMENT SYNTAX	**	
C167***			***	
C168*				
C169	02553	015402R	DATAS JSB CONST	FETCH A CONSTANT
C170	02554	025426R	JMP SYE12-1	NONE FOUND
C171	02555	017754R	JSB NUMOP	FIX UP PRECEDING OPERATOR
C172	02556	007400	CCB	CHECK
C173	02557	017066R	JSB SYMCK	FOR A
C174	02560	002230R	DEF COMMA-1	COMMA
C175	02561	025317R	JMP EOST	END-OF-STATEMENT TEST
C176	02562	025553R	JMP DATAS	FETCH ANOTHER NUMBER
C177*				
C178***			***	

CHECK SYNTAX AND TRANSLITERATE

C179** READ AND INPUT STATEMENT SYNTAX **

C180***

C181*

C182	02563	017562R	READS JSB VAROP	RECORD VARIABLE OPERAND
C183	02564	016274R	JSB ERROR	MISSING
C184	02565	000000	SYE13 NOP	
C185	02566	007400	CCB	CHECK
C186	02567	017066R	JSB SYMCK	FOR A
C187	02570	002230R	DEF COMMA-1	COMMA
C188	02571	002001	RSS	
C189	02572	025563R	JMP READS	IS, FETCH NEXT ITEM
C190	02573	005400	CLB	APPEND
C191	02574	175022R	STB SBPTR,I	END-OF-FORMULA
C192	02575	035022R	ISZ SBPTR	OPERATOR
C193	02576	025317R	JMP EOST	END OF STATEMENT TEST

C194*

C195***

C196** PRINT STATEMENT SYNTAX **

C197***

C198*

C199	02577	064051	PRIN1 LDB M2	NO,
C200	02600	017066R	JSB SYMCK	COMMA OR
C201	02601	002230R	DEF COMMA-1	SEMICOLON?
C202	02602	025610R	JMP PRIN2	NO
C203	02603	003400	PRINS CCA	YES, ENABLE
C204	02604	172046R	STA TEMP,I	FORMULA
C205	02605	035022R	ISZ SBPTR	
C206	02606	017377R	JSB GETCR	END-OF-STATEMENT?
C207	02607	025476R	JMP ACTST	YES
C208	02610	007400	PRIN2 CCB	
C209	02611	017066R	JSB SYMCK	QUOTE?
C210	02612	002226R	DEF QUOTE-1	
C211	02613	025627R	JMP PRIN3	NO
C212	02614	062143R	LDA .34	YES, SET QUOTE AS TERMINATOR
C213	02615	015437R	JSB CHRST	CHARACTER AND FETCH STRING
C214	02616	016274R	JSB ERROR	MISSING QUOTE
C215	02617	062227R	SYE14 LDA QUOTE	RECORD
C216	02620	172022R	STA SBPTR,I	QUOTE
C217	02621	035022R	ISZ SBPTR	
C218	02622	017377R	JSB GETCR	END-OF-STATEMENT?
C219	02623	025476R	JMP ACTST	YES
C220	02624	007400	CCB	ENABLE
C221	02625	175046R	STB TEMP,I	FORMULA
C222	02626	025577R	JMP PRIN1	NO
C223	02627	135046R	PRIN3 ISZ TEMP,I	TAB OR FORMULA PERMITTED?
C224	02630	016274R	JSB ERROR	NO
C225	02631	172022R	SYE15 STA SBPTR,I	SEARCH
C226	02632	062122R	LDA ATAB	FOR
C227	02633	007400	CCB	'TAB'
C228	02634	016242R	JSB TBSRH	
C229	02635	002401	CLA,RSS	NOT FOUND
C230	02636	062224R	LDA TABCN	
C231	02637	007400	CCB	BACKUP
C232	02640	045022R	ADB SBPTR	TO WORD WITH
C233	02641	075022R	STB SBPTR	PREVIOUS OPERATOR

CHECK SYNTAX AND TRANSLITERATE

0234	02642	002003	SZA,RSS	'TAB' ?
0235	02643	026653R	JMP PRIN4	NO
0236	02644	132022R	IOR SBPTR,I	
0237	02645	172022R	STA SBPTR,I	YES, RECORD IT
0238	02646	017744R	JSB GETPF	FETCH PARAMETER
0239	02647	006400	CLB	FOLLOW
0240	02650	176022R	STB SBPTR,I	WITH A
0241	02651	036022R	ISZ SBPTR	ZERO
0242	02652	026655R	JMP PRIN5	
0243	02653	017416R PRIN4	JSB BCKSP	BACKSPACE OVER LAST CHARACTER
0244	02654	017120R	JSB FSC	FETCH FORMULA
0245	02655	050065 PRIN5	CPA .10	END-OF-STATEMENT?
0246	02656	026476R	JMP ACTST	YES
0247	02657	026577R	JMP PRIN1	NO
0248*				
0249***			***	
0250**	MAT	STATEMENT SYNTAX	**	
0251***			***	
0252*				
0253	02660	017645R MATS	JSB LTR	FIRST
0254	02661	016274R	JSB ERROR	TWO CHARACTERS
0255	02662	017366R SYE16	JSB LETCK	LETTERS?
0256	02663	026726R	JMP MATS2	NO
0257	02664	036022R	ISZ SBPTR	YES, MOVE TO FRESH S-BUFFER WORD
0258	02665	066047R	LDB TEMP1	RETRIEVE FIRST LETTER AND
0259	02666	005727	BLF,BLF	PUT IT IN THE
0260	02667	030001	IOR 1	UPPER CHARACTER OF (A)
0261	02670	172022R	STA SBPTR,I	SEARCH
0262	02671	062107R	LDA MATIO	FUR
0263	02672	064051	LDB M2	'READ' OR
0264	02673	016242R	JSB TBSRH	'PRINT'
0265	02674	016274R	JSB ERROR	NOT FOUND
0266	02675	052213R SYE17	CPA RDUP	READ?
0267	02676	026714R	JMP MATS1	YES
0268	02677	017550R MATS0	JSB ARRID	RECORD ARRAY
0269	02700	050065	CPA .10	END-OF-STATEMENT?
0270	02701	026476R	JMP ACTST	YES
0271	02702	064051	LDB M2	NO,
0272	02703	017066R	JSB SYMCK	COMMA OR
0273	02704	002230R	DEF COMMA-1	SEMICOLON?
0274	02705	016274R	JSB ERROR	NO
0275	02706	017377R SYE18	JSB GETCR	END-OF-STATEMENT?
0276	02707	026712R	JMP *+3	YES
0277	02710	017416R	JSB BCKSP	NO
0278	02711	026677R	JMP MATS0	
0279	02712	036022R	ISZ SBPTR	INCLUDE
0280	02713	026476R	JMP ACTST	PARAMETER
0281	02714	017550R MATS1	JSB ARRID	RECORD ARRAY
0282	02715	017714R	JSB MATSB	IF SUBSCRIPT,
0283	02716	000000	NOP	RECORD IT
0284	02717	050065	CPA .10	END-OF-STATEMENT?
0285	02720	026476R	JMP ACTST	YES
0286	02721	007400	CCB	NO
0287	02722	017066R	JSB SYMCK	
0288	02723	002230R	DEF COMMA-1	COMMA?

CHECK SYNTAX AND TRANSLITERATE

C289	02724	026705R	JMP SYE18-1	NO
C290	02725	026714R	JMP MATS1	YES
C291	02726	072050R	MATS2 STA TEMP2	
C292	02727	062022R	LDA SBPTR	SAVE
C293	02730	073477R	STA ARYAD	OPERAND ADDRESS
C294	02731	062047R	LDA TEMP1	RETRIEVE FIRST LETTER
C295	02732	065152R	LDB .46	RECORD AN
C296	02733	017660R	JSB STROP	ARRAY
C297	02734	062050R	LDA TEMP2	RETRIEVE CHARACTER
C298	02735	007400	CCB	ASSIGNMENT
C299	02736	017066R	JSB SYMCK	
C300	02737	002242R	DEF ASSOP-1	OPERATOR?
C301	02740	026575R	JMP SYNE6	NO
C302	02741	163477R	LDA ARYAD,I	YES, RETRIEVE
C303	02742	012176R	AND MSK1	AND SAVE
C304	02743	172046R	STA TEMP,I	PREVIOUS ARRAY IDENTIFIER
C305	02744	017645R	JSB LTR	LETTER NEXT?
C306	02745	027014R	JMP MATS4	NO
C307	02746	017366R	JSB LETCK	YES, SECOND LETTER?
C308	02747	027030R	JMP MATS5	NO
C309	02750	036022R	ISZ SBPTR	YES,
C310	02751	066047R	LDB TEMP1	CONCATENATE
C311	02752	005727	BLF,BLF	LETTERS
C312	02753	030001	IOR 1	AND
C313	02754	172022R	STA SBPTR,I	SEARCH
C314	02755	062112R	LDA MATFN	FOR
C315	02756	064046	LDB M5	ARRAY
C316	02757	016242R	JSB TBSRH	FUNCTION
C317	02760	016274R	JSB ERROR	NOT FOUND
C318	02761	001727	SYE19 ALF,ALF	FOUND
C319	02762	001723	ALF,RAK	POSITION IT,
C320	02763	040062	ADA .7	COMPLETE OPERAND,
C321	02764	007400	CCB	COMBINE
C322	02765	046022R	ADB SBPTR	WITH
C323	02766	130001	IOR 1,I	OPERATOR,
C324	02767	032263R	IOR FLGBT	ADD FLAG BIT,
C325	02770	170001	STA 1,I	AND STORE
C326	02771	012176R	AND MSK1	'INV'
C327	02772	042253R	ADA D200	OR
C328	02773	002021	SSA,RSS	'TRN?
C329	02774	027002R	JMP MATS3	YES
C330	02775	017377R	JSB GETCR	NO, END-OF-STATEMENT?
C331	02776	026476R	JMP ACIST	YES
C332	02777	017714R	JSB MATSB	NO, SUBSCRIPT?
C333	03000	016274R	JSB ERROR	NO
C334	03001	026071R	SYE20 JMP NOEOF	
C335	03002	017377R	MATS3 JSB GETCR	
C336	03003	026070R	JMP EOF	
C337	03004	017671R	JSB LPCK	GET LEFT PARENTHESIS
C338	03005	017550R	JSB ARRID	FETCH AND RECORD AN ARRAY
C339	03006	017701R	JSB RPCK	RECORD A RIGHT PARENTHESIS
C340	03007	163477R	LDA ARYAD,I	RETRIEVE
C341	03010	012176R	AND MSK1	PREVIOUS ARRAY IDENTIFIER
C342	03011	152046R	CPA TEMP,I	MATCH LEFT-HAND SIDE ARRAY?
C343	03012	016274R	JSB ERROR	YES

CHECK SYNTAX AND TRANSLITERATE

```

0344 03013 026476R SYE21 JMP ACTST NO
0345 03014 036022R MATS4 ISZ SBPTR
0346 03015 017671R JSB LPCK FETCH LEFT PARENTHESIS
0347 03016 017120R JSB FSC FETCH FORMULA
0348 03017 017701R JSB RPCK FETCH RIGHT PARENTHESIS
0349 03020 007400 CCB MULTIPLICATION
0350 03021 017066R JSB SYMCK OPERATOR?
0351 03022 002250R DEF TIMES-1
0352 03023 016274R JSB ERROR NO
0353 03024 017550R SYE22 JSB ARRID YES, FETCH AND RECORD ARRAY
0354 03025 050065 CPA .10 END-OF-STATEMENT?
0355 03026 026476R JMP ACTST YES
0356 03027 026071R JMP NOEOF NO
0357 03030 072050R MATS5 STA TEMP2
0358 03031 062022R LDA SBPTR SAVE
0359 03032 073477R STA ARYAD OPERAND ADDRESS
0360 03033 062047R LDA TEMP1 RETRIEVE
0361 03034 066152R LDB .46 AND RECORD
0362 03035 017660R JSB STROP ARRAY
0363 03036 062050R LDA TEMP2 END-OF-
0364 03037 050065 CPA .10 STATEMENT?
0365 03040 026476R JMP ACTST YES
0366 03041 064050 LDB M3 NO, MUST BE
0367 03042 017066R JSB SYMCK A '+',
0368 03043 002244R DEF PLUS-1 '- ', OR '*'
0369 03044 016274R JSB ERROR ISN'T
0370 03045 006400 SYE23 CLB IS, SET FOR FALSE
0371 03046 040063 ADA .8
0372 03047 052251R CPA TIMES '*'?
0373 03050 027065R JMP MATS7 YES
0374 03051 076304R MATS6 STB PFLAG NO, SET PFLAG
0375 03052 017550R JSB ARRID GET SECOND ARRAY
0376 03053 050065 CPA .10 END-OF-STATEMENT?
0377 03054 002001 RSS YES
0378 03055 026071R JMP NOEOF NO
0379 03056 036304R ISZ PFLAG WAS OPERATOR A '*'?
0380 03057 026476R JMP ACTST NO
0381 03060 163477R LDA ARYAD,I YES, RETRIEVE
0382 03061 012176R AND MSK1 SECOND ARRAY
0383 03062 152046R CPA TEMP,I MATCH LEFT-HAND SIDE ARRAY?
0384 03063 016274R SYE24 JSB ERROR YES
0385 03064 026476R JMP ACTST NO
0386 03065 163477R MATS7 LDA ARYAD,I RETRIEVE
0387 03066 012176R AND MSK1 ARRAY
0388 03067 007400 CCB SET FOR TRUE
0389 03070 152046R CPA TEMP,I MATCH LEFT-HAND SIDE ARRAY?
0390 03071 027065R JMP SYE24 YES
0391 03072 027051R JMP MATS6 NO
0392**
0393*** JUMP TABLE FOR STATEMENT SYNTAX **
0394**
0395 03073 002310R SYNTB DEF LETS LET
0396 03074 002322R DEF DIMS DIM
0397 03075 002326R DEF COMS COM
0398 03076 002345R DEF DEFS DEF

```

CHECK SYNTAX AND TRANSLITERATE

C399	03077	002405R	DEF REMS	REM
C400	03100	002442R	DEF GOTOS	GO TO
C401	03101	002415R	DEF IFS	IF
C402	03102	002444R	DEF FORS	FOR
C403	03103	002476R	DEF NXTS	NEXT
C404	03104	002442R	DEF GOTOS	GOSUB
C405	03105	002510R	DEF ENDS+6	RETURN
C406	03106	002502R	DEF ENDS	END
C407	03107	002510R	DEF ENDS+6	STOP
C408	03110	002514R	DEF WAITS	WAIT
C409	03111	002516R	DEF CALLS	CALL
C410	03112	002553R	DEF DATAS	DATA
C411	03113	002563R	DEF READS	READ
C412	03114	002603R	DEF PRINS	PRINT
C413	03115	002563R	DEF READS	INPUT
C414	03116	002510R	DEF ENDS+6	RESTORE
C415	03117	002660R	DEF MATS	MAT
C416*				
C417***			***	
C418**	FORMULA SYNTAX CHECKER		**	
C419***			***	
C420*				
C421	03120	000000	FSC NOP	
C422	03121	002400	CLA	SET LEFT PARENTHESIS
C423	03122	172045R	STA TEMPS,I	COUNT TO ZERO
C424	03123	003400	FSC1 CCA	SET UNARY FLAG
C425	03124	072306R	STA UFLAG	TO TRUE
C426	03125	017562R	FSC2 JSB VAROP	LOOK FOR VARIABLE OPERAND
C427	03126	027265R	JMP FSC9	NOT FOUND
C428	03127	027233R	JMP FSC6	SUBSCRIPTED VARIABLE FOUND
C429	03130	017366R	JSB LETCK	FOLLOWED BY LETTER?
C430	03131	027233R	JMP FSC6	NO
C431	03132	064051	LDB M2	YES, LOOK FOR
C432	03133	017326R	JSB MCBCK	'AND' OR 'OR'
C433	03134	062047R	LDA TEMP1	NOT FOUND, FETCH PREVIOUS
C434	03135	001727	ALF,ALF	CHARACTER AND LEFT-JUSTIFY IT
C435	03136	032050R	IOR TEMP2	ADD LATEST CHARACTER
C436	03137	052257R	CPA FN	'FN'?
C437	03140	027166R	JMP FSC4	YES
C438	03141	172022R	STA SBPTR,I	NO,
C439	03142	062111R	LDA PDFNS	SEARCH FOR
C440	03143	065233R	LDB M11	PREDEFINED
C441	03144	015242R	JSB TBSRH	FUNCTION
C442	03145	027152R	JMP FSC3	NOT FOUND
C443	03146	001727	ALF,ALF	ASSEMBLE
C444	03147	001723	ALF,RAR	OPERAND
C445	03150	032263R	IOR FLGBT	ADD FLAG BIT
C446	03151	027174R	JMP FSC5	
C447	03152	035306R	FSC3 ISZ UFLAG	'NOT' PERMITTED?
C448	03153	027256R	JMP FSC8-1	NO
C449	03154	062121R	LDA ANOT	YES,
C450	03155	007400	CCB	SEARCH FOR
C451	03156	015242R	JSB TBSRH	'NOT'
C452	03157	027256R	JMP FSC8-1	'NOT' NOT FOUND
C453	03160	007400	CCB	RETRIEVE

CHECK SYNTAX AND TRANSLITERATE

0454	03161	045022R	ADB SBPTR	PREVIOUS WORD
0455	03162	160001	LDA 1,I	WORD
0456	03163	012220R	AND OPMSK	SET TO
0457	03164	170001	STA 1,I	NULL OPERAND
0458	03165	027323R	JMP FSC14	
0459	03166	017377R	FSC4 JSB GETCR	IDENTIFYING
0460	03167	025356R	JMP SYNE4	FUNCTION
0461	03170	017366R	JSB LETCK	LETTER?
0462	03171	025356R	JMP SYNE4	NO
0463	03172	040077	ADA D100	YES,
0464	03173	001723	ALF,RAR	ASSEMBLE AND SAVE
0465	03174	040062	FSC5 ADA .7 ::	
0466	03175	072047R	STA TEMP1	FUNCTION IDENTIFIER
0467	03176	007400	CCB	RETRIEVE
0468	03177	045022R	ADB SBPTR	PREVIOUS
0469	03200	160001	LDA 1,I	PROGRAM WORD
0470	03201	012220R	AND OPMSK	EXTRACT OPERATOR,
0471	03202	032047R	IOR TEMP1	APPEND OPERAND,
0472	03203	170001	STA 1,I	AND RECORD
0473	03204	017377R	JSB GETCR	LEFT PARENTHESIS
0474	03205	015274R	FSC61 JSB ERROR	OR
0475	03206	017671R	JSB LPOK	LEFT BRACKET?
0476	03207	017357R	JSB FRCUR	YES, SAVE LOCAL VARIABLES OF FSC
0477	03210	017120R	JSB FSC	FETCH ACTUAL PARAMETER
0478	03211	017334R	JSB FPOP	RESTORE LOCAL VARIABLES OF FSC
0479	03212	017701R	JSB RPOK	FETCH RIGHT PARENTHESIS
0480	03213	027276R	JMP FSC10+1	
0481	03214	064051	FSC7 LDB M2	CHECK FOR
0482	03215	017066R	JSB SYMCK	RIGHT PARENTHESIS
0483	03216	002234R	DEF RPARN-1	OR RIGHT BRACKET
0484	03217	027257R	JMP FSC8	NOT FOUND
0485	03220	062202R	LDA B2000 ::	
0486	03221	172022R	STA SBPTR,I	RIGHT PARENTHESIS
0487	03222	062146R	LDA .41	RESTORE RIGHT PARENTHESIS
0488	03223	007400	CCB	MATCHING
0489	03224	145045R	ADB TEMPS,I	LEFT
0490	03225	005020	SSB	PARENTHESIS?
0491	03226	027257R	JMP FSC8	NO
0492	03227	175045R	STB TEMPS,I	YES
0493	03230	036022R	ISZ SBPTR	
0494	03231	017377R	JSB GETCR	FETCH
0495	03232	060065	LDA .10	CHARACTER
0496	03233	050065	FSC6 CPA .10	END OF FORMULA?
0497	03234	027257R	JMP FSC8	YES
0498	03235	072306R	STA UFLAG	NO, SET UNARY FLAG TO FALSE
0499	03236	064046	LDB M5	SEARCH FOR A MULTICHARACTER
0500	03237	017326R	JSB MCHCK	BINARY OPERATOR
0501	03240	162022R	LDA SBPTR,I	NOT FOUND,
0502	03241	001727	ALF,ALF	RESTORE
0503	03242	010073	AND B177	CHARACTER
0504	03243	066300R	LDB MSFLG	SEARCH
0505	03244	017066R	JSB SYMCK	FOR A
0506	03245	002244R	DEF PLUS-1	BINARY OPERATOR
0507	03246	002001	RSS	NOT FOUND
0508	03247	027305R	JMP FSC12	FOUND

CHECK SYNTAX AND TRANSLITERATE

```

C509 03250 007400 CCB ASSIGNMENT
C510 03251 017066R JSB SYMCK
C511 03252 002242R DEF ASSOP-1 OPERATOR?
C512 03253 027214R JMP FSC7 NO
C513 03254 073534R STA SFLAG YES, SET
C514 03255 027123R JMP FSC1 'STORE OCCURRED' FLAG
C515 03256 062050R LDA TEMP2 RETRIEVE LETTER
C516 03257 166045R FSC8 LDB TEMPS,I ALL LEFT PARENTHESES
C517 03260 006002 SZB MATCHED?
C518 03261 016274R FSCE2 JSB ERROR NO
C519 03262 176022R STB SBPTR,I YES, RECORD AN
C520 03263 036022R ISZ SBPTR END-OF-FORMULA AND
C521 03264 127120R JMP FSC,I EXIT WITH CHARACTER IN (A)
C522*
C523 03265 052145R FSC9 CPA .40 LEFT
C524 03266 027301R JMP FSC11 PARENTHESIS
C525 03267 052170R CPA B133 OR LEFT BRACKET?
C526 03270 027301R JMP FSC11 YES
C527 03271 006400 CLB NO, SET SIGN
C528 03272 076041R STB SIGN POSITIVE
C529 03273 016430R JSB NUMCK NUMBER?
C530 03274 027310R JMP FSC13 NO
C531 03275 017754R FSC10 JSB NUMOP YES, FIX UP PRECEDING OPERATOR
C532 03276 064042 LDB M9 UPDATE
C533 03277 076300R STB MSFLG MULTIPLE STORE
C534 03300 027233R JMP FSC6 FLAG
C535 03301 036022R FSC11 ISZ SBPTR YES
C536 03302 052207R LDA B1140 ::
C537 03303 172022R STA SBPTR,I IT AND
C538 03304 136045R ISZ TEMPS,I COUNT IT
C539 03305 064042 FSC12 LDB M9 UPDATE
C540 03306 076300R STB MSFLG MULTIPLE STORE FLAG
C541 03307 027123R JMP FSC1 FLAG
C542 03310 036306R FSC13 ISZ UFLAG UNARY OPERATORS PERMITTED?
C543 03311 016274R FSCE3 JSB ERROR NO
C544 03312 066205R LDB UNMNC
C545 03313 052147R CPA .43 '+'?
C546 03314 027320R JMP *+4 YES
C547 03315 052150R CPA .45 NO, '-'?
C548 03316 027321R JMP *+3 YES
C549 03317 027311R JMP FSCE3 NO
C550 03320 046200R ADB B1400
C551 03321 036022R ISZ SBPTR UNARY
C552 03322 176022R STB SBPTR,I OPERATOR
C553 03323 064042 FSC14 LDB M9 UPDATE
C554 03324 076300R STB MSFLG MULTIPLE STORE FLAG
C555 03325 027125R JMP FSC2 FLAG
C556**
C557*** CHECK FOR A MULTICHARACTER BINARY OPERATOR **
C558**
C559 03326 000000 MCBCK NOP
C560 03327 172022R STA SBPTR,I SEARCH
C561 03330 062110R LDA MCBOP FOR 'AND'
C562 03331 016242R JSB TBSRH OR 'OR'
C563 03332 127320R JMP MCBCK,I NOT FOUND

```

CHECK SYNTAX AND TRANSLITERATE

```

C564 03333 027305R      JMP FSC12      FOUND
C565**
C566*** RESTORE FSC LOCAL QUANTITIES **
C567**
C568 03334 000000 FPOP  NOP
C569 03335 072047R      STA TEMP1      SAVE CHARACTER
C570 03336 066045R      LDB TEMPS
C571 03337 044046      ADB M5
C572 03340 076045R      STB TEMPS      RESTORE S-STACK TOP
C573 03341 006004      INB
C574 03342 160001      LDA 1,I
C575 03343 072300R      STA MSFLG      RESTORE MULTIPLE STORE FLAG
C576 03344 006004      INB
C577 03345 160001      LDA 1,I
C578 03346 072306R      STA UFLAG      RESTORE UNARY OPERATOR FLAG
C579 03347 006004      INB
C580 03350 160001      LDA 1,I
C581 03351 073120R      STA FSC      RESTORE FSC RETURN ADDRESS
C582 03352 006004      INB
C583 03353 160001      LDA 1,I      RESTORE
C584 03354 073562R      STA VAROP      VAROP RETURN ADDRESS
C585 03355 062047R      LDA TEMP1      RETRIEVE CHARACTER
C586 03356 127334R      JMP FPOP,I
C587**
C588*** SAVE LOCAL QUANTITIES OF FSC **
C589**
C590 03357 000000 FRCUR NOP
C591 03360 066045R      LDB TEMPS      FETCH CURRENT S-STACK POINTER
C592 03361 006004      INB      UPDATE IT
C593 03362 062300R      LDA MSFLG      DUMP MULTIPLE STORE
C594 03363 170001      STA 1,I      FLAG ON S-STACK
C595 03364 006004      INB
C596 03365 062306R      LDA UFLAG      STACK UNARY OPERATOR
C597 03366 170001      STA 1,I      FLAG
C598 03367 006004      INB
C599 03370 063120R      LDA FSC      STACK FSC
C600 03371 170001      STA 1,I      RETURN ADDRESS
C601 03372 063562R      LDA VAROP      STACK VAROP RETURN ADDRESS
C602 03373 017375R      JSB SSUV      AND CHECK FOR S-STACK OVERFLOW
C603 03374 127357R      JMP FRCUR,I
C604**
C605*** PUT ITEM ON S-STACK AND CHECK FOR OVERFLOW **
C606**
C607 03375 000000 SSUV  NOP      STORE QUANTITY
C608 03376 006004      INB      ADVANCE S-STACK POINTER
C609 03377 170001      STA 1,I      SAVE ITEM IN (A)
C610 03400 006004      INB      ADVANCE S-STACK POINTER
C611 03401 076045R      STB TEMPS      AND RECORD IT
C612 03402 007004      CMB,INB
C613 03403 045000R      ADB LWBM      LAST WORD
C614 03404 006020      SSB      EXCEEDED?
C615 03405 016274R FSC4 JSB ERROR      YES
C616 03406 127375R      JMP SSUV,I
C617**
C618*** CHECK FOR SUBSCRIPT PART **

```

CHECK SYNTAX AND TRANSLITERATE

C619**

C620	03407	000000	SBSCK	NO	CHARACTER IN (A)
C621	03410	064051	LDB	M2	LEFT BRACKET
C622	03411	017066R	JSB	SYMCK	OR
C623	03412	002270R	DEF	LBRAC-1	LEFT PARENTHESIS?
C624	03413	127407R	JMP	SBSCK,I	NO, RETURN VIA (P+1)
C625	03414	037407R	ISZ	SBSCK	YES, SET RETURN TO (P+2)
C626	03415	163477R	LDA	ARYAD,I	SET
C627	03416	012244R	AND	SBCON	
C628	03417	002004	INA		TO
C629	03420	173477R	STA	ARYAD,I	SINGLE SUBSCRIPT
C630	03421	062206R	LDA	B1100 ::	
C631	03422	172022R	SIA	SBPTR,I	LEFT BRACKET
C632	03423	005400	CLB		DIM OR COM
C633	03424	055302R	CPB	DFLAG	STATEMENT?
C634	03425	027477R	JMP	SBSC3	NO
C635	03426	015362R	JSB	PRGIN	FETCH INTEGER
C636	03427	000075	DEF	M256	SUBSCRIPT BOUND
C637	03430	005727	BLF	BLF	SAVE
C638	03431	075047R	STB	TEMP1	BOUND
C639	03432	007400	CCB		IS THE
C640	03433	017066R	JSB	SYMCK	NEXT CHARACTER
C641	03434	002240R	DEF	SCMMA-1	A COMMA?
C642	03435	027442R	JMP	SBSC1	NO
C643	03436	137477R	ISZ	ARYAD,I	YES, NOTE SECOND SUBSCRIPT
C644	03437	015362R	JSB	PRGIN	FETCH SECOND
C645	03440	000075	DEF	M256	INTEGER SUBSCRIPT BOUND
C646	03441	002001	RSS		
C647	03442	005404	SBSC1	CLB,INH	SET ONE-DIMENSIONAL CASE
C648	03443	035304R	ISZ	PFLAG	COM STATEMENT?
C649	03444	027454R	JMP	SBSC2	NO
C650	03445	072050R	STA	TEMP2	SAVE CHARACTER
C651	03446	060001	LDA	1	
C652	03447	032047R	IOR	TEMP1	RETRIEVE FIRST BOUND
C653	03450	017130R	JSB	MDIM	FIND STORAGE NEED
C654	03451	042054R	ADA	TEMPS+7	UPDATE COM
C655	03452	072054R	STA	TEMPS+7	STORAGE POINTER
C656	03453	062050R	LDA	TEMP2	RETRIEVE NEXT CHARACTER
C657	03454	064051	SBSC2	LDB M2	RIGHT PARENTHESIS
C658	03455	017066R	JSB	SYMCK	OR
C659	03456	002234R	DEF	RPARN-1	RIGHT BRACKET?
C660	03457	027261R	JMP	FSCE2	NO
C661	03460	062203R	LDA	B2400 ::	
C662	03461	172022R	SIA	SBPTR,I	RIGHT BRACKET
C663	03462	035022R	ISZ	SBPTR	ADJUST S-BUFFER POINTER
C664	03463	017377R	JSB	GETCR	FETCH FOLLOWING
C665	03464	060065	LDA	.10	CHARACTER
C666	03465	065302R	LDB	DFLAG	DIM OR COM
C667	03466	005002	SZB		STATEMENT?
C668	03467	127407R	JMP	SBSCK,I	YES
C669	03470	017334R	JSB	FPOP	RESTORE FSC LOCAL VARIABLES
C670	03471	064051	LDB	M2	RESTORE
C671	03472	045045R	ADB	TEMPS	S-STACK
C672	03473	075045R	SIB	TEMPS	POINTER
C673	03474	005004	INB		FETCH

CHECK SYNTAX AND TRANSLITERATE

C674	03475	164001	LDB 1,I	RETURN ADDRESS
C675	03476	124001	JMP 1,I	AND EXIT
C676	03477	063407R	SBSC3 LDA SBSC	SAVE
C677	03500	066045R	LDB TEMPS	RETURN ADDRESS
C678	03501	017375R	JSB SSUV	ON S-STACK
C679	03502	017357R	JSB FRCUR	SAVE FSC LOCAL VARIABLES
C680	03503	064042	LDB M9	SET MULTIPLE STORE FLAG
C681	03504	076300R	STB MSFLG	TO FALSE
C682	03505	063477R	LDA ARYAD	SAVE
C683	03506	066045R	LDB TEMPS	OPERAND
C684*				
C685	03507	017375R	JSB SSUV	ADDRESS
C686	03510	017120R	JSB FSC	GET SUBSCRIPT FORMULA
C687	03511	007400	CCB	CANCEL
C688	03512	045022R	ADB SBPTR	END-OF-FORMULA
C001	03513	076022R	STB SBPTR	OPERATOR
C002	03514	064051	LDB M2	RESTORE
C003	03515	045045R	ADB TEMPS	S-STACK
C004	03516	076045R	STB TEMPS	POINTER
C005	03517	006004	INB	RESTORE
C006	03520	164001	LDB 1,I	OPERAND
C007	03521	077477R	STB ARYAD	ADDRESS
C008	03522	007400	CCB	IS THE
C009	03523	017066R	JSB SYMCK	NEXT CHARACTER
C010	03524	002240R	DEF SCMA-1	A COMMA?
C011	03525	027454R	JMP SBSC2	NO
C012	03526	137477R	ISZ ARYAD,I	YES, NOTE SECOND SUBSCRIPT
C013	03527	017120R	JSB FSC	GET SUBSCRIPT FORMULA
C014	03530	007400	CCB	CANCEL
C015	03531	045022R	ADB SBPTR	END-OF-FORMULA
C016	03532	076022R	STB SBPTR	OPERATOR
C017	03533	027454R	JMP SBSC2	

CHECK SYNTAX AND TRANSLITERATE

```

C019**
C020*** CHECK SYNTAX OF ARRAY DEFINITIONS **
C021**
C022 03534 000000  ARSYS NOP
C023 03535 017550R      JSB ARRID      FETCH ARRAY IDENTIFIER
C024 03536 017407R      JSB SBSCCK     RECORD A SUBSCRIPT
C025 03537 016274R      JSB ERROR      MISSING SUBSCRIPT
C026 03540 050065  ARRE1 CPA .10      END-OF-STATEMENT?
C027 03541 127534R      JMP ARSYS,1    YES, RETURN VIA (P+1)
C028 03542 007400      CCB             NO,
C029 03543 017066R      JSB SYMCK      MUST BE
C030 03544 002230R      DEF COMMA-1    A COMMA
C031 03545 026071R      JMP NOEOF      ISN'T
C032 03546 037534R      ISZ ARSYS      IS, RETURN
C033 03547 127534R      JMP ARSYS,1    VIA (P+2)
C034**
C035*** FETCH ARRAY IDENTIFIER **
C036**
C037 03550 000000  ARRID NOP
C038 03551 017645R      JSB LTR        FETCH LETTER
C039 03552 016274R      JSB ERROR      NONE FOUND
C040 03553 062022R  ARRE2 LDA SBPTR    SAVE
C041 03554 073477R      STA ARYAD      OPERAND ADDRES
C042 03555 062047R      LDA TEMP1      RECORD
C043 03556 066152R      LDB .46        ARRAY
C044 03557 017660R      JSB STROP      IDENTIFIER
C045 03560 062050R      LDA TEMP2      RETRIEVE FOLLOWING CHARACTER
C046 03561 127550R      JMP ARRID,1
C047**
C048*** CHECK FOR VARIABLE OPERAND **
C049**
C050 03562 000000  VAROP NOP
C051 03563 017645R      JSB LTR        LETTER?
C052 03564 127562R      JMP VAROP,1    NO, EXIT VIA (P+1)
C053 03565 037562R      ISZ VAROP
C054 03566 052145R      CPA .40        LEFT PARENTHESIS?
C055 03567 027634R      JMP VAR05      YES
C056 03570 052170R      CPA B153      NO, LEFT BRACKET?
C057 03571 027634R      JMP VAR05      YES
C058 03572 037562R      ISZ VAROP      NO
C059 03573 017353R      JSB DIGCK      DIGIT?
C060 03574 027610R      JMP VAR01      NO
C061 03575 062047R      LDA TEMP1      YES, RETRIEVE LETTER,
C062 03576 046154R      ADB .46        AND RESTORE ASCII DIGIT
C063 03577 075047R      STB TEMP1
C064 03600 046242R      ADB D62 ::
C065 03601 006021      SSB,RSS ::
C066 03602 027311R      JMP FSCE3 ::
C067 03603 066047R      LDB TEMP1 ::
C068 03604 017660R      JSB STROP      RECORD VARIABLE
C069 03605 017377R      JSB GETCR      FETCH FOLLOWING
C070 03606 060065      LDA .10        CHARACTER
C071 03607 027614R      JMP VAR02
C072 03610 062047R  VAR01 LDA TEMP1    RETRIEVE LETTER,
C073 03611 066153R      LDB .47        SET 'NO DIGIT',

```

CHECK SYNTAX AND TRANSLITERATE

C074	03612	017660R	JSB STROP	AND RECORD VARIABLE
C075	03613	062050R	LDA TEMP2	RETRIEVE FOLLOWING CHARACTER
C076	03614	072050R	VAR02 STA TEMP2	SAVE CHARACTER
C077	03615	006400	CLB	INSIDE A
C078	03616	056304R	CPB PFLAG	DEF STATEMENT?
C079	03617	127562R	JMP VAROP,I	NO, EXIT VIA (P+3)
C080	03620	007400	CCB	
C081	03621	046022R	ADB SBPTR	RETRIEVE
C082	03622	160001	LDA 1,I	
C083	03623	012176R	AND MSK1	OPERAND
C084	03624	052304R	CFA PFLAG	MATCH PARAMETER?
C085	03625	027630R	JMP VAR04	YES
C086	03626	062050R	VAR05 LDA TEMP2	NO, RETRIEVE
C087	03627	127562R	JMP VAROP,I	CHARACTER AND EXIT VIA (P+3)
C088	03630	160001	VAR04 LDA 1,I	SET OPERAND TO
C089	03631	032263R	IUR FLGBT	ACTUAL PARAMETER
C090	03632	170001	STA 1,I	AND RECORD IT
C091	03633	027626R	JMP VAR03	
C092	03634	062022R	VAR05 LDA SBPTR	SAVE
C093	03635	073477R	STA ARYAD	OPERAND ADDRESS
C094	03636	062047R	LDA TEMP1	RETRIEVE LETTER
C095	03637	066152R	LDB .46	RECORD
C096	03640	017660R	JSB STROP	ARRAY IDENTIFIER
C097	03641	062170R	LDA B133	RETRIEVE LEFT BRACKET
C098	03642	017407R	JSB SBSCK	FETCH SUBSCRIPT
C099	03643	000000	NOP	
C100	03644	127562R	JMP VAROP,I	EXIT VIA (P+2)
C101**				
C102***	FETCH A LETTER **			
C103**				
C104	03645	000000	LTR NOP	
C105	03646	017377R	JSB GETCR	
C106	03647	060065	LDA .10	
C107	03650	017366R	JSB LETCK	LETTER?
C108	03651	127645R	JMP LTR,I	NO, EXIT VIA (P+1)
C109	03652	037645R	ISZ LTR	YES,
C110	03653	072047R	STA TEMP1	SAVE IT
C111	03654	017377R	JSB GETCR	NEXT CHARACTER
C112	03655	060065	LDA .10	TO (A)
C113	03656	072050R	STA TEMP2	SAVE SECOND CHARACTER
C114	03657	127645R	JMP LTR,I	EXIT VIA (P+2)
C115**				
C116***	STORE AN OPERAND NAME **			
C117**				
C118	03660	000000	STROP NOP	LETTER IN (A), NUMBER IN (B)
C119	03661	049077	ADA D100	NUMERICALLY ADJUST THE
C120	03662	046241R	ADB D53	OPERAND NAME
C121	03663	001723	ALF,RAR ::	
C122	03664	030001	IUR 1	TWO PARTS
C123	03665	132022R	IUR SBPTR,I	COMPLETE OPERAND-OPERATOR PAIR
C124	03666	172022R	STA SBPTR,I	AND STORE IT
C125	03667	036022R	ISZ SBPTR	UPDATE S-BUFFER POINTER
C126	03670	127660R	JMP STROP,I	

CHECK SYNTAX AND TRANSLITERATE

C128**

C129*** CHECK FOR LEFT PARENTHESIS **

C130**

C131	03671	000000	LPCK	NOP	CHARACTER IN (A)
C132	03672	064051		LDB M2	LEFT PARENTHESIS
C133	03673	017066R		JSB SYMCK	OR
C134	03674	002270R		DEF LBRAC-1	LEFT BRACKET?
C135	03675	027205R		JMP FSCE1	NO
C136	03676	062207R		LDA B1140	::
C137	03677	172022R		STA SBPTR,I	LEFT PARENTHESIS
C138	03700	127671R		JMP LPCK,I	EXIT

C139**

C140*** CHECK FOR RIGHT PARENTHESIS **

C141**

C142	03701	000000	RPCK	NOP	
C143	03702	064051		LDB M2	RIGHT PARENTHESIS
C144	03703	017066R		JSB SYMCK	OR
C145	03704	002234R		DEF RPARN-1	RIGHT BRACKET?
C146	03705	027261R		JMP FSCE2	NO
C147	03706	062202R		LDA B2000	YES, RECORD A
C148	03707	172022R		STA SBPTR,I	RIGHT PARENTHESIS
C149	03710	035022R		ISZ SBPTR	UPDATE SYNTAX BUFFER POINTER
C150	03711	017377R		JSB GETCH	FETCH
C151	03712	060065		LDA .10	FOLLOWING CHARACTER
C152	03713	127701R		JMP RPCK,I	

C153**

C154*** FETCH MAT STATEMENT SUBSCRIPT **

C155**

C156	03714	000000	MATSB	NOP	
C157	03715	064051		LDB M2	LEFT PARENTHESIS
C158	03716	017066R		JSB SYMCK	OR
C159	03717	002270R		DEF LBRAC-1	LEFT BRACKET?
C160	03720	127714R		JMP MATSB,I	NO
C161	03721	037714R		ISZ MATSB	YES, SET RETURN ADDRESS
C162	03722	062206R		LDA B1100	RECORD A
C163	03723	172022R		STA SBPTR,I	LEFT BRACKET
C164	03724	017120R		JSB FSC	FETCH SUBSCRIPT
C165	03725	007400		CCB	
C166	03726	017066R		JSB SYMCK	COMMA?
C167	03727	002230R		DEF COMMA-1	
C168	03730	002001		RSS	NO
C169	03731	017120R		JSB FSC	YES, FETCH SUBSCRIPT
C170	03732	064051		LDB M2	RIGHT PARENTHESIS
C171	03733	017066R		JSB SYMCK	OR
C172	03734	002234R		DEF RPARN-1	RIGHT BRACKET
C173	03735	027261R		JMP FSCE2	
C174	03736	062203R		LDA B2400	::
C175	03737	172022R		STA SBPTR,I	RIGHT BRACKET
C176	03740	035022R		ISZ SBPTR	
C177	03741	017377R		JSB GETCH	END-OF-STATEMENT?
C178	03742	025476R		JMP ACIST	YES
C179	03743	127714R		JMP MATSB,I	

CHECK SYNTAX AND TRANSLITERATE

C181**

C182*** FETCH PARENTHESIZED FORMULA **

C183**

C184	03744	000000	GETPF	NOP	
C185	03745	017377R	JSB	GETCR	
C186	03746	026070R	JMP	EOF	
C187	03747	036022R	ISZ	SBPTR	
C188	03750	017671R	JSB	LPCK	FETCH LEFT PARENTHESIS
C189	03751	017120R	JSB	FSC	FETCH FORMULA
C190	03752	017701R	JSB	RPCK	GET RIGHT PARENTHESIS
C191	03753	127744R	JMP	GETPF,I	

C192**

C193*** FLAG OPERATOR WHICH PRECEDES NUMBER **

C194**

C195	03754	000000	NUMOP	NOP	
C196	03755	072052R	STA	TEMP4	
C197	03756	064050	LDB	M3	FETCH
C198	03757	046022R	ADB	SBPTR	PRECEDING
C199	03760	160001	LDA	1,I	OPERATOR
C200	03761	032263R	IOR	FLGBT	ADD FLAG BIT
C201	03762	170001	STA	1,I	REPLACE OPERATOR
C202	03763	062052R	LDA	TEMP4	
C203	03764	127754R	JMP	NUMOP,I	

CHECK SYNTAX AND TRANSLITERATE

```

0205*
0206*   SYSTEM COMMAND TABLE
0207*
0208  03765 000003 SYCMD OCT 00003
0209  03766 051125      ASC 2,RUN      EXECUTE PROGRAM
        03767 047040
0210*
0211  03770 001003      OCT 1003  ::
0212  03771 051503      ASC 2,SCR      SCRATCH PROGRAM
        03772 051040
0213*
0214  03773 001404      OCT 1404  ::
0215  03774 045111      ASC 2,LIST     LIST COMMAND
        03775 051524
0216*
0217  03776 002405      OCT 2405  ::
0218  03777 050114      ASC 3,PLIST    PUNCH LIST COMMAND
        04000 044523
        04001 052040
0219*
0220  04002 005003      OCT 5003  ::
0221  04003 050124      ASC 2,PTA
        04004 040440
0222*
0223  04005 033004      OCT 33004
0224  04006 051524 SYCMD ASC 2,STOP    ABORT CURRENT ACTIVITY
        04007 047520
0225*
0226  04010 023003      OCT 23003 ::
0227  04011 052101      ASC 2,TAP      ACTIVATE TTY TAPE MODE
        04012 050040
0228*
0229  04013 024003      OCT 24003 ::
0230  04014 041131      ASC 2,BYE      EXIT SYSTEM
        04015 042440
0231  04016 025005      OCT 25005 ::
0232  04017 050125      ASC 3,PUNCH
        04020 047103
        04021 044040
0233  04022 027404      OCT 27404
0234  04023 045117      ASC 2,LOAD ::
        04024 040504
0235*****
0236**
0237*** PRINT NAME TABLE FOR OPERATORS **
0238**
0239  04025 015003 LET   OCT 15003 ::
0240  04026 045105      ASC 2,LET
        04027 052040
0241  04030 015403 DIM   OCT 15403 ::
0242  04031 042111      ASC 2,DIM
        04032 045440
0243  04033 015003 COM   OCT 16003 ::
0244  04034 041517      ASC 2,COM
        04035 045440

```

CHECK SYNTAX AND TRANSLITERATE

```
C245 04036 015403 DEF OCT 16403 ::
C246 04037 042105 ASC 2,DEF
      04040 043040
C247 04041 017003 REM OCT 17003 ::
C248 04042 051105 ASC 2,REM
      04043 046440
C249 04044 017404 GOTO OCT 17404 ::
C250 04045 043517 ASC 2,GOTO
      04046 052117
C251 04047 020002 IF OCT 20002 ::
C252 04050 044506 ASC 1,IF
C253 04051 020403 FOR OCT 20403 ::
C254 04052 043117 ASC 2,FOR
      04053 051040
C255 04054 021004 NEXT OCT 21004 ::
C256 04055 047105 ASC 2,NEXT
      04056 054124
C257 04057 021405 GOSUB OCT 21405 ::
C258 04060 043517 ASC 3,GOSUB
      04061 051525
      04062 041040
C259 04063 022006 RTN OCT 22006 ::
C260 04064 051105 ASC 3,RETURN
      04065 052125
      04066 051116
C261 04067 022403 END OCT 22403 ::
C262 04070 042516 ASC 2,END
      04071 042040
C263 04072 023004 STP OCT 23004 ::
C264 04073 051524 ASC 2,STOP
      04074 047520
C265 04075 023404 WAIT OCT 23404 ::
C266 04076 053501 ASC 2,WAIT
      04077 044524
C267 04100 024004 CALL OCT 24004 ::
C268 04101 041501 ASC 2,CALL
      04102 046114
C269 04103 024404 DATA OCT 24404 ::
C270 04104 042101 ASC 2,DATA
      04105 052101
C271 04106 025004 READ OCT 25004 ::
C272 04107 051105 ASC 2,READ
      04110 040504
C273 04111 025405 PRINT OCT 25405 ::
C274 04112 050122 ASC 3,PRINT
      04113 044516
      04114 052040
C275 04115 025005 INPUT OCT 26005 ::
C276 04116 044516 ASC 3,INPUT
      04117 050125
      04120 052040
C277 04121 026407 RSTOR OCT 26407 ::
C278 04122 051105 ASC 4,RESTORE
      04123 051524
      04124 047522
```

CHECK SYNTAX AND TRANSLITERATE

	04125	042440		
C279	04126	027003	MAT	OCT 27003 ::
C280	04127	046501		ASC 2,MAT
	04130	052040		
C281	04131	027404	THEN	OCT 27404
C282	04132	052110		ASC 2,THEN
	04133	042516		
C283	04134	030002	TO	OCT 30002 ::
C284	04135	052117		ASC 1,TO
C285	04136	030404	STEP	OCT 30404 ::
C286	04137	051524		ASC 2,STEP
	04140	042520		
C287	04141	013403	NOT	OCT 13403
C288	04142	047117		ASC 2,NOT
	04143	052040		
C289	04144	013003	AND	OCT 13003
C290	04145	040516		ASC 2,AND
	04146	042040		
C291	04147	012402	OR	OCT 12402
C292	04150	047522		ASC 1,OR
C293	04151	014002	GTE	OCT 14002 ::
C294	04152	037075		ASC 1,>=
C295	04153	014402	LTE	OCT 14402 ::
C296	04154	035075		ASC 1,<=
C297	04155	074002	AUNEQ	OCT 74002 ::
C298	04156	035076		ASC 1,<>
C299*				
C300	04157	000403	TAB	OCT 403 ::
C301	04160	052101		ASC 2,TAB
	04161	041040		
C302	04162	001003	SIN	OCT 1003 ::
C303	04163	051511		ASC 2,SIN
	04164	047040		
C304	04165	001403	COS	OCT 1403
C305	04166	041517		ASC 2,COS
	04167	051440		
C306	04170	002003	TAN	OCT 2003 ::
C307	04171	052101		ASC 2,TAN
	04172	047040		
C308	04173	002403	ATN	OCT 2403 ::
C309	04174	040524		ASC 2,ATN
	04175	047040		
C310	04176	003003	EXPN	OCT 3003 ::
C311	04177	042530		ASC 2,EXP
	04200	050040		
C312	04201	003403	LOG	OCT 3403 ::
C313	04202	046117		ASC 2,LOG
	04203	043440		
C314	04204	004003	ABS	OCT 4003 ::
C315	04205	040502		ASC 2,ABS
	04206	051440		
C316	04207	004403	SQR	OCT 4403 ::
C317	04210	051521		ASC 2,SQR
	04211	051040		
C318	04212	005003	INT	OCT 5003 ::

CHECK SYNTAX AND TRANSLITERATE

C319	04213	044516	ASC 2,INT	
	04214	052040		
C320	04215	005403	RND OCT 5403 ::	
C321	04216	051116	ASC 2,RND	
	04217	042040		
C322	04220	006003	SGN OCT 6003 ::	
C323	04221	051507	ASC 2,SGN	
	04222	047040		
C324	04223	006403	ZER OCT 6403 ::	
C325	04224	055105	ASC 2,ZER	
	04225	051040		
C326	04226	007003	CON OCT 7003 ::	
C327	04227	041517	ASC 2,CON	
	04230	047040		
C328	04231	007403	IDN OCT 7403 ::	
C329	04232	044504	ASC 2,IDN	
	04233	047040		
C330	04234	010003	INV OCT 10003 ::	
C331	04235	044516	ASC 2,INV	
	04236	053040		
C332	04237	010403	TRN OCT 10403 ::	
C333	04240	052122	ASC 2,TRN	
	04241	047040		
C334**				
C335*** TABLE SEARCH FOR MULTICHARACTER SYMBOLS **				
C336**				
C337	04242	000000	TBSRH NOP	
C338	04243	072362R	STA TABLE	STORE TABLE ADDRESS
C339	04244	075055R	STB LENGH	STORE -(NUMBER OF ENTRIES)
C340	04245	062017R	LDA BAADDR	SAVE
C341	04246	072051R	STA TEMP3	INPUT
C342	04247	062020R	LDA CCNT	BUFFER
C343	04250	072052R	STA TEMP4	STATUS
C344	04251	062022R	LDA SBPTR	INITIALIZE END-OF-SYMBOL
C345	04252	072400R	STA SMEND	POINTER
C346	04253	002404	CLA,INA	COUNT FIRST CHARACTER OF
C347	04254	072626R	STA SLENG	SYMBOL
C348	04255	162022R	LDA SBPTR,I	FETCH PARTIAL SYMBOL
C349	04256	010073	AND B177	TWO
C350	04257	152022R	CPA SBPTR,1	CHARACTERS?
C351	04260	002001	RSS	NO
C352	04261	025304R	JMP TSR10	YES
C353	04262	001727	ALF,ALF	LEFT-JUSTIFY
C354	04263	032141R	IOR .32	FIRST CHARACTER AND
C355	04264	172022R	STA SBPTR,I	APPEND BLANK
C356	04265	017377R	TSRC1 JSB GETCR	FETCH NEXT CHARACTER
C357	04266	025355R	JMP TSRC9	END-OF-STATEMENT
C358	04267	065626R	LDB SLENG	CHECK FOR
C359	04270	054065	CPB .10	IMPOSSIBLE LENGTH
C360	04271	025355R	JMP TSRC9	
C361	04272	004010	SLB	EVEN-NUMBERED CHARACTER?
C362	04273	025301R	JMP TSRC2	YES
C363	04274	035400R	ISZ SMEND	NO, FETCH FRESH WORD,
C364	04275	001727	ALF,ALF	LEFT-JUSTIFY CHARACTER,
C365	04276	032141R	IOR .32	APPEND BLANK,

CHECK SYNTAX AND TRANSLITERATE

0366	04277	172400R	STA SMEND,I	AND STORE
0367	04300	026304R	JMP TSR10	
0368	04301	042240R	TSRC2 ADA M32	DELETE BLANK,
0369	04302	142400R	ADA SMEND,I	FILL SECOND CHARACTER,
0370	04303	172400R	STA SMEND,I	AND STORE
0371	04304	036626R	TSR10 ISZ SLENG	COUNT IT
0372	04305	066055R	LDB LNGTH	INITIALIZE TABLE LENGTH
0373	04306	076053R	STB COUNT	COUNTER
0374	04307	062362R	LDA TABLE	
0375	04310	072553R	TSRC3 STA TBLPT	SET TABLE POINTER
0376	04311	162553R	LDA TBLPT,I	EXTRACT SYMBOL LENGTH
0377	04312	010070	AND .15	FROM TABLE AND COMPARE
0378	04313	052626R	CPA SLENG	WITH CURRENT SYMBOL
0379	04314	026323R	JMP TSRC5	EQUAL?
0380	04315	040056	TSRC4 ADA .3	DIFFERENT,
0381	04316	001100	AKS	UPDATE
0382	04317	042553R	ADA TBLPT	TABLE POINTER
0383	04320	036053R	ISZ COUNT	MORE ENTRIES?
0384	04321	026310R	JMP TSRC3	YES
0385	04322	026265R	JMP TSRC1	NO
0386	04323	066553R	TSRC5 LDB TBLPT	SET POINTER TO
0387	04324	076603R	STB TSPTR	TABLE SYMBOL
0388	04325	066022R	LDB SBPTR	SET (B) TO INPUT
0389	04326	026332R	JMP TSRC7	SYMBOL POINTER
0390	04327	056400R	TSRC6 CPB SMEND	ALL OF SYMBOL CONSIDERED?
0391	04330	026340R	JMP TSRC8	YES, MATCH OCCURRED
0392	04331	006004	INB	NO, INCREMENT
0393	04332	036603R	TSRC7 ISZ TSPTR	SYMBOL POINTERS
0394	04333	162603R	LDA TSPTR,I	FETCH WORD FROM TABLE
0395	04334	150001	CPA 1,1	MATCH WITH INPUT SYMBOL?
0396	04335	026327R	JMP TSRC6	YES
0397	04336	062626R	LDA SLENG	NO, WRONG
0398	04337	026315R	JMP TSRC4	SYMBOL
0399	04340	162553R	TSRC8 LDA TBLPT,I	EXTRACT
0400	04341	012220R	AND OPMSK	SYMBOL CODE
0401	04342	172022R	STA SBPTR,I	
0402	04343	162553R	LDA TBLPT,I	
0403	04344	001727	ALF,ALF	
0404	04345	001222	RAL,RAL ::	
0405	04346	001200	RAL ::	
0406	04347	010062	AND .7	
0407	04350	003000	CMA	
0408	04351	072272R	STA PARM	
0409	04352	162022R	LDA SBPTR,I	
0410	04353	036242R	ISZ TBSRH	AND RETURN VIA
0411	04354	126242R	JMP TBSRH,I	'SUCCESS' EXIT
0412	04355	062051R	TSRC9 LDA TEMP3	RESTORE
0413	04356	072017R	STA BADDR	INPUT
0414	04357	062052R	LDA TEMP4	BUFFER
0415	04360	072020R	STA CCNT	STATUS
0416	04361	126242R	JMP TBSRH,I	'FAILURE' EXIT
0417**				
0418***	FE	CH	AND RECORD PROGRAM INTEGER	**
0419**				
0420	04362	000000	PRGIN NOP	

CHECK SYNTAX AND TRANSLITERATE

C421	04363	162022R	LDA SBPTR,I	SET	
C422	04364	032263R	IOR FLGBT	'INTEGER	
C423	04365	040056	ADA .3	FOLLOWS'	
C424	04366	172022R	STA SBPTR,I	OPERAND	
C425	04367	162362R	LDA PRGIN,I	GIVE ADDRESS	
C426	04370	072375R	STA PRG11	TO INTCK	
C427	04371	036022R	ISZ SBPTR		
C428	04372	017377R	JSB GETCR		
C429	04373	016274R	SYE25 JSB ERROR		
C430	04374	016400R	JSB INTCK	FETCH	
C431	04375	000000	PRG11 NOP		
C432	04376	036362R	ISZ PRGIN		
C433	04377	126362R	JMP PRGIN,I		
C434**					
C435*** BUILD AN INTEGER **					
C436**					
C437	04400	000000	INTCK NOP	CHARACTER IN (A)	
C438	04401	006400	CLB	STORE	
C439	04402	076626R	STB INTGR	PARTIAL RESULT	
C440	04403	017353R	INTC1 JSB DIGCK	DIGIT?	
C441	04404	026422R	JMP INTC2	NO	
C442	04405	103101	CLC		
C443	04406	065626R	LDB INTGR	MULTIPLY	
C444	04407	044001	ADB 1	PARTIAL	
C445	04410	044001	ADB 1	RESULT	
C446	04411	046626R	ADB INTGR	BY	
C447	04412	044001	ADB 1	10	
C448	04413	044000	ADB 0	ADD LATEST DIGIT	
C449	04414	102201	SOC	OVERFLOW?	
C450	04415	026373R	JMP SYE25	YES	
C451	04416	075626R	STB INTGR	STORE PARTIAL RESULT	
C452	04417	017377R	JSB GETCR	NO, FETCH	
C453	04420	060065	LDA .10	NEXT CHARACTER	
C454	04421	026403R	JMP INTC1		
C455	04422	066626R	INTC2 LDB INTGR	ZERO	
C456	04423	005003	SZB,RSS	INTEGER?	
C457	04424	026373R	JMP SYE25	YES	
C458	04425	176022R	STB SBPTR,I	NO, RECORD IT	
C459	04426	165400R	LDB INTCK,I	INTEGER	
C460	04427	164001	LDB 1,I	TOO	
C461	04430	046626R	ADB INTGR	LARGE?	
C462	04431	005021	SSB,RSS		
C463	04432	026373R	JMP SYE25	YES	
C464	04433	066626R	LDB INTGR	NO,	
C465	04434	036022R	ISZ SBPTR	RETURN WITH	
C466	04435	036400R	ISZ INTCK	INTEGER	
C467	04436	126400R	JMP INTCK,I	IN (B)	
C468**					
C469*** PROCESS CHARACTER STRING **					
C470**					
C471	04437	000000	CHRST NOP		
C472	04440	072050R	STA TEMP2	RECORD TERMINATOR CHARACTER	
C473	04441	060065	LDA .10	DUMMY	
C474	04442	072271R	STA BLANK	DELETE CHARACTER	
C475	04443	017377R	CHRS1 JSB GETCR		

CHECK SYNTAX AND TRANSLITERATE

0476	04444	025462R	JMP CHRS3	TO END-OF-STATEMENT EXIT
0477	04445	052050R	CPA TEMP2	TERMINATOR CHARACTER?
0478	04446	025461R	JMP CHRS2	YES
0479	04447	132022R	IOR SBPTR,I	NO, FILL
0480	04450	172022R	STA SBPTR,I	SECOND CHARACTER
0481	04451	017377R	JSB GETCR	
0482	04452	025462R	JMP CHRS3	TO END-OF-STATEMENT EXIT
0483	04453	052050R	CPA TEMP2	TERMINATOR CHARACTER?
0484	04454	025461R	JMP CHRS2	YES
0485	04455	035022R	ISZ SBPTR	NO, MOVE TO NEW WORD
0486	04456	001727	ALF,ALF	AND STORE
0487	04457	172022R	STA SBPTR,I	FIRST CHARACTER
0488	04460	025443R	JMP CHRS1	
0489	04461	036437R	CHRS2 ISZ CHRST	SET (P+2) EXIT
0490	04462	035022R	CHRS3 ISZ SBPTR	MUVE TO NEXT BUFFER WORD
0491	04463	062141R	LDA .32	RESTORE BLANK AS
0492	04464	072271R	STA BLANK	DELETE CHARACTER
0493	04465	125437R	JMP CHRST,I	

CHECK SYNTAX AND TRANSLITERATE

```

C495**
C496***  DELETE STATEMENT  **
C497**
C498 04466 162021R DLSTM LDA SBUFA,I  LOAD SEQUENCE NUMBER
C499 04467 015553R JSB FNDPS  FIND STATEMENT TO BE DELETED
C500 04470 027611R JMP PEXMK  DOESN'T
C501 04471 027611R JMP PEXMK  EXIST
C502 04472 002400 CLA  ZERO WORD SKIP FOR .DESTINATION
C503 04473 005004 INB  ADDRESS OF SOURCE WORD SKIP IN B
C504 04474 015603R JSB CLPRG  CLOSE UP PROGRAM
C505 04475 027611R JMP PEXMK  EXIT TO PHASE 1 WAIT
C506*
C507***  ***
C508**  ACCEPT STATEMENT  **
C509***  ***
C510*
C511 04476 062021R ACTSI LDA SBUFA  COMPUTE
C512 04477 003004 CMA,INA  LENGTH
C513 04500 042022R ADA SBPTR  OF STATEMENT
C514 04501 172046R STA TEMP,I  AND RECORD IT
C515 04502 065010C LDB PBPTR  PICK UP STATEMENT ADDR.
C516 04503 075051R STB TEMP3  AND SAVE
C517 04504 162021R LDA SBUFA,I  LOAD SEQUENCE NUMBER
C518 04505 003004 CMA,INA  MAKE SEQ # NEGATIVE
C519 04506 042015R ADA ULDLN  ADD PREVIOUS SEQ #
C520 04507 002020 SSA  P-SEQ < S-SEQ ?
C521 04510 025531R JMP ACCS1  YES, APPEND STATEMENT TO PROG.
C522 04511 162021R LDA SBUFA,I  NO, CONTINUE CHECKING
C523 04512 015553R JSB FNDPS  SEARCH ON SEQUENCE NUMBER
C524 04513 025531R JMP ACCS1  APPEND STATEMENT TO PROGRAM
C525 04514 025547R JMP ACCS4  INSERT STATEMENT IN PROGRAM
C526 04515 005004 INB  REPLACE STATEMENT IN PROGRAM
C527 04516 060001 LDA 1
C528 04517 017204R JSB DREAD  COMPARE LENGTHS OF
C529 04520 003004 CMA,INA  STATEMENT BEING REPLACED
C530 04521 142046R ADA TEMP,I  AND STATEMENT
C531 04522 002003 SZA,RSS  REPLACING IT
C532 04523 025533R JMP ACCS2  EQUAL
C533 04524 002021 SSA,RSS
C534 04525 025550R JMP ACCS4+1  SHORTER
C535 04526 162046R LDA TEMP,I  LONGER,
C536 04527 015603R JSB CLPRG  CLOSE UP PROGRAM
C537 04530 025533R JMP ACCS2
C538 04531 162046R ACCS1 LDA TEMP,I  LOAD PROGRAM SPACE REQUIREMENT
C539 04532 015626R JSB UVCHK  SUFFICIENT PROGRAM SPACE LEFT?
C540 04533 005400 ACCS2 CLB  YES, SET COUNTER TO ZERO
C541 04534 062021R LDA SBUFA  INITIALIZE
C542 04535 072050R STA TEMP2  SOURCE ADDRESS
C543 04536 162050R ACCS3 LDA TEMP2,I  TRANSFER WORD FROM
C544 04537 017151R JSB DSTOR  S-BUFFER TO PROGRAM SPACE
C545 04540 000051R DEF TEMP3
C546 04541 035050R ISZ TEMP2  INCREMENT SOURCE AND
C547 04542 035051R ISZ TEMP3  .DESTINATION ADDRESSES
C548 04543 005004 INB  BUMP COUNTER
C549 04544 155046R CPB TEMP,I  ENTIRE STATEMENT MOVED?

```

CHECK SYNTAX AND TRANSLITERATE

C550	04545	027611R	JMP PEXMK	YES
C551	04546	026536R	JMP ACCS3	NO
✓C552	04547	162046R	ACCS4 LDA TEMP,I	LOAD PROGRAM SPACE REQUIREMENT
C553	04550	016626R	JSB OVCHK	SUFFICIENT PROGRAM SPACE LEFT?
C554	04551	016353R	JSB MVTUH	MAKE
C555	04552	026533R	JMP ACCS2	ROOM

CHECK SYNTAX AND TRANSLITERATE

C557**

C558*** FIND SEQUENTIAL POSITION **

C559**

C560	04553	000000	FNDPS	NOP	
C561	04554	072051R		STA TEMP3	SAVE SEQUENCE NUMBER
C562	04555	065007C		LDB PBUFF	STARTING ADDRESS
C563	04556	055010C	FNDP1	CPB PBPTR	END OF PROGRAM?
C564	04557	025601R		JMP FNDP4	YES, EXIT VIA (P+1)
C565	04560	060001		LDA 1	
C566	04561	017204R		JSB DREAD	SUBTRACT PROGRAM
C567	04562	003004		CMA, INA	SEQUENCE NUMBER FROM
C568	04563	042051R		ADA TEMP3	S-BUFFER SEQUENCE NUMBER
C569	04564	002003		SZA, RSS	EQUAL?
C570	04565	025577R		JMP FNDP2	YES, SET EXIT TO (P+3)
C571	04566	002020		SSA	NO, P-SEQ NO > S-SEQ NO ?
C572	04567	025600R		JMP FNDP3	YES, SET EXIT TO (P+2)
C573	04570	060001		LDA 1	POINT (A) TO
C574	04571	002004		INA	PROGRAM ADDRESS INCREMENT
C575	04572	073130R		STA IMP2	
C576	04573	017204R		JSB DREAD	
C577	04574	044000		ADB 0	COMPUTE NEW ADDRESS
C578	04575	063130R		LDA IMP2	
C579	04576	025556R		JMP FNDP1	
C580	04577	036553R	FNDP2	ISZ FNDPS	
C581	04600	036553R	FNDP3	ISZ FNDPS	
C582	04601	075051R	FNDP4	STB TEMP3	SAVE STATEMENT ADDRESS
C583	04602	125553R		JMP FNDPS, 1	

C584**

C585*** DELETE SPACE IN PROGRAM **

C586**

C587	04603	000000	CLPR6	NOP	REFERENCE LOCATION IN TEMP3
C588	04604	042051R		ADA TEMP3	SKIP (A) LOCATIONS FROM TEMP3
C589	04605	072052R		STA TEMP4	AND SAVE .DESTINATION ADDRESS
C590	04606	060001		LDA 1	
C591	04607	017204R		JSB DREAD	SKIP TO END OF STATEMENT BEING
C592	04610	070001		STA 1	
C593	04611	045051R		ADB TEMP3	DELETED, SOURCE ADDRESS IN (B)
C594	04612	055010C	CLPR1	CPB PBPTR	ALL OF PROGRAM MOVED?
C595	04613	025623R		JMP CLPR2	YES
C596	04614	060001		LDA 1	
C597	04615	017204R		JSB DREAD	NO, MOVE WORD FROM SOURCE TO
C598	04616	017151R		JSB DSTOR	.DESTINATION ADDRESS
C599	04617	000052R		DEF TEMP4	
C600	04620	035052R		ISZ TEMP4	INCREMENT .DESTINATION ADDRESS
C601	04621	005004		INB	INCREMENT SOURCE ADDRESS
C602	04622	025612R		JMP CLPR1	
C603	04623	062052R	CLPR2	LDA TEMP4	SET END-OF-PROGRAM
C604	04624	072010C		STA PBPTR	POINTER
C605	04625	125603R		JMP CLPR6, 1	

C606**

C607*** CHECK FOR PROGRAM SPACE OVERFLOW **

C608**

C609	04626	000000	OVCHK	NOP	NEW WORD REQUIREMENT IN (A)
C610	04627	065010C		LDB PBPTR	SET SOURCE ADDRESS
C611	04630	075050R		STB TEMP2	FOR PROGRAM RELOCATION

CHECK SYNTAX AND TRANSLITERATE

0612	04631	103101	CLO	
0613	04632	044000	ADB 0	SET .DESTINATION
0614	04633	076052R	STB TEMP4	ADDRESS
0615	04634	102201	SOC	ENOUGH FREE SPACE ?
0616	04635	027405R	JMP FSCE4	NO, PROGRAM SPACE OVERFLOW
0617	04636	066052R	LDB TEMP4	YES, RELOCATE FREE
0618	04637	076010C	STB PBPTR	PROGRAM SPACE POINTER
0619	04640	126626R	JMP OVCHK,I	

```

C621*
C622* *****
C623***      ***
C624*** LIST THE PROGRAM ***
C625***      ***
C626* *****
C627*
C628 04641 062010C LIST LDA PBPTR #
C629 04642 072001R STA LWBM+1
C630 04643 072272R STA PARM
C631 04644 065007C LDB PBUFF INITIALIZE TO FIRST
C632 04645 075045R STB TEMPS STATEMENT OF PROGRAM
C633 04646 017577R JSB GETCR SEQUENCE NUMBER GIVEN?
C634 04647 025707R JMP LIST0 NO
C635 04650 065016R LDB .BUFA YES, SET FOR
C636 04651 075022R STB SBPTR SEQUENCE NUMBER
C637 04652 015400R JSB INTCK FETCH
C638 04653 000256R DEF MAXSN IT
C639 04654 072045R STA TEMPS
C640 04655 162016R LDA .BUFA,I LOAD SEQUENCE NUMBER
C641 04656 015553R JSB FNDPS FIND INITIAL STATEMENT
C642 04657 027575R JMP RDYPT
C643 04660 000000 NOP SAVE
C644 04661 062045R LDA TEMPS
C645 04662 075045R STB TEMPS
C646 04663 052151R CPA .44
C647 04664 017577R JSB GETCR
C648 04665 025707R JMP LIST0
C649 04666 065016R LDB .BUFA
C650 04667 075022R STB SBPTR
C651 04670 015400R JSB INTCK
C652 04671 000256R DEF MAXSN
C653 04672 162016R LDA .BUFA,I
C654 04673 015553R JSB FNDPS
C655 04674 025707R JMP LIST0
C656 04675 025707R JMP LIST0
C657 04676 060001 LDA 1
C658 04677 005004 INB
C659 04700 073130R STA TMP2
C660 04701 060001 LDA 1
C661 04702 017204R JSB DREAD
C662 04703 064000 LDB 0
C663 04704 063130R LDA TMP2
C664 04705 040001 ADA 1
C665 04706 072001R STA LWBM+1
C666 04707 005400 LIST0 CLB HIGH-SPEED
C667 04710 055023R CPB IFLAG PUNCH?
C668 04711 025714R JMP LIST1 NO
C669 04712 062170R LDA B133 YES, EMIT
C670 04713 117131R JSB LISTR,1 LEADER
C671 04714 065045R LIST1 LDB TEMPS MORE
C672 04715 055001R CPB LWBM+1
C673 04716 027117R JMP LIS13 NO
C674 04717 003400 CCA INITIALIZE
C675 04720 042021R ADA SBUFA OUTPUT BUFFER

```

0676	04721	072017R	STA BADDR	POINTER
0677	04722	002400	CLA	INITIALIZE
0678	04723	072020R	STA CCNT	CHARACTER COUNT
0679	04724	062045R	LDA TEMPS	
0680	04725	017204R	JSB DREAD	OUTPUT
0681	04726	017132R	JSB OUTIN	SEQUENCE NUMBER
0682	04727	062271R	LDA BLANK	OUTPUT
0683	04730	017477R	JSB OUTCR	BLANK
0684	04731	035045R	ISZ TEMPS	FETCH
0685	04732	062045R	LDA TEMPS	
0686	04733	017204R	JSB DREAD	STATEMENT LENGTH
0687	04734	003004	CMA, INA	SET
0688	04735	002004	INA	WORD
0689	04736	073234R	STA SLWST	COUNTER
0690	04737	035045R	LIST3 ISZ TEMPS	MORE
0691	04740	037234R	ISZ SLWST	STATEMENT?
0692	04741	025752R	JMP LIST4	YES
0693	04742	065021R	LIST2 LDB SBUFA	OUTPUT
0694	04743	062020R	LDA CCNT	
0695	04744	117131R	JSB LISTR, I	STATEMENT
0696	04745	102501	LIA 01	.
0697	04746	012261R	AND HALF	
0698	04747	002002	SZA	IS SW #14 SET?
0699	04750	027575R	JMP R0YPT	YES, RETURN TO READY
0700	04751	025714R	JMP LIST1	
0701	04752	062045R	LIST4 LDA TEMPS	
0702	04753	017204R	JSB DREAD	
0703	04754	012220R	AND OPMSK	
0704	04755	002003	SZA, RSS	NULL OPERATOR?
0705	04756	025776R	JMP LIST5	YES
0706	04757	072050R	STA TEMP2	NO, SAVE OPERATOR
0707	04760	001727	ALF, ALF	SINGLE
0708	04761	064000	LDB 0	CHARACTER
0709	04762	042237R	ADA M21	
0710	04763	002021	SSA, RSS	OPERATOR?
0711	04764	027106R	JMP LIS12	NO
0712	04765	005000	BLS	YES
0713	04766	005004	INB	LOAD
0714	04767	045100R	ADB FDPBS	SYMBOL'S
0715	04770	160001	LDA 1, I	ASCII WORD
0716	04771	001727	ALF, ALF	ADJUST
0717	04772	010074	AND MSK0	CHARACTER
0718	04773	052143R	CPA .34	" ?
0719	04774	027125R	JMP LIS14	YES
0720	04775	017477R	JSB OUTCR	NO
0721	04776	062045R	LIST5 LDA TEMPS	
0722	04777	017204R	JSB DREAD	
0723	05000	012225R	AND OPMSK	SAVE
0724	05001	072051R	STA TEMP3	OPERAND
0725	05002	012223R	AND TYPFL	EXTRACT OPERAND TYPE
0726	05003	072362R	STA LFLAG	SET LFLAG FALSE
0727	05004	002020	SSA	FLAG BIT SET?
0728	05005	027041R	JMP LIST9	YES
0729	05006	002003	SZA, RSS	NO, NULL OPERAND?
0730	05007	025737R	JMP LIST3	YES

C731	05010	050062	CPA .7	FUNCTION	
C732	05011	027034R	JMP LIST8	YES	
C733	05012	040046	LIST6	ADA M5	
C734	05013	002020	SSA	LETTER-DIGIT COMBINATION?	
C735	05014	027017R	JMP LIST7	NO	
C736	05015	003400	CCA	YES, SET	
C737	05016	072362R	STA LFLAG	LFLAG FALSE	
C738	05017	062051R	LIST7	LDA TEMP3	
C739	05020	001727	ALF,ALF	RESTORE AND	
C740	05021	001722	ALF,RAL ::		
C741	05022	010073	AND B177	OUTPUT	
C742	05023	042160R	ADA B100		
C743	05024	017477R	JSB OUTCR	LETTER	
C744	05025	035362R	ISZ LFLAG	DIGIT FOLLOWS?	
C745	05026	025737R	JMP LIST3	NO	
C746	05027	062051R	LDA TEMP3	YES	
C747	05030	010062	AND .7	RESTORE	
C748	05031	042147R	ADA .43	DIGIT	
C749	05032	017477R	JSB OUTCR	OUTPUT DIGIT	
C750	05033	025737R	JMP LIST3		
C751	05034	062162R	LIST8	LDA F	OUTPUT
C752	05035	017477R	JSB OUTCR	'F'	
C753	05036	062166R	LDA N	OUTPUT	
C754	05037	017477R	JSB OUTCR	'N'	
C755	05040	027017R	JMP LIST7		
C756	05041	022263R	LIST9	XOR FLGBT	
C757	05042	002102	CLE,SZA	NUMBER?	
C758	05043	027064R	JMP LIS10	NO	
C759	05044	035045R	ISZ TEMPS	YES	
C760	05045	072041R	STA SIGN	SET SIGN FLAG FALSE	
C761	05046	062045R	LDA TEMPS		
C762	05047	017204R	JSB DREAD		
C763	05050	035045R	ISZ TEMPS		
C764	05051	073130R	STA TMP2		
C765	05052	062045R	LDA TEMPS		
C766	05053	017204R	JSB DREAD		
C001	05054	064000	LDB 0		
C002	05055	063130R	LDA TMP2		
C003	05056	037234R	ISZ SLWST		
C004	05057	037234R	ISZ SLWST		
C005	05060	002120	CLE,SSA	NEGATIVE NUMBER?	
C006	05061	002300	CCE	YES, SET SIGN FLAG TRUE	
C007	05062	015041R	JSB NUMOT		
C008	05063	025737R	JMP LIST3		
C009	05064	050056	LIS10	CPA .3	INTEGER?
C010	05065	027100R	JMP LIS11	YES	
C011	05066	050062	CPA .7	NO,FUNCTION	
C012	05067	002001	RSS	YES	
C013	05070	027012R	JMP LIST6	NO, MUST BE A PARAMETER	
C014	05071	062051R	LDA TEMP3	COMPUTE	
C015	05072	001722	ALF,RAL	PRINT	
C016	05073	012220R	AND OPMSK	TABLE	
C017	05074	072050R	STA TEMP2	CODE	
C018	05075	066122R	LDB ATAB	OUTPUT	
C019	05076	017216R	JSB MCOUT	FUNCTION NAME	

C020	05077	025737R	JMP LIST3	
C021	05100	036045R	LIS11 ISZ TEMPS	OUTPUT
C022	05101	037234R	ISZ SLWST	
C023	05102	062045R	LDA TEMPS	
C024	05103	017204R	JSB DREAD	INTEGER
C025	05104	017132R	JSB OUTIN	
C026	05105	025737R	JMP LIST3	OPERAND
C027	05106	062271R	LIS12 LDA BLANK	OUTPUT
C028	05107	017477R	JSB OUTCR	BLANK
C029	05110	066106R	LDB STIYP	OUTPUT
C030	05111	017216R	JSB MCOOT	OPERATOR
C031	05112	062212R	LDA REMOP	WAS IT
C032	05113	052050R	CPA TEMP2	A REM?
C033	05114	027173R	JMP OUTS1	YES, OUTPUT REMARK
C034	05115	062271R	LDA BLANK	NO, OUTPUT
C035	05116	025775R	JMP LIST5-1	A BLANK
C036	05117	006400	LIS13 CLB	HIGH-SPEED
C037	05120	036023R	CPB TFLAG	PUNCH?
C038	05121	027575R	JMP RDYPT	NO
C039	05122	062170R	LDA B133	YES, EMIT
C040	05123	117131R	JSB LISTR,1	TRAILER
C041	05124	027575R	JMP RDYPT	
C042	05125	017477R	LIS14 JSB OUTCR	OUTPUT "
C043	05126	017172R	JSB OUTST	OUTPUT QUOTE STRING
C044	05127	062143R	LDA .34	OUTPUT
C045	05130	025775R	JMP LIST5-1	
C046	05131	000000	LISTR NOP	
C047*			*	
C048**	OUTPUT AN INTEGER		**	
C049*			*	
C050	05132	000000	OUTIN NOP	INTEGER IN (A)
C051	05133	064047	LDB M4	SET
C052	05134	077701R	STB DIGCT	DIGIT COUNTER
C053	05135	067251R	LDB LDVSR	SET DIVISOR
C054	05136	076400R	STB DIVSR	ADDRESS
C055	05137	006400	CLB	SET LEADING
C056	05140	076626R	SIB LDZRO	ZERO FLAG
C057	05141	166400R	OUTI1 LDB DIVSR,1	NEGATE
C058	05142	007004	CMB,INB	AND STORE
C059	05143	076553R	STB MIND	DIVISOR
C060	05144	007400	CCB	SET QUOTIENT
C061	05145	006004	INB	TO ZERO
C062	05146	042553R	ADA MIND	SUBTRACT DIVISOR FROM INTEGER
C063	05147	002021	SSA,RSS	NEGATIVE RESULT?
C064	05150	027145R	JMP *-3	NO, INCREMENT QUOTIENT
C065	05151	142400R	ADA DIVSR,1	YES, RECOVER REMAINDER
C066	05152	073216R	STA MCOOT	AND SAVE IT
C067	05153	060001	LDA 1	
C068	05154	002002	SZA	ZERO?
C069	05155	027160R	JMP OUTI2	NO
C070	05156	052626R	CPA LDZRO	YES, LEADING ZERO?
C071	05157	027163R	JMP OUTI3	YES
C072	05160	042154R	OUTI2 ADA .48	NO, COMPUTE ASCII FOR DIGIT
C073	05161	072626R	STA LDZRO	SET 'ZEROES SIGNIFICANT'
C074	05162	017477R	JSB OUTCR	OUTPUT DIGIT

C075	05163	063216R	OUT13	LDA	MCOUT	RETRIEVE REMAINDER
C076	05164	066400R		ISZ	DIVSR	SET FOR NEXT DIVISOR
C077	05165	067701R		ISZ	DIGCT	DIVISION NECESSARY?
C078	05166	027141R		JMP	OUT11	YES
C079	05167	042154R		ADA	.48	NO, COMPUTE ASCII FOR LAST
C080	05170	017477R		JSB	OUTCR	DIGIT AND OUTPUT IT
C081	05171	127132R		JMP	OUTIN,1	
C082*				*		
C083**	OUTPUT A STRING				**	
C084*				*		
C085	05172	000000	OUTST	NOP		" ENTRY POINT
C086	05173	062045R	OUTS1	LDA	TEMPS	REM ENTRY POINT
C087	05174	017204R		JSB	DREAD	
C088	05175	010073		AND	B177	OUTPUT SECOND CHARACTER
C089	05176	002002		SZA		OF WORD IF
C090	05177	017477R		JSB	OUTCR	NOT NULL
C091	05200	066045R		ISZ	TEMPS	BUMP POINTER
C092	05201	037234R		ISZ	SLWST	REM COMPLETED?
C093	05202	002001		RSS		NO
C094	05203	026742R		JMP	LIST2	YES
C095	05204	062045R		LDA	TEMPS	
C096	05205	017204R		JSB	DREAD	EXTRACT
C097	05206	001727		ALF,ALF		FIRST CHARACTER
C098	05207	010073		AND	B177	OF WORD
C099	05210	050054		CPA	.1	EXIT
C100	05211	127172R		JMP	OUTST,1	IF A
C101	05212	052245R		CPA	STCON	CLOSING
C102	05213	127172R		JMP	OUTST,1	QUOTE
C103	05214	017477R		JSB	OUTCR	OUTPUT
C104	05215	027173R		JMP	OUTS1	CHARACTER
C105*				*		
C106**	LIST A MULTICHARACTER SYMBOL				**	
C107*				*		
C108	05216	000000	MCOUT	NOP		
C109	05217	160001	MCOU1	LDA	1,1	LOAD INFORMATION WORD
C110	05220	012220R		AND	OPMSK	COMPARE WITH
C111	05221	052050R		CPA	TEMP2	OPERATOR CODE
C112	05222	027231R		JMP	MCOU2	EQUAL
C113	05223	160001		LDA	1,1	UNEQUAL,
C114	05224	010070		AND	.15	COMPUTE
C115	05225	040056		ADA	.3	ENTRY
C116	05226	001100		ARS		LENGTH
C117	05227	044000		ADB	0	COMPUTE ADDRESS OF NEXT ENTRY
C118	05230	027217R		JMP	MCOU1	
C119	05231	160001	MCOU2	LDA	1,1	COMPUTE
C120	05232	010070		AND	.15	ENTRY
C121	05233	003004		CMA,INA		LENGTH
C122	05234	073701R		STA	DIGCT	AND SAVE IT
C123	05235	006104		CLE,INB		SET FOR FIRST CHARACTER
C124	05236	076051R		STB	TEMP3	SAVE SYMBOL ADDRESS
C125	05237	162051R	MCOU3	LDA	TEMP3,1	LOAD WORD
C126	05240	002041		SEZ,RSS		FIRST CHARACTER?
C127	05241	001727		ALF,ALF		YES, POSITION IT
C128	05242	010073		AND	B177	EXTRACT CHARACTER
C129	05243	017477R		JSB	OUTCR	OUTPUT IT

C130	05244	002240	SEZ,CME	SET FOR NEXT CHARACTER
C131	05245	036051R	ISZ TEMP3	MOVE TO NEXT WORD OF SYMBOL
C132	05246	037701R	ISZ DIGCT	MORE CHARACTERS?
C133	05247	027237R	JMP MC003	YES
C134	05250	127216R	JMP MCOUT,1	
C135*				
C136*				
C137	05251	005252R	LDVSR DEF *+1	
C138	05252	023420	DEC 10000	
C139	05253	001750	DEC 1000	
C140	05254	000144	DEC 100	
C141	05255	000012	DEC 10	
C142*				
C143*				
C144	03534		SFLAG EQU ARRY5	
C145	04362		TABLE EQU PRGIN	
C146	00055		LNGTH EQU TEMPS+8	
C147	04400		SMEND EQU INTCK	
C148	04626		SLENG EQU OVCHK	
C149	04553		TBLPT EQU FNDPS	
C150	04603		TSPTK EQU CLPRG	
C151	04626		INTGR EQU OVCHK	
C152	04362		LFLAG EQU PRGIN	
C153	04400		DIVSR EQU INTCK	
C154	04626		LDZRO EQU OVCHK	
C155	04553		MIND EQU FNDPS	

```

C157*          *****
C158*          PHASE 2 OF THE COMPILER
C159*          *****
C160*
C161*          THIS PHASE HAS THE FOLLOWING 3 FUNCTIONS:
C162*          1. SYMBOL TABLE CONSTRUCTION
C163*          2. FOR LOOP CHECKING
C164*          3. ARRAY STORAGE ALLOCATION
C165*
C166 05256 062010C MFASE LDA PBPTR      NULL
C167 05257 072007R      STA FDISK      ADDR. OF START OF ARRAY STORAGE
C168 05260 052007C      CPA PBUFF      PROGRAM?
C169 05261 027575R      JMP RDYPT      YES
C170 05262 062013R      LDA ASBTB
C171 05263 072006R      STA FCORE      NO, SET FOR-TABLE POINTER
C172 05264 062002R      LDA FWAM
C173 05265 072056R      STA COML      INITIALIZE COMMON POINTER
C174 05266 062011R      LDA SYMTA
C175 05267 072010R      STA SYMTF      INITIALIZE SYMBOL TABLE POINTER
C176 05270 062007C      LDA PBUFF
C177 05271 072022R      STA MPTR      INITIALIZE PROGRAM POINTER
C178 05272 062022R MLOP1 LDA MPTR
C179 05273 017204R      JSB DREAD
C180 05274 064000      LDB A
C181 05275 075033R      STB .LNUM      SET LINE NUMBER
C182 05276 065022R      LDB MPTR
C183 05277 035022R      ISZ MPTR
C184 05300 062022R      LDA MPTR
C185 05301 017204R      JSB DREAD
C186 05302 044000      ADB A      COMPUTE LOCATION OF NEXT
C187 05303 077266R      STB MNPTR      STATEMENT AND STORE THIS
C188 05304 035022R      ISZ MPTR
C189 05305 062022R      LDA MPTR
C190 05306 017204R      JSB DREAD      FETCH THE FIRST WORD IN THE
C191 05307 001727 MLO10 ALF,ALF      STATEMENT AND ITS TYPE
C192 05310 010075      AND B177      :::
C193 05311 072034R      STA TYPE
C194 05312 052152R      CPA .46      MAT STATEMENT?
C195 05313 027325R      JMP MLO12      YES
C196 05314 052140R      CPA .30      NO, REM STATEMENT?
C197 05315 075022R      STB MPTR      YES, SET TO SKIP IT
C198 05316 052156R      CPA .50      WRITE STATEMENT?
C199 05317 027321R      JMP *+2      YES, SET TO SKIP IT
C200 05320 052147R      CPA .43      NO, PRINT STATEMENT?
C201 05321 075022R      STB MPTR      YES, SET TO SKIP IT
C202 05322 003400      CCA      NO, SET
C203 05323 072057R      STA MWDND      'FIRST VARIABLE'
C204 05324 027341R      JMP MLOP2+1      FLAG
C205*
C206 05325 062022R MLO12 LDA MPTR      SEEK
C207 05326 002004      INA      SUBSIDIARY
C208 05327 017204R      JSB DREAD      STATEMENT
C209 05330 027307R      JMP MLO10      TYPE
C210*
C211 05331 012176R MLO13 AND MSK1      YES, ISOLATE OPERAND

```

0212	05332	066022R	LDB MPTR	INDEX THE PROGRAM POINTER BY
0213	05333	002003	SZA, RSS	AN AMOUNT APPROPRIATE TO THE
0214	05334	044055	ADB .2	OPERAND. THE FOLLOWING APPLIES
0215	05335	050056	CPA .3	OPERAND = 0 ADD 2 TO POINTER
0216	05336	006004	INB	OPERAND =3 ADD 1 TO POINTER
0217	05337	076022R	STB MPTR	
0218*				
0219	05340	036022R	MLOP2 ISZ MPTR	INCREMENT WORD-OF-STATEMENT PTR
0220	05341	062022R	LDA MPTR	STATEMENT
0221	05342	053266R	CPA MNPTR	EXHAUSTED?
0222	05343	027433R	JMP MLOP5	YES
0223	05344	062022R	LDA MPTR	
0224	05345	017204R	JSB DREAD	GET STATEMENT TYPE
0225	05346	012220R	AND UPMSK	
0226	05347	052215R	CPA PRNOP	PRINT?
0227	05350	027431R	JMP MLO14	YES, SKIP TO NEXT STATEMENT
0228	05351	052214R	CPA WRUP	WRITE?
0229	05352	027431R	JMP MLO14	YES, SKIP TO NEXT STATEMENT
0230	05353	062022R	LDA MPTR	NO
0231	05354	017204R	JSB DREAD	
0232	05355	002020	SSA	'CONSTANT' OPERAND?
0233	05356	027331R	JMP MLO13	YES
0234	05357	012176R	AND MSK1	NO
0235	05360	002003	SZA, RSS	NULL OPERAND?
0236	05361	027340R	JMP MLOP2	YES
0237	05362	072045R	STA MBOX1	NO, SAVE IT
0238	05363	010062	AND .7	PROGRAMMER-DEFINED
0239	05364	050062	CPA .7	FUNCTION?
0240	05365	027442R	JMP MLOP6	YES
0241	05366	040047	ADA M4	NO
0242	05367	002020	SSA	ARRAY VARIABLE?
0243	05370	027463R	JMP MLOP7	YES
0244	05371	062045R	LDA MBOX1	NO, SIMPLE VARIABLE
0245	05372	017613R	JSB SSYMT	ALREADY IN
0246	05373	006021	SSB, RSS	SYMBOL TABLE?
0247	05374	027403R	JMP MLOP3	YES
0248	05375	062263R	LDA MNEG	NO
0249	05376	066264R	LDB MNEG+1	ENTER
0250	05377	072046R	STA MBOX1+1	IT WITH
0251	05400	076047R	STB MBOX1+2	'UNDEFINED'
0252	05401	060050	LDA M3	VALUE
0253	05402	017660R	JSB ESYMT	
0254	05403	066034R	MLOP3 LDB TYPE	
0255	05404	062045R	LDA MBOX1	
0256	05405	055143R	CPB .34	NEXT STATEMENT?
0257	05406	027420R	JMP MLOP4	YES
0258	05407	055142R	CPB .33	NO, FOR STATEMENT?
0259	05410	036057R	ISZ MWONU	YES, FIRST VARIABLE?
0260	05411	027340R	JMP MLOP2	NO
0261	05412	036006R	ISZ FCORE	DEMAND
0262	05413	066006R	LDB FCORE	SPACE
0263	05414	055010R	CPB SYMTF	FOR NEW
0264	05415	027670R	JMP MER8-1	ENTRY
0265	05416	172006R	STA FCORE,1	SAVE VARIABLE NAME
0266	05417	027340R	JMP MLOP2	

C267*					
C268	05420	065006R	MLOP4	LDB FCORE	FOR-TABLE
C269	05421	055013R		CPB ASBTB	EMPTY?
C270	05422	016274R		JSB ERROR	YES
C271	05423	152006R	MER3	CPA FCORE,I	NO, MATCH LATEST ENTRY?
C272	05424	002001		RSS	YES
C273	05425	027422R		JMP MER3-1	NO
C274	05426	044052		ADB M1	REMOVE
C275	05427	075006R		STB FCORE	MATCHED
C276	05430	027340R		JMP MLOP2	ENTRY
C277*					
C278	05431	063266R	MLU14	LDA MNPTR	
C279	05432	072022R		STA MPTR	
C280	05433	052010C	MLOP5	CPA PBPTR	PROGRAM EXHAUSTED?
C281	05434	002001		RSS	YES
C282	05435	027272R		JMP MLOP1	NO
C283	05436	062034R		LDA TYPE	YES
C284	05437	052144R		CPA .37	END STATEMENT?
C285	05440	027561R		JMP M1LOP	YES
C286	05441	016274R		JSB ERROR	NO
C287	05442	062022R	MLOP6	LDA MPTR	ISOLATE
C288	05443	017204R		JSB DREAD	
C289	05444	012220R		AND OPMSK	PRECEDING OPERATOR
C290	05445	052211R		CPA DEFOP	'DEF' ?
C291	05446	002001		RSS	YES
C292	05447	027340R		JMP MLOP2	NO GO TO PROCESS NEXT WORD
C293	05450	062045R		LDA MBOX1	SEARCH SYMBOL TABLE FOR
C294	05451	017613R		JSB SSYMT	THE FUNCTION
C295	05452	006021		SSB,RSS	
C296	05453	016274R		JSB ERROR	FOUND. ERROR MULTIPLY DEFINED
C297	05454	062022R	MER4	LDA MPTR	
C298	05455	040056		ADA .3	ENTER THE FUNCTION INTO THE
C299	05456	032263R		IOR FLGHT	ADD V.A. FLAG
C300	05457	072046R		STA MBOX1+1	SYMBOL TABLE TOGETHER WITH
C301	05460	060051		LDA M2	ITS ENTRY POINT IN THE SOURCE
C302	05461	017660R		JSB ESYMT	CODE
C303	05462	027340R		JMP MLOP2	GO TO PROCESS THE NEXT WORD
C304*					
C305	05463	070001	MLOP7	STA 1	
C306	05464	062034R		LDA TYPE	
C307	05465	052136R		CPA .27	DIM STATEMENT?
C308	05466	027500R		JMP MLOP8	YES
C309	05467	052137R		CPA .28	NO, COM STATEMENT?
C310	05470	027500R		JMP MLOP8	YES
C311	05471	017701R		JSB MSYMT	NO, LOOK FOR IT IN SYMBOL TABLE
C312	05472	027340R		JMP MLOP2	FOUND
C313	05473	002400		CLA	NOT THERE
C314	05474	072046R		STA MBOX1+1	ENTER IT WITH
C315	05475	072047R		STA MBOX1+2	DIMENSIONS AND
C316	05476	072050R		STA MBOX1+3	DIMENSIONALITY
C317	05477	027542R		JMP MLOP0	UNDEFINED
C318*					
C319	05500	035022R	MLOP8	ISZ MPTR	PROCESS COM OR DIM STMT
C320	05501	035022R		ISZ MPTR	
C321	05502	062022R		LDA MPTR	PICK UP FIRST DIMENSION

PRE-EXECUTION PROCESSING

```

0322 05503 017204R JSB DREAD
0323 05504 001727 ALF,ALF SHIFT TO M. S. PART OF WORD
0324 05505 054050 CPB M3 IS THIS A SINGLE DIMENSION ARRAY
0325 05506 027516R JMP *+8 YES, JUMP
0326 05507 036022R ISZ MPTR NO, INDEX POINTER TO THE LOC.
0327 05510 036022R ISZ MPTR OF SECOND DIMENSION AND PACK
0328 05511 073130R STA TMP2 INTO A WITH
0329 05512 062022R LDA MPTR
0330 05513 017204R JSB DREAD
0331 05514 033130R IOR TMP2 THE FIRST DIMENSION
0332 05515 002001 RSS
0333 05516 030054 IOR .1
0334 05517 072047R STA MBOX1+2 SET UP TO STORE PACKED
0335 05520 072050R STA MBOX1+3 DIMENSIONS IN FORMAL AND ACTUAL
0336 05521 002400 CLA SLOTS AND UNDEFINED FLAG IN
0337 05522 072046R STA MBOX1+1 STORAGE ALLOCATION SLOT
0338 05523 017701R JSB MSYMT IN SYMBOL TABLE?
0339 05524 027545R JMP MLOP9 NO
0340 05525 062034R LDA TYPE YES
0341 05526 052137R CPA .28
0342 05527 002001 RSS IS STMT A COM
0343 05530 027542R JMP MLOP0 NO, JUMP
0344 05531 062047R LDA MBOX1+2 YES PICK UP PACKED DIMENSIONS
0345 05532 017130R JSB MDIM COMPUTE STORAGE REQUIRED
0346 05533 064000 LDB A
0347 05534 062056R LDA COML POINTER TO NEXT FREE LOC IN COM
0348 05535 032263R IOR FLGHT ADD V.A. FLAG
0349 05536 072046R STA MBOX1+1 STORE IN STORAGE ALLOCATION SLOT
0350 05537 012222R AND INF
0351 05540 040001 ADA B
0352 05541 072056R STA COML UPDATE PTR BY AMT OF STORAGE REQ.
0353 05542 060047 MLOP0 LDA M4 ENTER THE FOUR WORD ENTRY
0354 05543 017660R JSB ESYMT PREVIOUSLY SET UP IN MBOX1 INTO
0355 05544 027340R JMP MLOP2 SYMBOL TABLE AND CONTINUE
0356*
0357 05545 044055 MLOP9 ADB .2 CHECK THE FORMAL DIMENSIONS
0358 05546 160001 LDA 1,1 LOCATION TO SEE IF THE DIMENSION
0359 05547 002002 SZA IS ALREADY DEFINED
0360 05550 016274R JSB ERROR ERROR, DOUBLY DIMENSIONED
0361 05551 062034R MERS LDA TYPE
0362 05552 052137R CPA .28 COM STMT?
0363 05553 025331R JMP SYNE3-1 ERROR MISPLACED COM. STMT.
0364 05554 062047R LDA MBOX1+2
0365 05555 170001 STA 1,1 STORE THESE DIMENSIONS IN FORMAL
0366 05556 006004 INB AND ACTUAL SLOTS IN SYMBOL TABLE
0367 05557 170001 STA 1,1 ENTRY
0368 05560 027340R JMP MLOP2 GO TO PROCESS NEXT WORD
0369*
0370* THE SECTION WHICH FOLLOWS CHECKS
0371* THAT ALL FOR LOOPS HAVE BEEN
0372* TERMINATED, ASSIGNS THE STANDARD
0373* DIMENSIONS TO UNDIMENSIONED ARRAYS
0374* AND MAKES STORAGE ASSIGNMENTS FOR
0375* ALL ARRAYS WHICH DO NOT APPEAR IN
0376* A COM STMT

```

C377*					
C378	05561	062006R	M1LOP	LDA FCORE	ALL FORS
C379	05562	052013R		CPA ASBTB	MATCHED?
C380	05563	002001		RSS	YES
C381	05564	015274R		JSB ERROR	NO
C382	05565	065010R	MER6	LDB SYMTF	
C383*					
C384	05566	055011R	M2LOP	CPB SYMTA	MORE SYMBOLS?
C385	05567	027642R		JMP M4LOP	NO
C386	05570	160001		LDA 1,I	YES
C387	05571	010062		AND .7 ::	ACCOUNT FOR
C388	05572	044055		ADB .2	A FUNCTION
C389	05573	050062		CPA .7 ::	IS IT?
C390	05574	027566R		JMP M2LOP	YES
C391	05575	005004		INB	NO, ACCOUNT FOR
C392	05576	040047		ADA M4	SIMPLE VARIABLE
C393	05577	002025		SSA,INA,RSS	IS IT?
C394	05600	027566R		JMP M2LOP	YES
C395	05601	002003		SZA,RSS	NO, # OF SUBSCRIPTS KNOWN?
C396	05602	015274R		JSB ERROR	NO
C397	05603	002004	MER10	INA	SAVE
C398	05604	072046R		STA MBOX1+1	FLAG
C399	05605	075045R		STB MBOX1	SAVE POINTER
C400	05606	160001		LDA 1,I	DEFINED
C401	05607	002002		SZA	ARRAY?
C402	05610	027617R		JMP M3LOP	YES
C403	05611	063557R		LDA STDIM	NO, LOAD
C404	05612	035046R		ISZ MBOX1+1	APPROPRIATE
C405	05613	040064		ADA .9	STANDARD DIMENSIONS
C406	05614	170001		STA 1,I	RECORD AS
C407	05615	044052		ADB M1	FORMAL AND ACTUAL
C408	05616	170001		STA 1,I	DIMENSIONS
C409	05617	017130R	M3LOP	JSB MDIM	SAVE STORAGE
C410	05620	072046R		STA MBOX1+1	REQUIREMENT
C411	05621	065045R		LDB MBOX1	LOAD
C412	05622	044051		ADB M2	ADDRESS OF
C413	05623	160001		LDA 1,I	ELEMENT SPACE
C414	05624	012222R		AND INF	REMOVE V.A. BIT
C415	05625	050055		CPA .2	DEFINED IN COM?
C416	05626	027640R		JMP MER7	YES
C417	05627	062007R		LDA FDISK	NO, USE CURRENT
C418	05630	032263R		IOR FLGBT	ADD V.A. FLAG
C419	05631	170001		STA 1,I	FREE-CORE ADDRESS
C420	05632	012222R		AND INF	
C421	05633	103101		CLO	
C422	05634	042046R		ADA MBOX1+1	UPDATE FREE-CORE
C423	05635	072007R		STA FDISK	ADDRESS
C424	05636	102201		SOC	OUT OF SPACE ?
C425	05637	015274R		JSB ERROR	YES
C426	05640	044056	MER7	ADB .3	NO, ADVANCE POINTER
C427	05641	027566R		JMP M2LOP	TO NEXT ENTRY
C428*					
C429	05642	065010C	M4LOP	LDB PBPTR	INITIALIZE ALL
C430	05643	060001		LDA 1	
C431	05644	003000		CMA	

PRE-EXECUTION PROCESSING

0432	05645	042007R	ADA FDISK	ARRAY ELEMENTS
0433	05646	002020	SSA	
0434	05647	026173R	JMP XEC	TO 'UNDEFINED'
0435	05650	062263R	LDA MNEG	
0436	05651	017055R	JSB STBI	
0437	05652	005004	INB	
0438	05653	062264R	LDA MNEG+1	
0439	05654	017055R	JSB STBI	
0440	05655	005004	INB	
0441	05656	027643R	JMP M4LOP+1	
0442*				
0443	05657	005001	STOIM OCT 5001	

PRE-EXECUTION PROCESSING

```

C445*          *****
C446*          ENTER SYMBOL TABLE SUBROUTINE
C447*          *****
C448*
C449*          TRANSFER -(A) WORDS FROM THE BUFFER ADDRESSED
C450*          BY MBUF TO THE TOP OF THE SYMBOL TABLE.
C451*
C452 05660 000000 ESYMT NOP
C453 05661 073234R STA MBIN1      SAVE NEGATIVE OF LENGTH OF ENTRY
C454 05662 042010R ADA SYMTF
C455 05663 072010R STA SYMTF      MOVE SYMBOL TABLE START LOCATOR
C456 05664 073321R STA MBIN2      UP BY THE LENGTH OF ENTRY
C457 05665 003004 CMA,INA      CHECK THAT THE SYMBOL TABLE AND
C458 05666 042006R ADA FCORE      FOR TABLE DO NOT OVERLAP
C459 05667 002021 SSA,RSS
C460 05670 015274R JSB ERROR      OVERLAP ERROR
C461 05671 067722R MER8 LDB MBUF      POINTER TO REQD ENTRY
C462 05672 160001 LDA 1,I      TRANSFER ENTRY TO THE SYMBOL
C463 05673 173321R STA MBIN2,I TABLE
C464 05674 005004 INB
C465 05675 037321R ISZ MBIN2
C466 05676 037234R ISZ MBIN1
C467 05677 027672R JMP MER8+1
C468 05700 127660R JMP ESYMT,I RETURN
C469*
C470*          *****
C471*          SUBROUTINE TO SEARCH SYMBOL TABLE FOR AN ARRAY
C472*          *****
C473 05701 000000 MSYMT NOP      B GIVES ARRAY TYPE -3 = 1 DIM,
C474 05702 077234R STB MBIN1      -2 = 2DIM, -1 = UNDIMENSIONED
C475 05703 062045R LDA MBUX1      LOAD IDENTIFIER
C476 05704 017613R JSB SSYMT      SEARCH THE SYMBOL TABLE
C477 05705 005021 SSB,RSS
C478 05706 127701R JMP MSYMT,I FOUND, RETURN
C479 05707 037234R ISZ MBIN1      IF ARRAY UNDIMENSIONED
C480 05710 002001 RSS
C481 05711 027720R JMP MSYM      JUMP TO NOT FOUND EXIT
C482 05712 037234R ISZ MBIN1      SET UP TO CHECK THAT ARRAY DOES
C483 05713 040055 ADA .2      NOT APPEAR IN THE TABLE WITH
C484 05714 040052 ADA M1      DIFFERENT DIMENSIONS. CHANGE
C485 05715 017613R JSB SSYMT      TYPE 2 TO 1 & TYPE 1 TO 2 AND
C486 05716 005021 SSB,RSS      SEARCH AGAIN
C487 05717 015274R JSB ERROR      FOUND, INCONSISTENT DIMENSIONS
C488 05720 037701R MSYM ISZ MSYMT      NOT FOUND, INCREMENT RETURN
C489 05721 127701R JMP MSYMT,I ADDRESS AND RETURN
C490*
C491*
C492 05722 000045R MBUF DEF TEMPS
C493 00045 MBOX1 EQU TEMPS
C494 01234 MBIN1 EQU SLWST
C495 01321 MBIN2 EQU RSCHK
C496 00022 MPTR EQU SBPTR
C497 01256 MNPTR EQU OPCHK
C498 00056 COML EQU TEMPS+9
C499 00057 MWDNO EQU TEMPS+10

```

PRE-EXECUTION PROCESSING

C500 05701

DIGCT EQU MSYMT

```

C502*
C503***          ***
C504** EVALUATE A FORMULA **
C505***          ***
C506*
C507 05723 000000 FORMX NGP          FORMULA BEGINS IN (TEMPS)
C508 05724 005400          CLB          INITIALIZE OPERATOR
C509 05725 017234R          JSB SLWST          STACK
C510 05726 017062R FORM1 JSB LDIMP          FETCH OPERAND
C511 05727 035045R          ISZ TEMPS          SET FOR NEXT WORD OF FORMULA
C512 05730 012225R          AND UPDMK          EXTRACT OPERAND
C513 05731 072053R          STA TEMPS+6          AND SAVE IT
C514 05732 002003          SZA,RSS          NULL OPERAND?
C515 05733 027746R          JMP FORM2          YES
C516 05734 017243R          JSB BHSTP          SET STACK FOR OPERAND ADDRESS
C517 05735 002020          SSA          FLAG BIT SET?
C518 05736 025015R          JMP FORM4          YES
C519 05737 017613R          JSB SSYMT          FETCH OPERAND ADDRESS
C520 05740 005007          INB,SZB,RSS          EXISTANT?
C521 05741 027302R          JMP E8-1          NO
C522 05742 010062          AND .7          YES ::
C523 05743 050062          CPA .7          FUNCTION?
C524 05744 025033R          JMP FORM6          YES
C525 05745 175030R          STB HSIPT,I          NO, STACK OPERAND ADDRESS
C526 05746 017062R FORM2 JSB LDIMP          FETCH
C527 05747 012220R          AND UPMSK          OPERATOR
C528 05750 001727          ALF,ALF          POSITION IT
C529 05751 001200          RAL ::
C530 05752 064000          LDB 0          LOAD ADDRESS OF
C531 05753 045100R          ADB FOPBS          OPERATOR'S INFORMATION WORD
C532 05754 040043          ADA M8          NON-FORMULA
C533 05755 002020          SSA          OPERATOR?
C534 05756 005400          CLB          YES
C535 05757 042241R          ADA D53          NO, NON-FORMULA
C536 05760 002021          SSA,RSS          OPERATOR?
C537 05761 005400          CLB          YES
C538 05762 002400          CLA          NO
C539 05763 160001          LDA 1,1          LOAD INFORMATION WORD
C540 05764 012176R          AND MSK1          SAVE
C541 05765 072054R          STA TEMPS+7          PRECEDENCE
C542 05766 120001          XOR 1,1          SAVE
C543 05767 072053R          STA TEMPS+6          IDENTIFICATION
C544 05770 027776R          JMP FOR11
C545 05771 172026R FORM0 STA TSIPT,1          STACK HIGH WORD
C546 05772 062026R          LDA TSIPT          STACK OPERAND
C547 05773 172030R          STA HSIPT,I          ADDRESS
C548 05774 002004          INA          STORE
C549 05775 174000          STB 0,I          LOW WORD
C550 05776 162027R FOR11 LDA LSIPT,I          DOES OPERATOR
C551 05777 010074          AND MSK0          ON TOP OF
C552 06000 003000          CMA          OPERATOR STACK
C553 06001 042054R          ADA TEMPS+7          HAVE HIGHER
C554 06002 002020          SSA          PRECEDENCE?
C555 06003 025136R          JMP FORM9          YES, EXECUTE IT
C556 06004 002001          RSS          NO

```

C557	06005	035027R	FOR10	ISZ	LSTPT	
C558	06006	065054R		LDB	TEMPS+7	RETRIEVE PRECEDENCE
C559	06007	045235R		ADB	M15	NO, LEFT PARENTHESIS
C560	06010	005020		SSB		OR LEFT BRACKET?
C561	06011	044070		ADB	.15	NO, RESTORE PRECEDENCE
C562	06012	045053R		ADB	TEMPS+6	COMBINE IDENTIFICATION
C563	06013	017234R		JSB	SLWST	WITH PRECEDENCE AND STACK
C564	06014	027726R		JMP	FORM1	
C565	06015	052263R	FORM4	CPA	FLGBT	CONSTANT?
C566	06016	025024R		JMP	FORM5	YES
C567	06017	010062		AND	.7	NO, PREDIFINED
C568	06020	050062		CPA	.7	FUNCTION
C569	06021	025113R		JMP	FORM7	YES
C570	06022	065056R		LDB	TEMPS+9	NO, MUST BE A
C571	06023	027745R		JMP	FORM2-1	PARAMETER
C572	06024	065045R	FORM5	LDB	TEMPS	LOAD CONSTANT ADDRESS
C573	06025	035045R		ISZ	TEMPS	MOVE POINTER TO
C574	06026	035045R		ISZ	TEMPS	NEXT CODE WORD
C575	06027	060001		LDA	B	
C576	06030	032263R		IUR	FLGBT	ADD V.A. FLAG
C577	06031	064000		LDB	A	
C578	06032	027745R		JMP	FORM2-1	
C579	06033	075053R	FORM6	STB	TEMPS+6	SAVE SYMBOL TABLE POINTER
C580	06034	065026R		LDB	TSTPT	SAVE CURRENT POINTER
C581	06035	017234R		JSB	SLWST	TO TEMPORARY STACK
C582	06036	165053R		LDB	TEMPS+6, I	
C583	06037	017234R		JSB	SLWST	SAVE FUNCTION ADDRESS
C584	06040	063723R		LDA	FORMX	SAVE CURRENT
C585	06041	172030R		STA	HSTPT, I	FORMX RETURN ADDRESS
C586	06042	017723R		JSB	FORMX	EVALUATE THE PARAMETER
C587	06043	035045R		ISZ	TEMPS	UPDATE FORMULA POINTER
C588	06044	035045R		ISZ	TEMPS	PAST RIGHT PARENTHESIS
C589	06045	065045R		LDB	TEMPS	SWITCH
C590	06046	162027R		LDA	LSTPT, I	FORMULA POINTER
C591	06047	012222R		AND	INF	
C592	06050	072045R		STA	TEMPS	TO FUNCTION'S
C593	06051	175027R		STB	LSTPT, I	FORMULA
C594	06052	065056R		LDB	TEMPS+9	SET
C595	06053	162030R		LDA	HSTPT, I	PARAMETER POINTER
C596	06054	035027R		ISZ	LSTPT	TO NEW PARAMETER,
C597	06055	035030R		ISZ	HSTPT	SAVING PREVIOUS
C598	06056	175027R		STB	LSTPT, I	SETTING ON
C599	06057	072056R		STA	TEMPS+9	LOW-CORE STACK
C600	06060	052026R		CPA	TSTPT	PROTECT PARAMETER IF
C601	06061	017321R		JSB	RSCHK	ON TEMPORARY STACK
C602	06062	017723R		JSB	FORMX	EVALUATE FUNCTION
C603	06063	162027R		LDA	LSTPT, I	RESTORE OLD
C604	06064	072056R		STA	TEMPS+9	PARAMETER POINTER
C605	06065	065027R		LDB	LSTPT	CUT BACK
C606	06066	044050		ADB	M3	LOW-CORE
C607	06067	075027R		STB	LSTPT	STBCK
C608	06070	005004		INB		RESTORE ORIGINBL
C609	06071	160001		LDA	1, I	TEMPORARY STACK
C610	06072	072026R		STA	TSTPT	POINTER
C611	06073	005004		INB		RESTORE

EXECUTE THE PROGRAM

C612	06074	160001	LDA 1,I	ORIGINAL
C613	06075	012222R	AND INF	
C614	06076	072045R	STA TEMPS	FORMULA POINTER
C615	06077	101100	SWP	
C616	06100	017252R	JSB SITOP	POP RESULT

```

0618*
0619** PRE-DEFINED FUNCTIONS RETURN HERE WITH RESULT
0620*
0621 06101 172026R FOR12 STA TSTPT,I   STORE HIGH WORD
0622 06102 062026R      LDA TSTPT
0623 06103 002004      INA           STORE
0624 06104 174000      STB 0,I       LOW WORD
0625 06105 035030R     ISZ HSTPT
0626 06106 165030R     LDB HSTPT,I   RESTORE FORMX
0627 06107 077723R     STB FORMX    RETURN ADDRESS
0628 06110 040052      ADA M1       STACK ADDRESS
0629 06111 172030R     STA HSTPT,I   OF RESULT
0630 06112 027746R     JMP FORM2
0631 06113 062053R FORM7 LDA TEMPS+6  COMPUTE
0632 06114 001727      ALF,ALF
0633 06115 001722      ALF,RAL      FUNCTION
0634 06116 010071      AND .31
0635 06117 042104R     ADA PDFBS     ADDRESS
0636 06120 164000      LDB 0,I
0637 06121 017234R     JSB SLWST     SAVE FUNCTION ADDRESS
0638 06122 063723R     LDA FORMX    SAVE CURRENT
0639 06123 172030R     STA HSTPT,I   FORMX RETURN ADDRESS
0640 06124 017723R     JSB FORMX    EVALUATE THE PARAMETER
0641 06125 035045R     ISZ TEMPS    UPDATE FORMULA POINTER
0642 06126 035045R     ISZ TEMPS    PAST RIGHT PARENTHESIS
0643 06127 165027R     LDB LSTPT,I  POP
0644 06130 003400      CCA          FUNCTION
0645 06131 042027R     ADA LSTPT     ENTRY
0646 06132 072027R     STA LSTPT     ADDRESS
0647 06133 077660R     STB ESYMT    SAVE
0648 06134 017252R     JSB STIOP    POP PARAMETER
0649 06135 127660R     JMP ESYMT,I  EVALUATE FUNCTION
0650 06136 162027R FORM9 LDA LSTPT,I  UNSTACK
0651 06137 007400      CCB          OPERATOR
0652 06140 045027R     ADB LSTPT     INFORMATION
0653 06141 075027R     STB LSTPT    WORD
0654 06142 001727      ALF,ALF      COMPUTE
0655 06143 010073      AND B177     SUBROUTINE
0656 06144 042103R     ADA ARBAS     ADDRESS
0657 06145 124000      JMP 0,I      EXECUTE
0658**
0659*** EXECUTION BRANCH TABLE **
0660**
0661 06146 005406R XECTB DEF ELET     LET
0662 06147 005233R      DEF XEC4     DIM
0663 06150 005233R      DEF XEC4     COM
0664 06151 005233R      DEF XEC4     DEF
0665 06152 005233R      DEF XEC4     REM
0666 06153 005410R      DEF EGOTO    GO TO
0667 06154 005412R      DEF EIF      IF
0668 06155 005422R      DEF EFOR     FOR
0669 06156 005515R      DEF ENEXT    NEXT
0670 06157 005560R      DEF EGOSB    GOSUB
0671 06160 005571R      DEF ERTRN    RETURN
0672 06161 001575R      DEF RDYPT    END

```

```

0673 06162 001575R      DEF RDYPT      STOP
0674 06163 006600R      DEF EWAIT     WAIT
0675 06164 006617R      DEF ECALL     CALL
0676 06165 006233R      DEF XEC4      DATA
0677 06166 006647R      DEF EREAD     READ
0678 06167 006707R      DEF EPRIN     PRINT
0901 06170 007074R      DEF EINPT     INPUT
0902 06171 007112R      DEF ERSTR     RESTORE
0903 06172 011465R      DEF EMAT      MAT
0904*   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
0905*
0906* *****
0907***
0908*** EXECUTE THE PROGRAM ***
0909***
0910* *****
0911*
0912**
0913***INITIALIZE FOR OUTPUT **
0914**
0915 06173 002400 XEC    CLA              SET COUNTER FOR
0916 06174 072034R      STA TYPE          CHARACTERS OUTPUTTED
0917 06175 072043R      STA XH            INITIALIZE
0918 06176 002004      INA              RANDOM
0919 06177 072044R      STA XL            VARIABLE
0920 06200 060052      LDA M1            SET EXECUTION
0921 06201 073264R      STA XIT          EXIT FLAG
0922**
0923*** INITIALIZE THE DATA POINTER **
0924**
0925 06202 003400      CCA              SET
0926 06203 072037R      STA DCCNT        'NO
0927 06204 072035R      STA DSTRT        DATA'
0928 06205 066007C      LDB PBUFF        CONDITION
0929 06206 076036R      STB NXIDT
0930 06207 162115R      LDA ADATA,I      SEARCH FOR FIRST
0931 06210 016302R      JSB STSRH        DATA STATEMENT
0932 06211 026214R      JMP XEC2        NONE FOUND
0933 06212 076035R      STB DSTRT        SAVE STATEMENT LOCATION
0934 06213 016271R      JSB SETDP        SET DATA POINTER
0935**
0936*** INITIALIZE STACK POINTERS **
0937**
0938 06214 066010R XEC2  LDB SYMTF        INITIALIZE
0939 06215 076030R      STB HSTPT        POINTERS TO
0940 06216 066006R      LDB FCORE        'HIGH CORE' STACK,
0941 06217 076026R      STB TSTPT        'TEMPORARY'
0942 06220 046134R      ADB .23          STACK, AND
0943 06221 076012R      STB LSTAK        'LOW CORE'
0944 06222 076027R      STB LSTPT        STACK
0945 06223 007000      CM8              DO
0946 06224 046030R      ADB HSTPT        STACKS
0947 06225 006020      SSB              MEET?
0948 06226 027240R      JMP E1          YES
0949 06227 066226R      LDB RMODE      NO, SHIFT TO

```

C050	06230	077131R	STB LISTR	RUN MODE
C051	06231	066007C	LDB PBUFF	BEGIN
C052	06232	026246R	JMP XEC5	EXECUTION

```

C054**
C055*** FIND NEXT STATEMENT TO BE EXECUTED **
C056**
C057 06235 062032R XEC4 LDA NXTST NEXT STATEMENT NUMBER
C058 06234 066031R LDB PRADD PROSPECTIVE ADDRESS
C059 06235 072032R STA NXTST
C060 06236 017051R JSB LDBI
C061 06237 052032R CPA NXTST DESIRED STATEMENT?
C062 06240 026247R JMP XEC6 YES
C063 06241 066007C LDB PRUFF NO, FIND
C064 06242 062032R LDA NXTST
C065 06243 016553R JSB FNDPS STATEMENT
C066 06244 000000 NOP NON-EXISTENT
C067 06245 015274R JSB ERROR STATEMENT
C068 06246 017051R XEC5 JSB LDBI SAVE NEW
C069 06247 072033R XEC6 STA .LNUM SEQUENCE NUMBER
C070**
C071*** SET SUCCESSOR STATEMENT **
C072**
C073 06250 016352R JSB FLWST
C074 06251 012220R AND UPMSK EXTRACT STATEMENT TYPE
C075 06252 001727 ALF,ALF POSITION
C076 06253 042102R ADA XECBR COMPUTE EXECUTION ADDRESS
C077 06254 124000 JMP 0,I BRANCH TO EXECUTION CODE
C078**
C079*** EVALUATE FORMULA AND RETURN RESULT **
C080**
C081 06255 000000 FETCH NOP
C082 06256 017723R JSB FORMX EVALUATE FORMULA
C083 06257 017266R JSB UPCHK
C084 06260 036030R ISZ HSIPT UNSTACK RESULT ADDRESS
C085 06261 006020 SSB RESULT ON DISK?
C086 06262 025266R JMP FTCH2 YES, LOAD A AND B WITH RESULT
C087 06263 104200 DLD 1,I LOAD RESULT
C088 06265 126255R JMP FETCH,I EXIT
C089 06266 060001 FTCH2 LDA B
C090 06267 017505R JSB FTHOP LOAD RESULT
C091 06270 126255R JMP FETCH,I
C092**
C093*** SET POINTER TO START OF DATA STATEMENT **
C094**
C095 06271 000000 SETDP NOP STATEMENT ADDRESS IN (B)
C096 06272 006004 INB LOAD
C097 06273 017051R JSB LDBI STATEMENT LENGTH
C098 06274 003004 CMA,INA SET
C099 06275 002004 INA DATA COUNTER
C100 06276 072037R STA DCCNT TO 1-STATEMENT LENGTH
C101 06277 006004 INB SET 'NEXT DATA' POINTER ONE
C102 06300 076036R STB NXIDT WORD ABOVE FIRST CONSTANT
C103 06301 126271R JMP SETDP,I
C104**
C105*** SEARCH FOR STATEMENT OF GIVEN TYPE **
C106**
C107 06302 000000 STSRH NOP TYPE IN (A), ADDRESS IN (B)

```

C108	06303	012220R	AND UPMSK	EXTRACT
C109	06304	072052R	STA TEMP4	STATEMENT TYPE
C110	06305	060001	STSR1 LDA 1	EXTRACT
C111	06306	040055	ADA .2	PROGRAM
C112	06307	017204R	JSB DREAD	STATEMENT
C113	06310	012220R	AND UPMSK	TYPE
C114	06311	052052R	CPA TEMP4	DESIRED TYPE?
C115	06312	025322R	JMP STSR2	YES
C116	06313	060001	LDA 1	NO, FETCH
C117	06314	002004	INA	STATEMENT LENGTH
C118	06315	017204R	JSB DREAD	
C119	06316	044000	ADB A	COMPUTE NEW ADDRESS
C120	06317	056010C	CPB PBPTR	PAST LAST STATEMENT?
C121	06320	125302R	JMP STSRH,1	YES
C122	06321	025305R	JMP STSR1	NO
C123	06322	036302R	STSR2 ISZ STSRH	
C124	06323	125302R	JMP STSRH,1	
C125**				
C126***	FETCH A DATA ITEM **			
C127**				
C128	06324	000000	FDATA NOP	
C129	06325	035037R	FDAT1 ISZ DCCNT	MORE DATA?
C130	06326	025335R	JMP FDATA2	YES
C131	06327	162115R	LDA ADATA,1	NO, SEARCH
C132	06330	065036R	LDB NXTDT	FOR NEXT
C133	06331	015302R	JSB STSRH	DATA STATEMENT
C134	06332	015274R	JSB ERROR	NONE FOUND
C135	06333	015271R	E4 JSB SETDP	INITIALIZE THE
C136	06334	025325R	JMP FDATA1	DATA POINTERS
C137	06335	035037R	FDATA2 ISZ DCCNT	UPDATE
C138	06336	035037R	ISZ DCCNT	POINTER
C139	06337	035036R	ISZ NXTDT	
C140	06340	062036R	LDA NXTDT	
C141	06341	017204R	JSB DREAD	LOAD
C142	06342	035036R	ISZ NXTDT	DATA
C143	06343	073130R	STA TMP2	
C144	06344	062036R	LDA NXTDT	
C145	06345	017204R	JSB DREAD	
C146	06346	064000	LDB A	ITEM
C147	06347	063130R	LDA TMP2	
C148	06350	035036R	ISZ NXTDT	UPDATE POINTER
C149	06351	125324R	JMP FDATA,1	
C150**				
C151***	SET FOR FOLLOWING STATEMENT **			
C152**				
C153	06352	000000	FLWST NOP	(B) HOLDS PRESENT ADDRESS
C154	06353	060001	LDA 1	COMPUTE
C155	06354	002004	INA	ADDRESS
C156	06355	017204R	JSB DREAD	OF
C157	06356	040001	ADA 1	NEXT
C158	06357	072031R	STA PRADD	STATEMENT
C159	06360	017204R	JSB DREAD	RECORD THE
C160	06361	072032R	STA NXTST	SEQUENCE NUMBER
C161	06362	044055	ADB .2	FETCH
C162	06363	075045R	STB TEMPS	FIRST WORD

EXECUTE THE PROGRAM

```

0163 06364 017051R JSB LDB1 OF CURRENT
0164 06365 125352R JMP FLWST,I STATEMENT
0165**
0166*** SEARCH STACK FOR GIVEN FOR-VARIABLE **
0167**
0168 06366 000000 FVSRH NOP
0169 06367 017062R JSB LDIMP FETCH
0170 06370 012176R AND MSK1 FOR-VARIABLE
0171 06371 073440R STA EDELM SAVE FOR-VARIABLE
0172 06372 017613R JSB SSYMT FIND ADDRESS IN
0173 06373 005004 INB SYMBOL TABLE
0174 06374 062030R LDA HSTPT SAVE
0175 06375 072051R STA TEMP3 STACK TOP
0176 06376 052010R FVSR1 CPA SYMTF STACK BOTTOM?
0177 06377 125366R JMP FVSRH,I YES, EXIT VIA (P+1)
0178 06400 154000 CPB 0,1 MATCHING FOR-VARIABLE?
0179 06401 025404R JMP FVSR2 YES
0180 06402 040061 ADA .6 NO, MOVE TO
0181 06403 025376R JMP FVSR1 NEXT STACK ENTRY
0182 06404 035366R FVSR2 ISZ FVSRH EXIT
0183 06405 125366R JMP FVSRH,I VIA (P+2)
0184*
0185*** ***
0186** EXECUTE LET **
0187*** ***
0188*
0189 06406 017723R ELET JSB FORMX
0190 06407 025233R JMP XEC4
0191*
0192*** ***
0193** EXECUTE GO TO **
0194*** ***
0195*
0196 06410 017604R EGOTO JSB NXSQN LOAD SEQUENCE NUMBER
0197 06411 025234R JMP XEC4+1 FIND REFERENCED STATEMENT
0198*
0199*** ***
0200** EXECUTE IF **
0201*** ***
0202*
0203 06412 015255R EIF JSB FETCH FETCH VALUE OF FORMULA
0204 06413 002003 SZA,RSS RESULTANT TRUE?
0205 06414 025233R JMP XEC4 NO
0206 06415 035045R ISZ TEMPS YES, BRANCH TO
0207 06416 035045R ISZ TEMPS
0208 06417 035045R LD3 TEMPS FOLLOWING
0209 06420 017062R JSB LDIMP FOLLOWING STATEMENT
0210 06421 025251R JMP XSA AND EXECUTE
0211*
0212*** ***
0213** EXECUTE FOR **
0214*** ***
0215*
0216 06422 015366R EFOR JSB FVSRH FOR-VARIABLE ALREADY IN STACK?
0217 06423 025434R JMP EFOR1 NO

```

0218	06424	072050R	STA TEMP2	YES, SAVE SOURCE ADDRESS
0219	06425	040061	ADA .6	SAVE
0220	06426	072052R	STA TEMP4	.DESTINATION ADDRESS
0221	06427	075047R	STB TEMP1	SAVE FOR-VARIABLE ADDRESS
0222	06430	015367R	JSB MVTOC	COMPRESS STACK
0223	06431	065047R	LDB TEMP1	RESTORE FOR-VARIABLE ADDRESS
0224	06432	002400	CLA	COMPUTE
0225	06433	002401	CLA, RSS	COMPUTE
0226	06434	060045	EFOR1 LDA M6	NEW TOP OF
0227	06435	042030R	ADA HSTPT	FOR-STACK
0228	06436	072030R	STA HSTPT	POINTER
0229	06437	072047R	STA TEMP1	
0230	06440	003004	CMA, INA	STACK
0231	06441	042027R	ADA LSTPT	
0232	06442	002021	SSA, RSS	OVERFLOW?
0233	06443	027240R	JMP E1	YES
0234	06444	175047R	STB TEMP1, I	NO, RECORD FOR-VARIABLE ADDRESS
0235	06445	017123R	JSB FORMX	INITIALIZE FOR VARIABLE
0236	06446	035045R	ISZ TEMPS	
0237	06447	035047R	ISZ TEMP1	SAVE
0238	06450	062047R	LDA TEMP1	LIMIT
0239	06451	072357R	STA ENEX2	ADDRESS
0240	06452	015255R	JSB FETCH	FETCH
0241	06453	172047R	STA TEMP1, I	AND
0242	06454	035047R	ISZ TEMP1	STORE
0243	06455	175047R	STB TEMP1, I	LIMIT
0244	06456	035047R	ISZ TEMP1	
0245	06457	064051	LDB M2	SET FOR STEP SIZE
0246	06460	075324R	STB FDATA	SIGN CHECK
0247	06461	017062R	JSB LDIMP	LOOK FOR
0248	06462	002002	SZA	FOLLOWING 'STEP'
0249	06463	025467R	JMP EFOR2	FOUND
0250	06464	062261R	LDA HONE	NOT FOUND,
0251	06465	064055	LDB .2	DEFAULT
0252	06466	002001	RSS	IS 1.0
0253	06467	015255R	EFOR2 JSB FETCH	
0254	06470	002020	SSA	STEP SIZE NEGATIVE?
0255	06471	035324R	ISZ FDATA	YES
0256	06472	172047R	STA TEMP1, I	SAVE
0257	06473	035047R	ISZ TEMP1	STEP
0258	06474	175047R	STB TEMP1, I	SIZE
0259	06475	035047R	ISZ TEMP1	SET POINTER
0260	06476	062032R	LDA NXIST	TO STATEMENT
0261	06477	172047R	STA TEMP1, I	FOLLOWING THE FOR
0262	06500	162113R	EFOR3 LDA ANEXT, I	FIND
0263	06501	065031R	LDB PRADD	'NEXT'
0264	06502	015302R	JSB STSRH	STATEMENT
0265	06503	000000	NOP	
0266	06504	015352R	JSB FLWST	FIND FOLLOWING STATEMENT
0267	06505	012176R	AND MSK1	SAME
0268	06506	053440R	CPA EDELM	FOR-VARIABLE?
0269	06507	002001	RSS	YES
0270	06510	025500R	JMP EFOR3	NO
0271	06511	165030R	LDB HSTPT, I	LOAD
0272	06512	104200	OLD 1, I	VALUE OF FOR VARIABLE

```

06515 100001
C273 06514 025542R JMP .ENE2 CHECK ACCEPTABILITY
C274*
C275*** ***
C276** EXECUTE NEXT **
C277*** ***
C278*
C279 06515 015366R ENEXT JSB FVSRH FIND CORRESPONDING STACK ENTRY
C280 06516 025233R JMP XEC4 NONE PRESENT
C281 06517 072030R STA HSTPT RESET TOP OF STACK
C282 06520 075556R STB ENEX1 SAVE FOR-VARIABLE ADDRESS
C283 06521 002004 INA SAVE LIMIT
C284 06522 072557R STA ENEX2 ADDRESS
C285 06523 040055 ADA .2 SAVE STEP SIZE
C286 06524 072047R STA TEMP1 ADDRESS
C287 06525 064051 LDB M2 SET STEP SIZE
C288 06526 075324R STB FDATA SIGN CHECK
C289 06527 162047R LDA TEMP1,I LOAD
C290 06530 035047R ISZ TEMP1 STEP
C291 06531 165047R LDB TEMP1,I SIZE
C292 06532 035047R ISZ TEMP1
C293 06533 002020 SSA CHECK
C294 06534 035324R ISZ FDATA SIGN
C295 06535 015002X FAD ENEX1,I INCREMENT FOR VARIABLE
06536 105556R
C296 06537 172556R STA ENEX1,I AND
C297 06540 035556R ISZ ENEX1 SAVE
C298 06541 175556R STB ENEX1,I VALUE
C299 06542 015003X .ENE2 FSB ENEX2,I COMPUTE FOR VARIABLE LIMIT
06543 105557R
C300 06544 035324R ISZ FDATA POSITIVE STEP SIZE?
C301 06545 001600 ELA YES, COMPLEMENT SIGN
C302 06546 002020 SSA NO, NON-NEGATIVE RESULT?
C303 06547 025552R JMP ENEX3 NO
C304 06550 162047R LDA TEMP1,I YES, GO TO FIRST
C305 06551 025234R JMP XEC4+1 STATEMENT OF LOOP
C306 06552 062030R ENEX3 LDA HSTPT FAILS,
C307 06553 040061 ADA .6 ERASE
C308 06554 072030R STA HSTPT STACK
C309 06555 025233R JMP XEC4 ENTRY
C310 06556 000000 ENEX1 BSS 1
C311 06557 000000 ENEX2 BSS 1
C312*
C313*** ***
C314** EXECUTE GOSUB **
C315*** ***
C316*
C317 06560 000000 EGOSB NOP
C318 06561 017604R JSB NXSGN LOAD (A) WITH NEXT SEQ NUMBER
C319 06562 065032R LDB NXTST LOAD (B) WITH
C320 06563 072032R STA NXTST RETURN SEQUENCE NUMBER
C321 06564 017234R JSB SLWST STACK RETURN ON LOW-CORE STACK
C322 06565 040041 ADA M10 GOSUBS NESTED
C323 06566 052012R CPA LSTAK 10 DEEP?
C324 06567 015274R JSB ERROR YES

```

```
C325 06570 025233R E2      JMP XEC4      NO
C326*
C327***                      ***
C328** EXECUTE RETURN      **
C329***                      ***
C330*
C331 06571 065027R ERTRN LDB LSTPT      RETURN STACK
C332 06572 055012R      CPB LSTAK      EMPTY?
C333 06573 015274R      JSB ERROR      YES
C334 06574 162027R E3      LDA LSTPT,1  NO, LOAD RETURN ADDRESS
C335 06575 044052      ADB M1      RESET
C336 06576 075027R      STB LSTPT      STACK POINTER
C337 06577 025234R      JMP XEC4+1
```

EXECUTE THE PROGRAM

```

C339*
C340***          ***
C341** EXECUTE WAIT **
C342***          ***
C343*
C344 06600 036045R EWAIT ISZ TEMPS      POINT (TEMPS) TO FORMULA
C345 06601 016255R      JSB FETCH      FETCH EVALUATED FORMULA
C346 06602 002020      SSA      NEGATIVE?
C347 06603 026233R      JMP XEC4      YES
C348 06604 017160R      JSB IFIX      CONVERT TO INTEGER
C349 06605 002404      CLA,INA      LARGE INTEGER
C350 06606 002003      SZA,RSS      SMALL
C351 06607 007021      CMB,SSB,RSS      INTEGER?
C352 06610 066263R      LDB MNEG      NO
C353 06611 006007 EWAIT INB,SZB,RSS      WAIT?
C354 06612 026233R      JMP XEC4      NO
C355 06613 062254R      LDA M310      YES, SET INNER LOOP
C356 06614 002006      INA,SZA      MORE?
C357 06615 026614R      JMP *-1      YES
C358 06616 026611R      JMP EWAIT1      NO
C359*
C360***          ***
C361** EXECUTE CALL **
C362***          ***
C363*
C364 06617 036045R ECALL ISZ TEMPS      FETCH
C365 06620 036045R      ISZ TEMPS      SUBROUTINE
C366 06621 017062R      JSB LDTMP      NUMBER
C367 06622 064000      LDB A
C368 06623 017115R      JSB FND5B      FIND
C369 06624 006004      INB      ENTRY
C370 06625 164001      LDB 1,I      POINT AND
C371 06626 075060R ECAL4 STB TEMPS+11
C372 06627 062030R      LDA HSTPT      SAVE HIGH CORE
C373 06630 072057R      STA TEMPS+10      STACK POINTER
C374 06631 036045R ECAL1 ISZ TEMPS      ANY
C375 06632 017062R      JSB LDTMP      PARAMETERS
C376 06633 052202R      CPA B2000 ::
C377 06634 026637R      JMP ECAL2      NO
C378 06635 017723R      JSB FORMX      YES, EVALUATE
C379 06636 026631R      JMP ECAL1      A PARAMETER
C380 06637 003400 ECAL2 CCA      LOAD ADDRESS OF
C381 06640 042057R      ADA TEMPS+10      PARAMETER ADDRESSES
C382 06641 116060R      JSB TEMPS+11,I      AND BRANCH TO SUBROUTINE
C383 06642 052057R      LDA TEMPS+10      RESTORE
C384 06643 072030R      STA HSTPT
C385 06644 066006R      LDB FCORE      POINTERS
C386 06645 076026R      STB TSTPT
C387 06646 026233R      JMP XEC4

```

EXECUTE THE PROGRAM

```

C389*
C390***          ***
C391** EXECUTE READ **
C392***          ***
C393*
C394 06647 056031R EREAD CPB PRADD      END-OF-STATEMENT?
C395 06650 026233R      JMP XEC4        YES
C396 06651 017723R      JSB FORMX       NO, EVALUATE THE NEXT RECORD
C397 06652 162030R      LDA HSTPT,I     RECORD
C398 06653 073461R      STA OUTLN       ADDRESS
C399 06654 002020      SSA
C400 06655 026665R      JMP ERD2
C401 06656 016324R      JSB FDATA       GET DATA ITEM
C402 06657 104400      DST OUTLN,I     STORE DATA ITEM
      06660 101461R
C403 06661 036030R ERD1  ISZ HSTPT
C404 06662 066045R      LDB TEMPS
C405 06663 006004      INB
C406 06664 026647R      JMP EREAD
C407 06665 016324R ERD2  JSB FDATA
C408 06666 017342R      JSB DDST
C409 06667 001461R      DEF OUTLN
C410 06670 026661R      JMP ERD1
C411**
C412*** INITIALIZE FOR PRINT **
C413**
C414 06671 000000 PRNIN NOP
C415 06672 003400      CCA              INITIALIZE
C416 06673 042016R      ADA .BUFA       BUFFER
C417 06674 072017R      STA BADDR       POINTER
C418 06675 062034R      LDA TYPE        INITIALIZE
C419 06676 003004      CMA,INA          'CHARACTERS OUTPUTTED'
C420 06677 072020R      STA CCNT        COUNTER
C421 06700 002011      SLA,RSS          START ON ODD CHARACTER POSITION?
C422 06701 126671R      JMP PRNIN,I     NO
C423 06702 040052      ADA M1           YES, BIAS
C424 06703 072020R      STA CCNT        COUNTER
C425 06704 002400      CLA              OUTPUT A
C426 06705 017477R      JSB OUTCR       NULL CHARACTER
C427 06706 126671R      JMP PRNIN,I
C428*
C429***          ***
C430** EXECUTE PRINT **
C431***          ***
C432*
C433 06707 002400 EPRIN CLA
C434 06710 072272R      STA PARM
C435 06711 016671R      JSB PRNIN
C436 06712 002400      CLA              TURN ON
C437 06713 026720R      JMP EPR11+1     'END-OF-LINE' FLAG
C438 06714 002400 EPR10 CLA              EXECUTE COMMA
C439 06715 052402R      CPA EOL         IF NOT FOLLOWING
C440 06716 017440R      JSB EDELM       A TAB
C441 06717 003400 EPR11 CCA              TURN OFF
C442 06720 072402R      STA EOL         'END-OF-LINE' FLAG

```

EXECUTE THE PROGRAM

0443	06721	017062R	JSB LDTMP	EXTRACT
0444	06722	012225R	AND OPDMK	OPERAND
0445	06723	002002	SZA	NULL?
0446	06724	025744R	JMP EPRI3	NO, FORMULA OR TAB
0447	06725	035045R	ISZ TEMPS	YES
0448	06726	065045R	LDB TEMPS	MORE
0449	06727	055031R	CPB PRADD	STATEMENT?
0450	06730	027010R	JMP EPRI7	NO
0451	06731	017062R	JSB LDTMP	YES, EXTRACT
0452	06732	012220R	AND OPMSK	NEXT OPERATOR
0453	06733	052177R	CPA B1000	::
0454	06734	025714R	JMP EPRI0	YES
0455	06735	052200R	CPA B1400	::
0456	06736	025717R	JMP EPRI1	YES
0457	06737	052174R	CPA B400	
0458	06740	025753R	JMP EPRI4	YES
0459	06741	003400	CCA	NO, MUST BE +, -, OR (
0460	06742	042045R	ADA TEMPS	BACKUP TO PRIOR
0461	06743	072045R	STA TEMPS	NULL OPERAND
0462	06744	003400	EPRI3 CCA	SET
0463	06745	072402R	STA EOL	TAB FLAG
0464	06746	016255R	JSB FETCH	EVALUATE
0465	06747	035402R	ISZ EOL	TAB?
0466	06750	025725R	JMP EPRI2	YES
0467	06751	017426R	JSB ENOUT	NO, PRINT NUMBER
0468	06752	025725R	JMP EPRI2	
0469	06753	002400	EPRI4 CLA	TURN ON
0470	06754	072402R	STA EOL	'END-OF-LINE' FLAG
0471	06755	073234R	STA SLWST	ZERO
0472	06756	073461R	STA OUTLN	CHARACTER COUNT
0473	06757	017051R	EPRI5 JSB LDB1	
0474	06760	010074	AND MSK0	NON-NULL
0475	06761	002003	SZA, RSS	LOW CHARACTER?
0476	06762	025773R	JMP EPRI6	NO
0477	06763	037461R	ISZ OUTLN	YES, COUNT IT
0478	06764	005004	INB	
0479	06765	017051R	JSB LDB1	
0480	06766	012220R	AND OPMSK	
0481	06767	052174R	CPA B400	::
0482	06770	025773R	JMP EPRI6	YES
0483	06771	037461R	ISZ OUTLN	NO, COUNT HIGH CHARACTER
0484	06772	025757R	JMP EPRI5	
0485	06773	037461R	EPRI6 LDB OUTLN	WILL
0486	06774	045020R	ADB CCNT	LINE
0487	06775	052272R	LDA PARM	
0488	06776	002021	SSA, RSS	
0489	06777	027003R	JMP *+4	
0490	07000	045237R	ADB M21	
0491	07001	005021	SSB, RSS	
0492	07002	015274R	WRER JSB ERROR	
0493	07003	045250R	WRERR ADB M73	
0494	07004	005021	SSB, RSS	72 CHARACTERS?
0495	07005	017461R	JSB OUTLN	YES, GET FRESH LINE
0496	07006	017172R	JSB OUTST	OUTPUT STRING
0497	07007	025721R	JMP EPRI1+2	

EXECUTE THE PROGRAM

C498	07010	036402R	EPRI7	ISZ EOL	'END-OF-LINE' ?
C499	07011	027034R		JMP EPRI8	YES
C500	07012	066034R		LDB TYPE	NO, LOAD COUNT OF
C501	07013	007004		CMB, INB	CHARACTERS OUTPUTTED
C502	07014	076272R		STB PARM	
C503	07015	062171R		LDA B137	
C504	07016	017477R		JSB OUTCR	
C505	07017	066272R		LDB PARM	
C506	07020	062020R		LDA CCNT	LOAD LINE LENGTH
C507	07021	003004		CMA, INA	SAVE NEW COUNT OF
C508	07022	072034R		STA TYPE	CHARACTERS OUTPUTTED
C509	07023	040001		ADA 1	COMPUTE CHARACTERS NOT YET OUT
C510	07024	004010		SLB	CORRECT FOR START ON
C511	07025	040052		ADA M1	ODD PRINT POSITION
C512	07026	003004		CMA, INA	
C513	07027	066016R		LDB .BUFA	OUTPUT
C514	07030	002002		SZA	NON-EMPTY
C515	07031	017161R		JSB EXECW	BUFFER
C516	07032	036034R		ISZ TYPE	
C517	07033	026233R		JMP XEC4	
C518	07034	017461R	EPRI8	JSB OUTLN	PRINT LINE
C519	07035	026233R		JMP XEC4	
C520**					
C521***	TAB	TELEPRINTER	**		
C522**					
C523	07036	117062R	ETAB	JSB IENTA, I	SMALL INTEGER?
C524	07037	027060R		JMP ETAB1	NO
C525	07040	006400		CLB	YES, SET
C526	07041	076402R		STB EOL	'TAB' FLAG TRUE
C527	07042	042247R		ADA M72	EXCEED
C528	07043	002021		SSA, RSS	72?
C529	07044	027060R		JMP ETAB1	YES
C530	07045	003004		CMA, INA	NO, COMPUTE
C531	07046	042247R		ADA M72	BLANKS?
C532	07047	042020R		ADA CCNT	REQUIRED
C533	07050	002021		SSA, RSS	ANY?
C534	07051	026101R		JMP FOR12	NO
C535	07052	073461R		STA OUTLN	YES,
C536	07053	062141R		LDA .32	OUTPUT
C537	07054	017477R		JSB OUTCR	REQUIRED
C538	07055	037461R		ISZ OUTLN	NUMBER
C539	07056	027053R		JMP *-3	OF BLANKS
C540	07057	026101R		JMP FOR12	
C541	07060	017461R	ETAB1	JSB OUTLN	OUTPUT THE
C542	07061	026101R		JMP FOR12	LINE
C543*					
C544	07062	011420R	IENTA	DEF .IENT	
C545*					
C546***			***		
C547**	EXECUTE	INPUT	**		
C548***			***		
C549*					
C550	07063	002006	EINP1	INA, SZA	END-OF-INPUT?
C551	07064	017667R		JSB DRQST	YES, CALL FOR MORE
C552	07065	016402R	EINP2	JSB CONST	CONVERT AND STORE NUMBER

EXECUTE THE PROGRAM

C553	07065	027063R	JMP EINP1	NOT NUMBER
C554	07067	066045R	LDB TEMPS	END-OF-
C555	07070	006004	INB	
C556	07071	056031R	CPB PRADD	STATEMENT?
C557	07072	027106R	JMP EINP3	YES
C558	07073	050065	CFA .10	NO, INSURE MORE INPUT
C559	07074	017667R	EINP1 JSB DRGST	CALL FOR INPUT
C560	07075	017723R	JSB FORMX	COMPUTE THE VARIABLE ADDRESS
C561	07076	162030R	LDA HSTPT,I	
C562	07077	002020	SSA	
C563	07100	027103R	JMP *+3	
C564	07101	003400	CCA	STORE
C565	07102	142030R	ADA HSTPT,I	ADDRESS-1
C566	07103	036030R	LSZ HSTPT	IN
C567	07104	072022R	STA SBPTR	POINTER
C568	07105	027065R	JMP EINP2	
C569	07106	002400	EINP3 CLA	RESET
C570	07107	072034R	STA TYPE	OUTPUT BUFFER
C571	07110	026233R	JMP XEC4	

```

C573**
C574*** EXIT FORMULA ON EMPTY STACK **
C575**
C576 07111 105723R DEF FORMX,I
C577*
C578*** ***
C579** EXECUTE RESTORE **
C580*** ***
C581*
C582 07112 066035R ERSTR LDB DSTRT GET FIRST DATA STATEMENT ADDRESS
C583 07113 054052 CPB M1 IMPOSSIBLE ADDRESS?
C584 07114 026233R JMP XEC4 YES, DONE
C585 07115 016271R JSB SETDP NO, SET DATA POINTER
C586 07116 026233R JMP XEC4 DONE
C587**
C588*** FORMULA OPERATOR JUMP TABLE **
C589**
C590 07117 007164R AROT8 DEF ESCMA SUBSCRIPT SEPARATOR
C591 07120 007251R DEF ESTR ASSIGNMENT OPERATOR
C592 07121 007344R DEF EFAD '+'
C593 07122 007350R DEF EFSB '- '
C594 07123 007354R DEF EFMP '* '
C595 07124 007360R DEF EFDV '/ '
C596 07125 007364R DEF EPWR '^ '
C597 07126 007516R DEF EGTRT '> '
C598 07127 007524R DEF ELST '< '
C599 07130 007554R DEF ENEQL '# '
C600 07131 007532R DEF EEQL '='
C601 07132 007567R DEF EUMIN UNARY '- '
C602 07133 007572R DEF ELBRC '['
C603 07134 005726R DEF FORM1 '('
C604 07135 005776R DEF FOR11 UNARY
C605 07136 007153R DEF EOR OR
C606 07137 007160R DEF EAND AND
C607 07140 007600R DEF ENOT NOT
C608 07141 007540R DEF EGORE '>='
C609 07142 007546R DEF ELURE '<='
C610**
C611*** EXECUTE A BINARY OPERATOR **
C612**
C613 07143 000000 BINOP NOP SAVE
C614 07144 017266R JSB OPCHK SAVE ADDRESS OF
C615 07145 077152R STB BIN02 TOP OPERAND
C616 07146 035030R ISZ HSTPT UNSTACK ADDRESS
C617 07147 017252R JSB STTOP LOAD SECOND OPERAND
C618 07150 127143R JMP BINOP,I
C619 07151 000000 BIN01 BSS 1
C620 07152 000000 BIN02 BSS 1
C621**
C622*** EXECUTE OR **
C623**
C624 07153 017143R EOR JSB BINOP VALIDATE
C625 07154 002002 ORS SZA SECOND OPERAND NON-ZERO?
C626 07155 027564R JMP TRUE YES
C627 07156 017517R ORS1 JSB LDUP2 NO, CHECK SECOND

```

0628	07157	027557R	JMP ENEQ1	OPERAND
0629**				
0630***	EXECUTE	AND **		
0631**				
0632	07160	017143R EAND	JSB BINOP	VALIDATE
0633	07161	002003 ANDS	SZA,RSS	SECOND OPERAND ZERO?
0634	07162	027561R	JMP FALSE	YES
0635	07163	027156R	JMP ORS1	NO

EXECUTE THE PROGRAM

C637**

C638*** EXECUTE SUBSCRIPT COMMA **

C639**

C640	07164	017234R	ESCMA	JSB ESBS	INTEGERIZE COLUMN SUBSCRIPT
C641	07165	035027R		ISZ LSTPT	
C642	07166	017234R		JSB ESBS	INTEGERIZE ROW SUBSCRIPT
C643	07167	165030R		LDB HSTPT,I	FETCH
C644	07170	044055		ADB .2	SUBSCRIPT
C645	07171	160001		LDA 1,I	BOUNDS
C646	07172	010074		AND MSK0	SAVE
C647	07173	073461R		STA OUTLN	COLUMN BOUND
C648	07174	160001		LDA 1,I	EXTRACT
C649	07175	001727		ALF,ALF	ROW
C650	07176	010074		AND MSK0	BOUND
C651	07177	003004		CMA,INA	ACTUAL
C652	07200	142027R		ADA LSTPT,I	ROW SUBSCRIPT
C653	07201	002021		SSA,RSS	LEGAL?
C654	07202	027221R		JMP E6-1	NO
C655	07203	063461R		LDA OUTLN	YES
C656	07204	050054		CPA .1	COLUMN MATRIX?
C657	07205	027211R		JMP ESCM1	YES
C658	07206	100200		MPY LSTPT,I	NO, COMPUTE ADDRESS DISPLACEMENT
	07207	100027R			
C659	07210	002001		RSS	DUE TO ROWS
C660	07211	162027R	ESCM1	LDA LSTPT,I	
C661	07212	007400		CCB	UNSTACK
C662	07213	045027R		ADB LSTPT	ROW
C663	07214	076027R		STB LSTPT	SUBSCRIPT
C664	07215	067461R		LDB OUTLN	ACTUAL
C665	07216	007004		CMB,INB	COLUMN
C666	07217	145027R		ADB LSTPT,I	SUBSCRIPT
C667	07220	005021		SSB,RSS	LEGAL?
C668	07221	015274R		JSB ERROR	NO
C669	07222	142027R	E6	ADA LSTPT,I	YES, ADD IN COLUMN DISPLACEMENT
C670	07223	001000		ALS	DOUBLE DISPLACEMENT
C671	07224	165030R		LDB HSTPT,I	COMPUTE
C672	07225	140001		ADA 1,I	ACTUAL
C673	07226	032263R		IGR FLGHT	ADD V.A. BIT
C674	07227	172030R		STA HSTPT,I	ADDRESS
C675	07230	065027R		LDB LSTPT	UNSTACK
C676	07231	044052		ADB M1	
C677	07232	076027R		STB LSTPT	[
C678	07233	027726R		JMP FORM1	

C679**

C680*** INTEGERIZE A SUBSCRIPT **

C681**

C682	07234	000000	ESBS	NOP	
C683	07235	017266R		JSB OPCHK	VALIDATE SUBSCRIPT
C684	07236	005021		SSB,RSS	SUBSCRIPT ON DISK?
C685	07237	027243R		JMP ESB1	NO, FETCH SUBSCRIPT
C686	07240	060001		LDA B	
C687	07241	017505R		JSB FTHOP	YES, FETCH FROM DISK
C688	07242	027245R		JMP *+3	
C689	07243	104200	ESB1	OLD 1,I	
	07244	100001			

EXECUTE THE PROGRAM

C690	07245	017147R	JSB SBFIX	INTEGERIZE
C691	07246	175027R	STB LSTPT,I	SAVE IN OPERATOR STACK
C692	07247	035030R	ISZ HSIPT	POP OPERAND STACK
C693	07250	127234R	JMP ESBS,I	

EXECUTE THE PROGRAM

```

C695**
C696*** EXECUTE STORE **
C697**
C698 07251 065054R ESTR LDB TEMPS+7 IS NEXT OPERATOR
C699 07252 006002 SZB AN END-OF-FORMULA?
C700 07253 025005R JMP FOR10 NO, DEFER AND STORE
C701 07254 055053R CPB TEMPS+6 YES, FIRST STORE OPERATOR USED?
C702 07255 027305R JMP ESTR2 YES
C703 07256 162030R ESTR1 LDA HSIPT,I SET
C704 07257 072056R STA TEMPS+9 .DESTINATION
C705 07260 062053R LDA TEMPS+6 SOURCE ADDRESS IN (A)
C706 07261 002020 SSA SOURCE ADDR. ON DISK?
C707 07262 027311R JMP ESTR4 YES, TRANSFER DATA
C708 07263 164000 LDB 0,1 TRANSFER HIGH
C709 07264 073130R STA TMP2 SAVE SOURCE ADDR.
C710 07265 062056R LDA TEMPS+9 .DESTINATION ADDR ON DISK?
C711 07266 002020 SSA
C712 07267 017334R JSB ESTR5 STORE DATA & RETURN (P+2)
C713 07270 175056R STB TEMPS+9,I PART OF SOURCE
C714 07271 063130R LDA TMP2
C715 07272 035056R ISZ TEMPS+9 UPDATE
C716 07273 002004 INA POINTERS
C717 07274 164000 LDB 0,1 TRANSFER LOW
C718 07275 073130R STA TMP2 SAVE SOURCE ADDR.
C719 07276 062056R LDA TEMPS+9 .DESTINATION ADDR. ON DISK?
C720 07277 002020 SSA
C721 07300 017334R JSB ESTR5 YES, STORE DATA & RETURN (P+2)
C722 07301 175056R STB TEMPS+9,I PART OF SOURCE
C723 07302 063130R LDA TMP2
C724 07303 035030R ISZ HSTPT POP STACK
C725 07304 027776R JMP FOR11
C726 07305 017266R ESTR2 JSB OPCHK SAVE ADDRESS
C727 07306 075053R STB TEMPS+6 OF QUANTITY
C728 07307 035030R ISZ HSTPT YES, POP HIGH-CORE
C729 07310 027256R JMP ESTR1 STACK AND EXECUTE STORE
C730 07311 012222R ESTR4 AND INF REMOVE V.A. BIT
C731 07312 073130R STA TMP2 SAVE SOURCE ADDRESS
C732 07313 017204R JSB DREAD HIGH PART OF SOURCE--> (A)
C733 07314 070001 STA B
C734 07315 062056R LDA TEMPS+9 .DESTINATION ON DISK?
C735 07316 002020 SSA
C736 07317 017334R JSB ESTR5 YES, STORE LOW PART OF SOURCE
C737 07320 175056R STB TEMPS+9,I
C738 07321 035056R ISZ TEMPS+9 UPDATE POINTERS
C739 07322 037130R ISZ TMP2
C740 07323 063130R LDA TMP2 GET SOURCE ADDR.
C741 07324 017204R JSB DREAD LOW PART OF SOURCE --> (A)
C742 07325 070001 STA B
C743 07326 062056R LDA TEMPS+9 .DESTINATION ON DISK?
C744 07327 002020 SSA
C745 07330 017334R JSB ESTR5 YES, STORE HIGH PART OF SOURCE
C746 07331 175056R STB TEMPS+9,I
C747 07332 035030R ISZ HSTPT POP HIGH CORE
C748 07333 027776R JMP FOR11
C749 07334 000000 ESTR5 NOP

```

EXECUTE THE PROGRAM

```

C0750 07335 012222R      AND INF      REMOVE V.A. BIT
C0751 07336 073132R      STA TMP3      AND SAVE
C0001 07337 060001      LDA B
C0002 07340 017151R      JSB DSTOR      (A)--> V.A. ON DISK
C0003 07341 015152R      DEF TMP3
C0004 07342 037334R      ISZ ESTR5
C0005 07343 127334R      JMP ESTR5,1
C006**
C007*** CALL ADD **
C008**
C009 07344 017143R EFAD  JSB BINOP
C010 07345 017403R      JSB .FAD.
C011 07346 007152R      DEF BIN02
C012 07347 027771R      JMP FORM0
C013**
C014*** CALL SUBTRACT **
C015**
C016 07350 017143R EFSB  JSB BINOP
C017 07351 017412R      JSB .FSB.
C018 07352 007152R      DEF BIN02
C019 07353 027771R      JMP FORM0
C020**
C021*** CALL MULTIPLY **
C022**
C023 07354 017143R EFMP  JSB BINOP
C024 07355 017421R      JSB .FMP.
C025 07356 007152R      DEF BIN02
C026 07357 027771R      JMP FORM0
C027**
C028*** CALL DIVIDE **
C029**
C030 07360 017143R EFDV  JSB BINOP
C031 07361 017430R      JSB .FDV.
C032 07362 007152R      DEF BIN02
C033 07363 027771R      JMP FORM0
C034**
C035*** EXECUTE ^ **
C036**
C037 07364 165030R EPWR  LDB HSTPT,I  LOAD
C038 07365 077130R      STB TMP2
C039 07366 017364R      JSB F/PAK      POWER
C040 07367 015130R      DEF TMP2
C041 07370 017160R      JSB IFIX
C042 07371 027374R      JMP *+3
C043 07372 102301      SOS      INTEGER?
C044 07373 027406R      JMP EPWR1    YES
C045 07374 017143R      JSB BINOP    NO
C046 07375 017464R RPWR  JSB PCHK    CHECK ARGUMENTS
C047 07376 002020      SSA      NEGATIVE BASE?
C048 07377 015274R      JSB ERROR    YES
C049 07400      BASER EQU *
C050 07400 067151R      LDB BIN01    NO, LOAD BASE
C051 07401 115067R      JSB .LOGA,I  TAKE NATURAL LOG
C052 07402 017421R      JSB .FMP.    MULTIPLY BY POWER
C053 07403 007152R      DEF BIN02

```

C054	07404	017206R	JSB .EXP	EXPONENTIATE
C055	07405	027771R	JMP FORM0	RESULT
C056	07406	077463R	EPWR1 STB TT1	SAVE SIGN
C057	07407	005020	SSB	SAVE
C058	07410	007004	CMB, INB	ABSOLUTE VALUE
C059	07411	077464R	STB TT2	OF POWER
C060	07412	017143R	JSB BIN0P	
C061	07413	017464R	IPWR JSB PCHK	CHECK ARGUMENTS
C062	07414	067151R	LDB BIN01	STORE
C063	07415	073151R	STA BIN01	
C064	07416	077152R	STB BIN02	BASE
C065	07417	062261R	LDA HONE	INITIALIZE
C066	07420	072051R	STA TT3	RESULT
C067	07421	060055	LDA .2	TO
C068	07422	072052R	STA TT4	1.0
C069	07423	067464R	IPWR1 LDB TT2	DIVIDE POWER
C070	07424	004031	SLB, BRS	BY 2
C071	07425	027444R	JMP IPWR3	WAS ODD
C072	07426	077464R	STB TT2	WAS EVEN
C073	07427	005002	IPWR2 SZB	ZERO?
C074	07430	027455R	JMP IPWR4	NO
C075	07431	063463R	LDA TT1	YES
C076	07432	002020	SSA	POSITIVE POWER?
C077	07433	027437R	JMP IPWR5	NO
C078	07434	062051R	LDA TT3	YES, LOAD
C079	07435	065052R	LDB TT4	RESULT
C080	07436	027771R	JMP FORM0	
C081	07437	062261R	IPWR5 LDA HONE	LOAD
C082	07440	064055	LDB .2	1.0
C083	07441	015004X	JSB .FDV	DIVIDE BY
C084	07442	000051R	DEF TT3	RESULT
C085	07443	027771R	JMP FORM0	
C086	07444	077464R	IPWR3 STB TT2	SAVE POWER
C087	07445	063151R	LDA BIN01	LOAD
C088	07446	067152R	LDB BIN02	BASE
C089	07447	015001X	FMP TT3	MULTIPLY BY RESULT-SO-FAR
	07450	000051R		
C090	07451	072051R	STA TT3	SAVE PARTIAL
C091	07452	075052R	STB TT4	RESULT
C092	07453	067464R	LDB TT2	LOAD POWER
C093	07454	027427R	JMP IPWR2	
C094	07455	063151R	IPWR4 LDA BIN01	LOAD
C095	07456	067152R	LDB BIN02	BASE
C096	07457	015001X	FMP BIN01	SQUARE IT
	07460	007151R		
C097	07461	073151R	STA BIN01	SAVE
C098	07462	077152R	STB BIN02	RESULT
C099	07463	027423R	JMP IPWR1	
C100**				
C101***	INSURE VALID OPERATION **			
C102**				
C103	07464	000000	PCHK NOP	
C104	07465	077151R	STB BIN01	LOAD
C105	07466	073130R	STA TMP2	
C106	07467	063152R	LDA BIN02	

EXECUTE THE PROGRAM

C107	07470	002020	SSA	POWER ON DISK?
C108	07471	027475R	JMP PCK1	YES
C109	07472	164000	LDB A,1	NO, LOAD POWER FROM CORE
C110	07473	063130R	LDA TMP2	
C111	07474	027500R	JMP **4	
C112	07475	012222R	PCK1 AND INF	
C113	07476	017204R	JSB DREAD	LOAD POWER FROM DISK
C114	07477	064000	LDB A	
C115	07500	063130R	LDA TMP2	
C116	07501	002002	SZA	BASE ZERO?
C117	07502	027513R	JMP PCHK1	NO
C118	07503	006003	SZB,RSS	YES, POWER ZERO?
C119	07504	016274R	JSB ERROR	YES
C120	07505		POWER EQU *	
C121	07505	006021	SSB,RSS	NO, POWER POSITIVE?
C122	07506	027561R	JMP FALSE	YES
C123	07507	016274R	JSB ERROR	NO
C124	07510	062222R	ZRING LDA INF	USE POSITIVE
C125	07511	064051	LDB M2	INFINITY
C126	07512	027771R	JMP FORM0	
C127	07513	006003	PCHK1 SZB,RSS	POWER ZERO?
C128	07514	027564R	JMP TRUE	YES, RETURN 1.0
C129	07515	127464R	JMP PCHK,1	NO
C130**				
C131***	EXECUTE > **			
C132**				
C133	07516	017143R	EGTRT JSB BINOP	COMPUTE OPERAND
C134	07517	017412R	JSB .FSB.	
C135	07520	007152R	DEF BIN02	
C136	07521	002020	SSA	NEGATIVE?
C137	07522	027561R	JMP FALSE	YES
C138	07523	027557R	JMP ENEQ1	NO
C139**				
C140***	EXECUTE < **			
C141**				
C142	07524	017143R	ELST JSB BINOP	COMPUTE OPERAND
C143	07525	017412R	JSB .FSB.	
C144	07526	007152R	DEF BIN02	
C145	07527	002020	SSA	NEGATIVE?
C146	07530	027564R	JMP TRUE	YES
C147	07531	027561R	JMP FALSE	NO
C148**				
C149***	EXECUTE = **			
C150**				
C151	07532	017143R	EEQL JSB BINOP	COMPUTE OPERAND
C152	07533	017412R	JSB .FSB.	
C153	07534	007152R	DEF BIN02	
C154	07535	002002	EEQL1 SZA	ZERO?
C155	07536	027561R	JMP FALSE	NO
C156	07537	027564R	JMP TRUE	YES

```

C158**
C159*** EXECUTE >= **
C160**
C161 07540 017143R EGORE JSB BINOP      COMPUTE OPERAND
C162 07541 017412R      JSB .FSB.
C163 07542 007152R      DEF BIN02
C164 07543 002020      SSA              POSITIVE?
C165 07544 027561R      JMP FALSE      NO
C166 07545 027564R      JMP TRUE       YES
C167**
C168*** EXECUTE <= **
C169**
C170 07546 017143R ELORE JSB BINOP      COMPUTE OPERAND
C171 07547 017412R      JSB .FSB.
C172 07550 007152R      DEF BIN02
C173 07551 002020      SSA              NEGATIVE?
C174 07552 027564R      JMP TRUE       YES
C175 07553 027535R      JMP EENL1     NO
C176**
C177*** EXECUTE # **
C178**
C179 07554 017143R ENEQL JSB BINOP      COMPUTE OPERAND
C180 07555 017412R      JSB .FSB.
C181 07556 007152R      DEF BIN02
C182 07557 002002      ENEQ1 SZA       NON-ZERO?
C183 07560 027564R      JMP TRUE       YES
C184**
C185*** SET LOGICAL VALUES **
C186**
C187 07561 002400      FALSE CLA        LOAD
C188 07562 005400      CLB              ZERO
C189 07563 027771R      JMP FORM0
C190 07564 062261R TRUE LDA HONE        LOAD
C191 07565 064055      LDB .2           ONE
C192 07566 027771R      JMP FORM0
C193**
C194*** EXECUTE UNARY - **
C195**
C196 07567 017252R EUMIN JSB STTOP      LOAD NUMBER
C197 07570 017217R      JSB ARINV      NEGATE NUMBER
C198 07571 027771R      JMP FORM0
C199**
C200*** EXECUTE LEFT BRACKET **
C201**
C202 07572 035027R ELBRC ISZ LSTPT      LOAD SUBSCRIPT COMMA
C203 07573 065201R      LDB SCCNT      INFORMATION WORD
C204 07574 017234R      JSB SLWST      STACK IT
C205 07575 017243R      JSB BHSTP      STACK
C206 07576 017321R      JSB RSCHK
C207 07577 027564R      JMP TRUE       1
C208**
C209*** EXECUTE NOT **
C210**
C211 07600 017252R ENOT JSB STTOP      LOAD OPERAND
C212 07601 002002      SZA              ZERO?

```

EXECUTE THE PROGRAM

0213	07602	027561R	JMP FALSE	NO	
0214	07603	027564R	JMP TRUE	YES	
0215*					
0216	07604	000000	NXSN	NOP	EVALUATE FORMULA AND FIND
0217	07605	076045R	STB	TEMPS	NEXT SEQUENCE NUMBER
0218	07606	015255R	JSB	FETCH	
0219	07607	017160R	JSB	IFIX	INTEGERIZE SEQUENCE
0220	07610	000000	NOP		NUMBER
0221	07611	060001	LDA	B	
0222	07612	127604R	JMP	NXSN, I	
0223*					
0224	06251		XSA	END	XEC6+2

```

0226*          *****
0227*          SYMBOL TABLE SEARCH SUBROUTINE
0228*          *****
0229*
0230*          THE SUBROUTINE IS CALLED WITH THE IDENTIFIER TO BE
0231*          SEARCHED FOR IN A . THE SUBROUTINE RETURNS WITH
0232*          THE ADDRESS OF THE MATCHING ENTRY IN B OR -1 IN
0233*          B IF THERE IS NO MATCHING ENTRY
0234*          THE FOLLOWING RULES APPLY WHEN SEARCHING FOR ARRAYS
0235*
0236*          TYPE 1 (1 DIMENSION) SEARCH FOR CORRESPONDING TYPE 1
0237*          OR TYPE 3 ARRAY. IF TYPE 3 IS FOUND CHANGE THE ENTRY
0238*          TYPE TO TYPE 1
0239*
0240*          TYPE 2 (2 DIMENSIONS) SEARCH FOR CORRESPONDING TYPES
0241*          OR TYPE 3 ARRAY. IF TYPE 3 IS FORND CHANGE THE ENTRY
0242*          TYPE TO TYPE 2
0243*
0244*          TYPE 3 (UNDIMENSIONED) SEARCH FOR CORRESPONDING
0245*          TYPE 3 OR TYPE 1 OR TYPE 2 ARRAY
0246*
0247 07613 000000 SSYMT NOP
0248 07614 072051R STA STEMP      STORE IDENTIFIER
0249 07615 010062 AND .7      ISULATE IDENTIFIER TYPE ::
0250 07616 040047 ADA M4
0251 07617 002024 SSA,INA
0252 07620 027624R JMP *+4      JUMP IF ARRAY TYPE
0253 07621 062051R LDA STEMP      RESTORE A
0254 07622 070001 STA 1        STORE IN B
0255 07623 027636R JMP SYMT1+3
0256 07624 002020 SSA          SKIP IF UNDIMENSIONED
0257 07625 027633R JMP SYMT1
0258 07626 062051R LDA STEMP      RESTORE A
0259 07627 010044 AND MSK3     177771B SET TYPE TO 1
0260 07630 070001 STA 1
0261 07631 005004 INB          SET TYPE IN B TO 2
0262 07632 027636R JMP *+4
0263 07633 007400 SYMT1 CCB      SET DIMENSIONED FLAG IN B
0264 07634 060056 LDA .5
0265 07635 032051R IOR STEMP      SET TYPE TO UNDEFINED
0266 07636 072052R STA STEMP+1    STORE A
0267 07637 075053R STB STEMP+2    STORE B
0268 07640 066010R LDB SYMTF      START OF SYMBOL TABLE
0269 07641 027662R JMP SYMT4
0270 07642 160001 SYMT2 LDA 1,1    PICK UP 1ST WORD OF ENTRY
0271 07643 052051R CFA STEMP      COMPARE WITH IDENTIFIER
0272 07644 127613R JMP SSYMT,1    MATCH ? RETURN
0273 07645 052052R CFA STEMP+1    COMPARE WITH DIFFERENT DIM.
0274 07646 027667R JMP SYMT3
0275 07647 052053R CFA STEMP+2    COMPARE WITH DIFFERENT DIM.
0276 07650 027667R JMP SYMT3
0277 07651 160001 LDA 1,1
0278 07652 010062 AND .7      ISULATE ENTRY TYPE ::
0279 07653 050062 CFA .7      FUNCTION
0280 07654 027661R JMP *+5      YES

```

C281	07655	040047	ADA M4	
C282	07656	002020	SSA	ARRAY ?
C283	07657	005004	INB	YES INCREMENT POINTER
C284	07660	005004	INB	INCREMENT POINTER
C285	07661	044055	ADB .2	ADD 2 TO POINTER
C286	07662	055011R	SYM14 CPB SYMTA	SYMBOL TABLE EXHAUSTED?
C287	07663	007401	CCB, RSS	YES
C288	07664	027642R	JMP SYMT2	NO, CHECK NEXT ENTRY FOR MATCH
C289	07665	062051R	LDA STEMP	RETRIEVE SYMBOL
C290	07666	127613R	JMP SSYMT, I	RETURN WITH B NEGATIVE
C291	07667	062051R	SYM13 LDA STEMP	RESTORE A
C292	07670	035053R	ISZ STEMP+2	DIMENSIONED IDENTIFIER?
C293	07671	002001	RSS	NO, SKIP
C294	07672	170001	STA 1, I	YES CHANGE 1ST WORD OF ENTRY TO
C295	07673	127613R	JMP SSYMT, I	APPROPRIATE DIMENSION TYPE

C297**

C298*** ERROR TABLE **

C299**

C300	07674	000071R	ERR	DEF EOF+1	PREMATURE STATEMENT END
C301	07675	001643R		DEF RILE	INPUT EXCEEDS 71 CHARACTERS
C302	07676	001705R		DEF INVSC	SYSTEM COMMAND NOT RECOGNIZED
C303	07677	002202R		DEF SYNE1	NO STATEMENT TYPE FOUND
C304	07700	000532R		DEF NUMER+1	BAD EXPONENT PART
C305	07701	002662R		DEF SYE16	NO LETTER WHERE EXPECTED
C306	07702	002317R		DEF SYNE2	LET STATEMENT HAS NO STORE
C307	07703	002332R		DEF SYNE3	ILLEGAL COM STATEMENT
C308	07704	002357R		DEF SYNE4+1	NO FUNCTION IDENTIFIER (OR BAD)
C309	07705	002371R		DEF SYNE5	MISSING PARAMETER
C310	07706	002376R		DEF SYNE6+1	MISSING ASSIGNMENT OPERATOR
C311	07707	002423R		DEF SYNE7	MISSING 'THEN'
C312	07710	002447R		DEF SYNE8+1	MISSING OR IMPROPER FOR-VARIABLE
C313	07711	002461R		DEF SYNE9	MISSING 'TO'
C314	07712	002474R		DEF SYE10	BAD 'STEP' PART IN FOR STATEMENT
C315	07713	001122R		DEF CALER	CALLED ROUTINE DOES NOT EXIST
C316	07714	002543R		DEF SYE11+1	WRONG NUMBER OF CALL PARAMETERS
C317	07715	000427R		DEF SYE12	NO CONSTANT WHERE EXPECTED
C318	07716	002565R		DEF SYE13	NO VARIABLE WHERE EXPECTED
C319	07717	002617R		DEF SYE14	NO CLOSING QUOTE FOR STRING
C320	07720	002631R		DEF SYE15	PRINT JUXTAPOSES FORMULAS
C321	07721	002675R		DEF SYE17	IMPROPER WORD IN MAT STATEMENT
C322	07722	002706R		DEF SYE18	NO COMMA WHERE EXPECTED
C323	07723	002761R		DEF SYE19	IMPROPER ARRAY FUNCTION
C324	07724	003001R		DEF SYE20	NO SUBSCRIPT WHERE EXPECTED
C325	07725	003013R		DEF SYE21	ARRAY INVERSION INTO SELF
C326	07726	003024R		DEF SYE22	MISSING MULTIPLICATION OPERATOR
C327	07727	003045R		DEF SYE23	IMPROPER ARRAY OPERATOR
C328	07730	003064R		DEF SYE24+1	ARRAY MULTIPLICATION INTO SELF
C329	07731	003206R		DEF FSCE1+1	MISSING LEFT PARENTHESIS
C330	07732	003262R		DEF FSCE2+1	MISSING RIGHT PARENTHESIS
C331	07733	003312R		DEF FSCE3+1	UNRECOGNIZED OPERAND

0352	07734	003540R	DEF ARRE1	MISSING SUBSCRIPT
0353	07735	003553R	DEF ARRE2	MISSING ARRAY IDENTIFIER
0354	07736	004374R	DEF SYE25+1	MISSING OR BAD INTEGER
0355	07737	000072R	DEF NOEOF+1	CHARACTERS AFTER STATEMENT END
0356	07740	003406R	DEF FSCE4+1	OUT OF CORE DURING SYNTAX
0357	07741	001754R	DEF PRERR	PHOTO READER NOT READY
0358	07742	005454R	DEF MER4	FUNCTION MULTIPLY DEFINED
0359	07743	005565R	DEF MER6	UNMATCHED FOR STATEMENT
0360	07744	005423R	DEF MER3	UNMATCHED NEXT
0361	07745	005671R	DEF MER8	OUT OF STORAGE-SYMBOL TABLE
0362	07746	005720R	DEF MSYM	INCONSISTENT DIMENSIONS
0363	07747	005442R	DEF MLOP6	LAST STATEMENT IS NOT 'END'
0364	07750	005551R	DEF MERS	ARRAY DOUBLE DIMENSIONED
0365	07751	005603R	DEF MER10	NO OF DIMENSIONS UNSPECIFIED
0366	07752	001144R	DEF MER9	ARRAY TOO LARGE
0367	07753	005640R	DEF MER7	OUT OF STORAGE-ARRAY ALLOCATION
0368	07754	007222R	DEF E6	SUBSCRIPT TOO LARGE
0369	07755	001303R	DEF E8	UNDEFINED OPERAND ACCESSED
0370	07756	007400R	DEF BASER	NEGATIVE BASE POWERED TO REAL
0371	07757	007505R	DEF POWER	ZERO TO ZERO POWER
0372	07760	005246R	DEF XECS	MISSING STATEMENT
0373	07761	005570R	DEF E2	GOSUBS NESTED 10 DEEP
0374	07762	005574R	DEF E3	RETURN FINDS NO ADDRESS
0375	07763	005333R	DEF E4	OUT OF DATA
0376	07764	001241R	DEF E1+1	OUT OF STORAGE - EXECUTION
0377	07765	012000R	DEF E7	RE-DIMENSIONED ARRAY TOO LARGE
0378	07766	012067R	DEF LERR+1	DIMENSIONS NOT COMPATIBLE
0379	07767	012131R	DEF LCHK5	MATRIX UNASSIGNED
0380	07770	012670R	DEF LDUM1	NEARLY SINGULAR MATRIX
0381	07771	010425R	DEF TRGER	ARGUMENT TOO LARGE
0382	07772	010776R	DEF SQRER	SQRT HAS NEGATIVE ARGUMENT
0383	07773	011104R	DEF LOGER	LOG OF NEGATIVE ARGUMENT
0384	07774		RCERR EQU *	** RECOVERABLE ERRORS FOLLOW **
0385	07774	000727R	DEF OVRER	OVERFLOW
0386	07775	000725R	DEF UNDER	UNDERFLOW
0387	07776	011156R	DEF LNZR	LOG OF ZERO
0388	07777	011312R	DEF EXPR	EXPONENTIAL OVERFLOW
0389	10000	000000	NOP	
0390	10001	007510R	DEF ZRING	ZERO TO NEGATIVE POWER
0391	*****		RECOVERABLE ERROR END *****	
0392	10002	007003R	DEF WRERR	
0393	10003	001546R	DEF NOCOR	OUT OF CORE DURING BUFFER ALLOCATION
0394	10004	015075R	DEF DIA1	INVALID DISK ADDR
0395	10005	002435R	DEF SYE1	INVALID COND. EXECUTABLE STATEMENT
0396	10006	002042R	DEF NOFIL	FILE NOT ON DISK
0397	*			
0398	*			
0399	10007	005412	EBUFF OCT 6412	
0400	10010	042522	ASC 5,ERROR	
	10011	051117		
	10012	051040		
0401	10013	000000	EBFF BSS 2	
0402	10015	020111	LBUFF ASC 5, IN LINE	
	10016	047040		
	10017	045111		

PAGE 0108 #15
EXECUTE THE PROGRAM

4:49 PM SUN., 11 MAR., 1979

	10020	047105	
	10021	020040	
0383	10022	000000	LNBFF BSS 5

C385**

C386*** PREDEFINED FUNCTION JUMP TABLE **

C387**

C388	10025	007036R	PDFT	DEF	ETAB
C389	10026	010654R		DEF	ESIN
C390	10027	010652R		DEF	ECUS
C391	10030	010413R		DEF	ETAN
C392	10031	010537R		DEF	EATN
C393	10032	011204R		DEF	EEXP
C394	10033	011075R		DEF	ELUG
C395	10034	010647R		DEF	EABS
C396	10035	010772R		DEF	ESQR
C397	10036	011062R		DEF	EINT
C398	10037	010743R		DEF	ERND
C399	10040	011173R		DEF	ESGN

C400**

C401*** OUTPUT A NUMBER **

C402**

C403	10041	000000	NUMOT	NCP	NUMBER IN (A) AND (B)
C404	10042	073366R		STA EXPON	SAVE NUMBER
C405	10043	002041		SEZ,RSS	SIGN?
C406	10044	025056R		JMP NS2	NO
C407	10045	002021		SSA,RSS	YES, NEGATIVE NUMBER?
C408	10046	025053R		JMP NS1	NO
C409	10047	017217R		JSB ARINV	YES, INVERT IT
C410	10050	073366R		STA EXPON	
C411	10051	062150R		LDA .45	
C412	10052	002001		RSS	
C413	10053	062141R	NS1	LDA .32	STORE
C414	10054	072041R		STA SIGN	SIGN
C415	10055	063366R		LDA EXPON	
C416	10056	076351R	NS2	STB GETDG	SAVE NUMBER
C417	10057	017160R		JSB IFIX	INTEGERIZE
C418	10060	000000		NCP	
C419	10061	102201		SOC	WAS IT AN INTEGER?
C420	10062	025113R		JMP NUM02	NO

C421**

C422*** OUTPUT AN INTEGER **

C423**

C424	10063	002400		CLA	SAVE
C425	10064	075062R		STB B1+1	INTEGER
C426	10065	045255R		ADB B1000	5 OR MORE
C427	10066	006021		SSB,RSS	CHARACTERS?
C428	10067	040056		ADA .3	YES
C429	10070	040061		ADA .6	COMPUTE
C430	10071	042020R		ADA CCNT	END-OF-FIELD
C431	10072	003004		CMA,INA	SAVE MARKER TO
C432	10073	072060R		STA MLBX1+1	END-OF-FIELD
C433	10074	065272R		LDB PARM	
C434	10075	006021		SSB,RSS	
C435	10076	025102R		JMP *+4	
C436	10077	042132R		ADA .21	
C437	10100	002020		SSA	
C438	10101	027002R		JMP WRERR-1	
C439	10102	042164R		ADA .74	ENOUGH

0440	10103	002020	SSA	ROOM?
0441	10104	017461R	JSB OUTLN	
0442	10105	062041R	LDA SIGN	
0443	10106	002002	SZA	SIGN?
0444	10107	017477R	JSB OUTCR	YES, OUTPUT IT
0445	10110	062062R	LDA 31+1	OUTPUT
0446	10111	017132R	JSB OUTIN	THE INTEGER
0447	10112	126041R	JMP NUMOT,I	
0448	10113	003400	NUMO2 CCA	SET 'FIXED'
0449	10114	073037R	STA FFLAG	FLAG FALSE
0450	10115	063366R	LDA EXPON	LOAD
0451	10116	066351R	LDB GETDG	NUMBER
0452	10117	016002X	FAD MAXFX	IS NUMBER LESS THAN
	10120	000265R		
0453	10121	002021	SSA,RSS	999999.5 ?
0454	10122	026132R	JMP NUMO5	NO
0455	10123	063366R	LDA EXPON	YES, IS
0456	10124	066351R	LDB GETDG	NUMBER
0457	10125	016002X	FAD MINFX	LESS THAN
	10126	009267R		
0458	10127	056127R	LDB .12	0.09999995?
0459	10130	002021	SSA,RSS	
0460	10131	037037R	ISZ FFLAG	NO, SET FFLAG = 0 AND SKIP
0461	10132	064070	NUMO5 LDB .15	YES, LOAD 'FLOATING' FIELD WIDTH
0462	10133	046020R	ADB CCNT	SAVE
0463	10134	007004	CMB,INB	END-OF-FIELD
0464	10135	076060R	STB MLBX1+1	MARKER
0465	10136	046165R	ADB .75	ROOM
0466	10137	006020	SSB	ENOUGH?
0467	10140	017461R	JSB OUTLN	
0468**				
0469*** OUTPUT A FLOATING POINT NUMBER **				
0470**				
0471	10141	063366R	LDA EXPON	
0472	10142	073066R	STA MANT1	
0473	10143	066351R	LDB GETDG	UNPACK
0474	10144	017223R	JSB .FLUN	
0475	10145	077130R	STB MANT2	NUMBER
0476	10146	072042R	STA EXP	
0477	10147	062041R	LDA SIGN	
0478	10150	002002	SZA	SIGN?
0479	10151	017477R	JSB OUTCR	YES, OUTPUT IT
0480	10152	002400	CLA	INITIALIZE COUNTER
0481	10153	073366R	STA EXPON	FOR DECIMAL EXPONENT
0482	10154	052042R	CPA EXP	EXPONENT ZERO?
0483	10155	026200R	JMP EOUT4	YES
0484	10156	016777R	EOUT2 JSB MBY10	NO,
0485	10157	062042R	LDA EXP	MULTIPLY
0486	10160	003004	CMA,INA	NUMBER BY 10
0487	10161	002020	SSA	UNTIL
0488	10162	026165R	JMP *+3	IT IS
0489	10163	037366R	ISZ EXPON	GREATER
0490	10164	026156R	JMP EOUT2	THAN 1
0491	10165	017030R	JSB DBY10	DIVIDE BY 10
0492	10166	063366R	LDA EXPON	

0493	10167	066042R	EOUT3	LDB EXP	DIVIDE
0494	10170	007004		CMB, INB	NUMBER
0495	10171	006021		SSB, RSS	BY 10
0496	10172	026200R		JMP EOUT4	UNTIL
0497	10173	073366R		STA EXPON	IT IS
0498	10174	017030R		JSB DBY10	LESS
0499	10175	003400		CCA	THAN
0500	10176	043366R		ADA EXPON	1
0501	10177	026167R		JMP EOUT3	
0502	10200	003000	EOUT4	CMA	SET EXPONENT
0503	10201	073366R		STA EXPON	TO TRUE VALUE-1
0504	10202	064044		LDB M7	SET DIGIT
0505	10203	076375R		STB RETCR	COUNTER
0506	10204	007400		CCB	SET DECIMAL
0507	10205	077461R		STB OUTLN	POINT FLAG
0508	10206	057037R		CFB FFLAG	FIXED POINT?
0509	10207	026216R		JMP EOUT6	NO
0510	10210	003000		CMA	YES, SET
0511	10211	073461R		STA OUTLN	DECIMAL POINT FLAG
0512	10212	050054		CPA .1	.1?
0513	10213	026222R		JMP EOUT5	YES
0514	10214	002021		SSA, RSS	LEADING DECIMAL POINT?
0515	10215	026230R		JMP EOUT7+2	YES
0516	10216	016351R	EOUT6	JSB GETDG	OUTPUT
0517	10217	042154R		ADA .48	A
0518	10220	017477R		JSB OUTCR	DIGIT
0519	10221	026232R		JMP EOUT8	
0520	10222	062152R	EOUT5	LDA .46	OUTPUT
0521	10223	017477R		JSB OUTCR	DECIMAL POINT
0522	10224	062154R		LDA .48	OUTPUT
0523	10225	026231R		JMP EOUT8-1	LEADING ZERO
0524	10226	037461R	EOUT7	ISZ OUTLN	DECIMAL POINT NEXT?
0525	10227	026216R		JMP EOUT6	NO
0526	10230	062152R		LDA .46	YES,
0527	10231	017477R		JSB OUTCR	OUTPUT IT
0528	10232	036375R	EOUT8	ISZ RETCR	MORE MANTISSA?
0529	10233	026226R		JMP EOUT7	YES
0530	10234	062020R		LDA CCNT	NO,
0531	10235	072347R		STA NUM01	SAVE
0532	10236	062017R		LDA BADDR	OUTPUT
0533	10237	072350R		STA NUM03	POINTERS
0534	10240	016351R		JSB GETDG	NEXT DIGIT
0535	10241	040046		ADA M5	FIVE OR
0536	10242	002020		SSA	GREATER?
0537	10243	026306R		JMP EOUT1	NO
0538	10244	003400		CCA	SET DECIMAL
0539	10245	073066R	ERND1	STA SYMCK	POINT COUNTER
0540	10246	016375R		JSB RETCR	RETRIEVE CHARACTER
0541	10247	052152R		CPA .46	DECIMAL POINT?
0542	10250	026244R		JMP ERND1-1	YES, FLAG IT
0543	10251	017353R		JSB DIGCK	NO, DIGIT?
0544	10252	026265R		JMP ERND2	NO
0545	10253	050064		CPA .9	YES, 9?
0546	10254	026257R		JMP *+3	YES
0547	10255	042155R		ADA .49	NO, BUMP

0548	10256	026301R	JMP ERND3	DIGIT 1
0549	10257	062154R	LDA .48	OVERLAY
0550	10260	017477R	JSB OUTCR	A ZERO
0551	10261	016375R	JSB RETCR	BACKSPACE
0552	10262	003400	CCA	DECREMENT
0553	10263	043066R	ADA SYMCK	DECIMAL POINT
0554	10264	025245R	JMP ERND1	COUNTER
0555	10265	017477R	ERND2 JSB OUTCR	RESTORE CHARACTER
0556	10266	037366R	ISZ EXPON	CORRECT
0557	10267	000000	NOP	EXPONENT
0558	10270	062155R	LDA .49	OVERLAY A 1
0559	10271	067037R	LDB FFLAG	FIXED
0560	10272	006002	SZB	POINT?
0561	10273	026301R	JMP ERND3	NO
0562	10274	017477R	JSB OUTCR	YES, OUTPUT CHARACTER
0563	10275	062154R	LDA .48	PREPARE TO OVERLAY A ZERO
0564	10276	037066R	ISZ SYMCK	DECIMAL POINT NEXT?
0565	10277	026274R	JMP *-3	NO
0566	10300	062152R	LDA .46	YES
0567	10301	017477R	ERND3 JSB OUTCR	RESTORE
0568	10302	062347R	LDA NUM01	OUTPUT
0569	10303	072020R	STA CCNT	POINTERS
0570	10304	062350R	LDA NUM03	
0571	10305	072017R	STA BADDR	
0572	10306	037037R	EOUT1 ISZ FFLAG	NO, FIXED POINT?
0573	10307	025336R	JMP EOUT9	YES
0574	10310	062161R	LDA E	NO,
0575	10311	017477R	JSB OUTCR	OUTPUT 'E'
0576	10312	062150R	LDA .45	LOAD '--'
0577	10313	067366R	LDB EXPON	POSITIVE
0578	10314	005020	SSB	EXPONENT?
0579	10315	007005	CMB, INB, RSS	NO
0580	10316	062147R	LDA .43	YES, LOAD '+'
0581	10317	077366R	STB EXPON	
0582	10320	017477R	JSB OUTCR	OUTPUT SIGN
0583	10321	067366R	LDB EXPON	
0584	10322	062154R	LDA .48	COMPUTE
0585	10323	044041	ADB M10	
0586	10324	005020	SSB	EXPONENT
0587	10325	025330R	JMP *+3	
0588	10326	002004	INA	DIGIT
0589	10327	025323R	JMP *-4	
0590	10330	045157R	ADB .58	COMPUTE
0591	10331	077366R	STB EXPON	SECOND DIGIT
0592	10332	017477R	JSB OUTCR	OUTPUT
0593	10333	063366R	LDA EXPON	
0594	10334	017477R	JSB OUTCR	EXPONENT
0595	10335	125041R	JMP NUM01, I	
0596	10336	016375R	EOUT9 JSB RETCR	RETRIEVE CHARACTER
0597	10337	052154R	CFA .48	ZERO?
0598	10340	025343R	JMP EOU10	YES
0599	10341	017477R	JSB OUTCR	NO, RESTORE CHARACTER
0600	10342	125041R	JMP NUM01, I	
0601	10343	062141R	EOU10 LDA .32	OVERLAY
0602	10344	017477R	JSB OUTCR	A BLANK

EXECUTE THE PROGRAM

0603	10345	015375R	JSB RETCR	BACKSPACE
0604	10346	025336R	JMP EOUT9	
0605	10347	000000	NUM01 BSS 1	
0606	10350	000000	NUM03 BSS 1	

```

C608**
C609*** GET DIGIT TO OUTPUT **
C610**
C611 10351 000000 GETDG NOP
C612 10352 015777R JSB MBY10 MULTIPLY BY 10
C613 10353 065042R LDB EXP GET EXPONENT IN (B)
C614 10354 007004 CMB,INB AS NEGATIVE
C615 10355 012230R AND HIMSK KEEP 5 HIGH BITS OF (A)
C616 10356 001200 RAL NORMALIZE TO BIT 15
C617 10357 005024 SSB,INB ROTATE INTEGER
C618 10360 025356R JMP *-2 INTO (A)
C619 10361 010074 AND MSK0 EXTRACT
C620 10362 072430R STA NUMCK DIGIT
C621 10363 065042R LDB EXP ROTATE
C622 10364 007004 CMB,INB
C623 10365 001300 RAR BACK
C624 10366 005024 SSB,INB
C625 10367 025356R JMP *-2
C626 10370 023066R XOR MANT1 REMOVE
C627 10371 067130R LDB MANT2 DIGIT
C628 10372 015742R JSB NORML NORMALIZE REMAINDER
C629 10373 062430R LDA NUMCK LOAD (A) WITH DIGIT
C630 10374 125351R JMP GETDG,I
C631**
C632*** RETRIEVE CHARACTER FROM OUTPUT BUFFER **
C633**
C634 10375 000000 RETCR NOP
C635 10376 065020R LDB CCNT DECREMENT
C636 10377 044052 ADB M1 CHARACTER
C637 10400 075020R STB CCNT COUNT
C638 10401 162017R LDA BADDR,I POSITION
C639 10402 005011 SLB,RSS AND
C640 10403 001727 ALF,ALF EXTRACT
C641 10404 010074 AND MSK0 CHARACTER
C642 10405 004010 SLB FIRST CHARACTER OF WORD?
C643 10406 125375R JMP RETCR,I NO
C644 10407 065017R LDB BADDR YES, DECREMENT
C645 10410 044052 ADB M1 BUFFER
C646 10411 075017R STB BADDR POINTER
C647 10412 125375R JMP RETCR,I
C648*
C649*

```

```

0651*          *****
0652*          SUBROUTINE TO CALCULATE TAN(X)
0653*          *****
0654*
0655*          CALLED BY A JMP ETAN WITH THE ARGUMENT
0656*          IN FLOATING RADIAN IN THE REGISTERS.
0657*          THE FLOATING RESULT IS RETURNED IN A & B
0658*
0659 10413 016001X ETAN FMP FUPI
          10414 010502R
0660 10415 072506R          STA XTEMP
0661 10416 075507R          STB XTEMP+1
0662 10417 016002X          FAD K1
          10420 010504R
0663 10421 017445R          JSB .PWR2
0664 10422 177776          DEC -2
0665 10423 017420R          JSB .IENT
0666 10424 016274R          JSB ERROR
0667 10425 017437R TRGER JSB FLOAT
0668 10426 017217R          JSB ARINV
0669 10427 017445R          JSB .PWR2
0670 10430 000002          DEC 2
0671 10431 016002X          FAD XTEMP
          10432 010506R
0672 10433 072506R          STA XTEMP
0673 10434 075507R          STB XTEMP+1      X=X-4*ENTIER((X+1)/4)
0674 10435 016003X          FSB K1
          10436 010504R
0675 10437 073037R          STA SBUX          TEMPURARY
0676 10440 002020          SSA          X<1?
0677 10441 026474R          JMP ELSE1          YES
0678 10442 062514R          LDA K2          NO
0679 10443 065515R          LDB K2+1
0680 10444 016003X          FSB XTEMP
          10445 010506R
0681 10446 072510R BOTH1 STA YTEMP
0682 10447 075511R          STB YTEMP+1      Y= 2-X
0683 10450 016001X          FMP YTEMP
          10451 010510R
0684 10452 016001X          FMP K2
          10453 010514R
0685 10454 016003X          FSB K1
          10455 010504R
0686 10456 017531R          JSB .CHEB
0687 10457 010516R          DEF COEFF
0688 10460 016001X          FMP YTEMP
          10461 010510R
0689 10462 072510R          STA YTEMP
0690 10463 075511R          STB YTEMP+1      Y=Y*CHEBY(2*Y**2-1)
0691 10464 063037R          LDA SBUX
0692 10465 002020          SSA          X<1 ?
0693 10466 026477R          JMP ELSE2          YES
0694 10467 062504R          LDA K1
0695 10470 065505R          LDB K1+1
0696 10471 016004X          FDV YTEMP

```

```

10472 010510R
0697 10473 025101R      JMP FOR12
0698 10474 062506R ELSE1 LDA XTEMP
0699 10475 065507R      LDB XTEMP+1
0700 10476 025446R      JMP BOTH1      Y=X
0701 10477 062510R ELSE2 LDA YTEMP
0702 10500 065511R      LDB YTEMP+1
0703 10501 025101R      JMP FOR12
0704*
0705 10502 050574      FUPI DEC 1.273239545      4/PI
10503 141002
0706 10504 040000      K1 DEC 1.
10505 000002
0707 10506 000000      XTEMP BSS 2
0708 10510 000000      YTEMP BSS 2
0709 10512 000000      UTEMP BSS 2
0710 10514 040000      K2 DEC 2.
10515 000004
0711 10516 075061      COEFF DEC 1.4458E-8
10517 075315
0712 10520 065034      DEC 2.013766E-7
10521 175725
0713 10522 057035      DEC 2.804816E-6
10523 030335
0714 10524 050755      DEC 3.906637E-5
10525 114745
0715 10526 043523      DEC 5.4417038E-4
10527 052355
0716 10530 075112      DEC 7.586101578E-3
10531 065763
0717 10532 065520      DEC .10675392857
10533 163773
0718 10534 070512      DEC 1.7701474227
10535 014002
0719 10536 000000      OCT 0

```

```

0721*          *****
0722*          SUBROUTINE TO CALCULATE ATN(X)
0723*          *****
0724*
0725*          CALLED BY A JMP EATN WITH THE ARGUMENT
0726*          IN FLOATING POINT FORM IN THE REGISTERS.
0727*          THE FLOATING RESULT IN THE RANGE -PI/2
0728*          TO PI/2 IS RETURNED IN A & B
0729*
0730 10537 072506R EATN  STA XTEMP
0731 10540 075507R      STB XTEMP+1
0732 10541 060001      LDA 1
0733 10542 010074      AND MSK0
0734 10543 073037R      STA SBOX          TAN = EXP OF (X)
0735 10544 002002      SZA
0736 10545 000010      SLA              ABS (X) > 1 ?
0737 10546 025603R      JMP ELS1          NO
0738 10547 062504R      LDA K1
0739 10550 065505R      LDB K1+1
0740 10551 015004X      FDV XTEMP          U=1/X
      10552 010506R
0741 10553 072512R BTH1 STA UTEMP
0742 10554 075513R      STB UTEMP+1
0743 10555 015001X      FMP UTEMP
      10556 010512R
0744 10557 015001X      FMP K2
      10560 010514R
0745 10561 015003X      FSB K1
      10562 010504R
0746 10563 017331R      JSB .CHEB
0747 10564 010622R      DEF COEF
0748 10565 015001X      FMP UTEMP
      10566 010512R
0749 10567 072510R      STA YTEMP
0750 10570 075511R      STB YTEMP+1      Y=U*CHEBY(2*U**2-1)
0751 10571 063037R      LDA SBOX
0752 10572 002002      SZA
0753 10573 000010      SLA              ABS(X)>1 ?
0754 10574 025606R      JMP ELS2          NO
0755 10575 062506R      LDA XTEMP
0756 10576 002020      SSA              X<U ?
0757 10577 025611R      JMP ELS3          YES
0758 10600 062616R      LDA PIBY2
0001 10601 065617R      LDB PIBY2+1
0002 10602 025613R      JMP ELS3+2      ANS = PI/2 - Y
0003 10603 062506R ELS1 LDA XTEMP
0004 10604 065507R      LDB XTEMP+1
0005 10605 025553R      JMP BTH1          U=X
0006 10606 062510R ELS2 LDA YTEMP
0007 10607 065511R      LDB YTEMP+1
0008 10610 025101R      JMP FOR12      ANSWER = Y
0009 10611 062620R ELS3 LDA MP2
0010 10612 065621R      LDB MP2+1
0011 10613 015003X      FSB YTEMP
      10614 010510R

```

```

0012 10615 025101R      JMP FOR12
0013*
0014 10616 062207  PIBY2 DEC 1.5707963268  PI/2
      10617 166402
0015 10620 115570  MP2   DEC -1.5707963268  -PI/2
      10621 011402
0016 10622 105671  COEF   DEC -1.33034E-8
      10623 102315
0017 10624 055335          DEC 8.64888E-8
      10625 156323
0018 10626 131601          DEC -56.99186E-8
      10627 137731
0019 10630 040033          DEC 3.821037E-6
      10631 035737
0020 10632 111013          DEC -2.6215196E-5
      10633 123343
0021 10634 060542          DEC 1.8574297E-4
      10635 060351
0022 10636 122573          DEC -1.381195004E-3
      10637 062757
0023 10640 055471          DEC .01113584206
      10641 107365
0024 10642 111620          DEC -.1058929245
      10643 147373
0025 10644 070320          DEC 1.762747174
      10645 155002
0026 10646 000000          OCT 0

```

```

0028*
0029*
0030*
0031*
0032*
0033*
0034*
0035*
0036*
0037 10647 002020  EABS  SSA      NEGATIVE?
0038 10650 017217R      JSB ARINV  YES, NEGATE IT
0039 10651 025101R      JMP FOR12

```

```

0041*          *****
0042*          SUBROUTINE TO CALCULATE SIN(X)
0043*          *****
0044*
0045*          CALLED BY A JMP ESIN WITH THE ARGUMENT
0046*          IN FLOATING RADIAN IN THE REGISTERS.
0047*          THE FLOATING RESULT IS RETURNED IN A & B
0048*
0049 10652 016002X ECUS   FAD PIBY2
      10653 010616R
0050 10654 016001X ESIN   FMP TOPI
      10655 010724R
0051 10656 072506R      STA XTEMP
0052 10657 075507R      STB XTEMP+1      X=2*X/PI
0053 10660 016002X      FAD K1
      10661 010504R
0054 10662 017445R      JSB .PWR2
0055 10663 177775      DEC -2
0056 10664 017420R      JSB .IENT
0057 10665 025424R      JMP TRGR-1      ERROR IF EXPONENT >= 15
0058 10666 017437R      JSB FLOAT
0059 10667 016001X      FMP MM4
      10670 010726R
0060 10671 016002X      FAD XTEMP
      10672 010506R
0061 10673 072506R      STA XTEMP
0062 10674 075507R      STB XTEMP+1      X=X-4*ENTIER((X+1)/4)
0063 10675 016003X      FSB K1
      10676 010504R
0064 10677 002020      SSA          X<1 ?
0065 10700 025707R      JMP PAST      YES
0066 10701 062514R      LDA K2
0067 10702 065515R      LDB K2+1
0068 10703 016003X      FSB XTEMP
      10704 010506R
0069 10705 072506R      STA XTEMP
0070 10706 075507R      STB XTEMP+1      X=2-X
0071 10707 062506R PAST  LDA XTEMP
0072 10710 065507R      LDB XTEMP+1
0073 10711 016001X      FMP XTEMP
      10712 010506R
0074 10713 017445R      JSB .PWR2
0075 10714 000001      DEC 1
0076 10715 016003X      FSB K1
      10716 010504R
0077 10717 017331R      JSB .CHEB
0078 10720 010730R      DEF COEF1
0079 10721 016001X      FMP XTEMP
      10722 010506R
0080 10723 025101R      JMP FOR12      ANS = X+CHEBY (2*X**2-1)
0081*
0082 10724 050574 TOPI  DEC .636619772      2/PI
      10725 141000
0083 10726 100000 MM4   DEC -4.
      10727 000004

```

```

C084 10730 047605 COEF1 DEC 1.18496E-6
      10731 072733
C085 10732 134143 DEC -1.365875E-4
      10733 104751
C086 10734 045261 DEC 9.118016E-3
      10735 157365
C087 10736 133371 DEC -.2852615692
      10737 014777
C088 10740 050656 DEC 2.5525579248
      10741 107004
C089 10742 000000 OCT 0

```

```

C091* *****
C092* SUBROUTINE TO COMPUTE RND(X)
C093* *****
C094*
C095* THE ARGUMENT OF RND IS A DUMMY ONE
C096* THE ROUTINE GENERATES A RANDOM NUMBER
C097* IN THE A & B REGISTERS
C098*
C099* R=X/M, X=C*X MOD M, M=2^31, C=2^15 + 3
C100*
C101 10743 002400 ERND CLA X IS INITIALLY 1
C102 10744 072042R STA EXP INITIALIZE EXPONENT
C103 10745 062043R LDA XH COMPUTE
C104 10746 001000 ALS HIGH
C105 10747 042043R ADA XH PART
C106 10750 065044R LDB XL 2*XH
C107 10751 004065 CLE,ERB + XH +
C108 10752 040001 ADA 1 XL*2^15
C109 10753 065044R LDB XL
C110 10754 005275 RBL,CLE,SLB,ERB ADD XL[15] TO
C111 10755 002004 INA (A) (FROM 2*XL)
C112 10756 004066 CLE,ELB 2*XL
C113 10757 045044R ADB XL + XL
C114 10760 001675 ELA,CLE,SLA,ERA ADD OVERFLOW
C115 10761 002104 CLE,INA TO (A)
C116 10762 045263R ADB FLGBT ADD IN TRAILING BIT OF XL*2^15
C117 10763 002040 SEZ ADD OVERFLOW
C118 10764 002004 INA TO (A)
C119 10765 001665 ELA,CLE,ERA ERASE A[15]
C120 10766 072043R STA XH STORE
C121 10767 075044R STB XL INTEGER
C122 10770 015647R JSB .PACK NORMALIZE AND PACK
C123 10771 025101R JMP FOR12

```

```

C125*          *****
C126*          SUBROUTINE TO CALCULATE SQR(X)
C127*          *****
C128*
C129*          CALLED BY A JMP ESQR WITH THE ARGUMENT
C130*          IN FLOATING POINT FORM IN THE REGISTERS.
C131*          THE FLOATING RESULT IS RETURNED IN A & B
C132*
C133 10772 002003 ESQR  SZA,RSS          X=0 ?
C134 10773 025101R      JMP FOR12        YES, ANS = 0
C135 10774 002020      SSA                X<0 ?
C136 10775 015274R      JSB ERROR        YES ERROR
C137 10776 072506R SNRER STA XTEMP
C138 10777 017223R      JSB .FLUN
C139 11000 000031      SLA,ARS          EXP(X) ODD?
C140 11001 027041R      JMP ODD
C141 11002 040052      ADA R1
C142 11003 073037R      STA SBOX          SBOX=EXP(X)/2-1
C143 11004 075507R      STB XTEMP+1      LOW MANTISSA/2
C144 11005 062506R      LDA XTEMP
C145 11006 015001X      FMP SA2
C146 11007 011054R
C146 11010 015002X      FAD SB2
C146 11011 011060R
C147 11012 072510R BTH2 STA YTEMP
C148 11013 075511R      STB YTEMP+1
C149 11014 062506R      LDA XTEMP
C150 11015 065507R      LDB XTEMP+1
C151 11016 015004X      FDV YTEMP
C151 11017 010510R
C152 11020 015002X      FAD YTEMP
C152 11021 010510R
C153 11022 017445R      JSB .PWR2
C154 11023 177777      DEC -1
C155 11024 072510R      STA YTEMP
C156 11025 075511R      STB YTEMP+1      Y=(Y+X/Y)/2
C157 11026 062506R      LDA XTEMP
C158 11027 065507R      LDB XTEMP+1
C159 11030 016004X      FDV YTEMP
C159 11031 010510R
C160 11032 015002X      FAD YTEMP
C160 11033 010510R
C161 11034 072510R      STA YTEMP
C162 11035 075511R      STB YTEMP+1
C163 11036 017445R      JSB .PWR2
C164 11037 000000 SBOX OCT 0
C165 11040 025101R      JMP FOR12        ANS = (P+F/P)*2**SBOX
C166 11041 073037R ODD  STA SBOX          SBOX = EXP(X)/2
C167 11042 044074      ADB MSK0
C168 11043 075507R      STB XTEMP+1      LOW MANTISSA/2
C169 11044 062506R      LDA XTEMP
C170 11045 015001X      FMP SA1
C170 11046 011052R
C171 11047 015002X      FAD SB1
C171 11050 011056R

```

```

C172 11051 027012R      JMP BTH2      Y=SB1+SA1*X
C173*
C174 11052 070000      SA1      DEC .875
      11053 000000
C175 11054 045000      SA2      DEC .578125
      11055 000000
C176 11056 043524      SB1      DEC .27863
      11057 046377
C177 11060 066000      SB2      DEC .421875
      11061 000377

```

```

C179*      *****
C180*      SUBROUTINE TO CALCULATE INT(X)
C181*      *****
C182*
C183*      CALLED BY A JMP EINT WITH THE ARGUMENT
C184*      IN FLOATING POINT FORM IN THE REGISTERS.
C185*      ENTIER(X) IS FLOATED AND RETURNED IN A & B
C186*
C187*      CALCULATES THE FLOATING POINT EQUIVALENT
C188*      OF ENTIER(X)
C189*
C190 11062 077037R EINT  STB SBOX      SAVE LOW WORD
C191 11063 064071      LDB .31      PRESET BIAS FOR
C192 11064 075042R      STB EXP      BINARY POINT
C193 11065 067037R      LDB SBOX      24-BIT
C194 11066 017160R      JSB IFIX      INTEGER?
C195 11067 027072R      JMP EINT1     NO
C196 11070 015647R      JSB .PACK     YES, PACK IT
C197 11071 025101R      JMP FOR12
C198 11072 063377R EINT1 LDA GETCR      RETRIEVE
C199 11073 067037R      LDB SBOX      NUMBER
C200 11074 025101R      JMP FOR12

```

LIBRARY ROUTINES

```

0202*          *****
0203*          SUBROUTINE TO CALCULATE LOG(X)
0204*          *****
0205*
0206*          CALLED BY A JMP ELUG WITH THE ARGUMENT
0207*          IN FLOATING POINT FORM IN THE REGISTERS.
0208*          THE FLOATING RESULT IS RETURNED IN A & B
0209*
0210 11075 017077R ELUG JSB .LOG
0211 11076 025101R JMP FOR12
0212 11077 000000 .LOG NOP
0213 11100 002003 SZA,RSS NON-ZERO ARGUMENT?
0214 11101 027155R JMP .LOG1 NO
0215 11102 002020 SSA YES, POSITIVE ARGUMENT?
0216 11103 016274R JSB ERROR NO
0217 11104 072506R LUGER STA XTEMP YES
0218 11105 017223R JSB .FLUN
0219 11106 076507R STB XTEMP+1
0220 11107 017437R JSB FLOAT
0221 11110 072510R STA YTEMP
0222 11111 076511R STB YTEMP+1 Y=EXPO(X)
0223 11112 062506R LDA XTEMP
0224 11113 066507R LDB XTEMP+1
0225 11114 016002X FAD R22
      11115 011161R
0226 11116 072512R STA UTEMP
0227 11117 076513R STB UTEMP+1 U=X+SQR(0.5)
0228 11120 062506R LDA XTEMP
0229 11121 066507R LDB XTEMP+1
0230 11122 016003X FSB R22
      11123 011161R
0231 11124 016004X FDV UTEMP
      11125 010512R
0232 11126 072512R STA UTEMP
0233 11127 076513R STB UTEMP+1 U=(X-SQR(0.5))/U
0234 11130 016001X FMP UTEMP
      11131 010512R
0235 11132 016003X FSB CCC
      11133 011171R
0236 11134 072506R STA XTEMP
0237 11135 076507R STB XTEMP+1
0238 11136 063167R LDA MB
0239 11137 067170R LDB MB+1
0240 11140 016004X FDV XTEMP
      11141 010506R
0241 11142 016002X FAD AAA
      11143 011165R
0242 11144 016001X FMP UTEMP
      11145 010512R
0243 11146 016003X FSB HALF
      11147 000261R
0244 11150 016002X FAD YTEMP
      11151 010510R
0245 11152 016001X FMP LE2
      11153 011163R

```

LIBRARY ROUTINES

```

0246 11154 127077R      JMP .LOG,I      ANS=LOG(2)*(EXPO(X)-0.5+U*
0247*                               (A-B/X))
0248 11155 015274R .LOG1 JSB ERROR      LOG OF ZERO
0249 11156 062263R LNZR LDA MNEG      RETURN
0250 11157 065175R      LDB B776      NEGATIVE
0251 11160 127077R      JMP .LOG,I      INFINITY
0252*
0253 11161 055202 R22 DEC .707106781    SQR(0.5)
      11162 075000
0254 11163 054271 LE2 DEC .6931471806    LOG BASE E OF 2
      11164 095000
0255 11165 051260 AAA DEC 1.2920070987
      11166 037402
0256 11167 125606 MB DEC -2.6398577035
      11170 044404
0257 11171 065010 CCC DEC 1.6567626301
      11172 063002

0259*
0260* *****
0261* SUBROUTINE TO COMPUTE SGN(X)
0262* *****
0263*
0264* CALLED BY A JMP ESGN WITH THE ARGUMENT
0265* IN FLOATING POINT FORM IN THE REGISTERS.
0266* ON RETURN A & B CONTAIN THE FOLLOWING:
0267*
0268* IF X>0 THEN +1.
0269* IF X=0 THEN 0
0270* IF X<0 THEN -1.
0271 11173 006400 ESGN CLB
0272 11174 002003      SZA,RSS      ZERO?
0273 11175 025101R      JMP FOR12      YES
0274 11176 002021      SSA,RSS      NO, POSITIVE?
0275 11177 064055      LDB .2      YES, SET EXPONENT
0276 11200 062263R      LDA FLGBT      LOAD MANTISSA
0277 11201 005002      SZB      POSITIVE?
0278 11202 001300      RAR      YES, CORRECT MANTISSA
0279 11203 025101R      JMP FOR12

```

```

0281*          *****
0282*          SUBROUTINE TO CALCULATE EXP(X)
0283*          *****
0284*
0285*          CALLED BY A JMP EEXP WITH THE ARGUMENT
0286*          IN FLOATING POINT FORM IN THE REGISTERS.
0287*          THE FLOATING RESULT IS RETURNED IN A & B
0288*
0289 11204 017206R EEXP JSB .EXP
0290 11205 025101R JMP FOR12
0291 11206 000000 .EXP NOP
0292 11207 015001X FMP L2E
      11210 011327R
0293 11211 072506R STA XTEMP
0294 11212 075507R STB XTEMP+1 X=Z*LOG2(E)
0295 11213 017420R JSB .IENT
0296 11214 027306R JMP .EXP1
0297 11215 073301R STA INTE INTE = ENTIER(X)
0298 11216 017437R JSB FLOAT
0299 11217 072510R STA YTEMP
0300 11220 075511R STB YTEMP+1 Y=ENTIER(X)
0301 11221 063301R LDA INTE
0302 11222 043315R ADA M124
0303 11223 002021 SSA,RSS INTE >=124 ?
0304 11224 027311R JMP EXPR-1 YES,ERROR
0305 11225 043316R ADA .244 INTE <-120 ?
0306 11226 002020 SSA
0307 11227 027303R JMP ZERE YES,ANS=0
0308 11230 062506R LDA XTEMP
0309 11231 065507R LDB XTEMP+1
0310 11232 015003X FSB YTEMP
      11233 010510R
0311 11234 072506R STA XTEMP
0312 11235 075507R STB XTEMP+1 X=X-ENTIER(X)
0313 11236 015001X FMP XTEMP
      11237 010506R
0314 11240 072512R STA UTEMP
0315 11241 075513R STB UTEMP+1 U=X**2
0316 11242 015002X FAD AAAA
      11243 011317R
0317 11244 072510R STA YTEMP
0318 11245 075511R STB YTEMP+1 Y=X**2+AAAA
0319 11246 063321R LDA BBBB
0320 11247 067322R LDB BBBB+1
0321 11250 015004X FDV YTEMP
      11251 010510R
0322 11252 072510R STA YTEMP
0323 11253 075511R STB YTEMP+1 Y=BBBBB/Y
0324 11254 063323R LDA CCCC
0325 11255 067324R LDB CCCC+1
0326 11256 015001X FMP UTEMP
      11257 010512R
0327 11260 015002X FAD DDDD
      11261 011325R
0328 11262 015003X FSB XTEMP

```

	11263	010506R			
C329	11264	015003X	FSB	YTEMP	
	11265	010510R			
C330	11266	072510R	STA	YTEMP	
C331	11267	075511R	STB	YTEMP+1	$Y = -X + DDDD + CCCC * X ** 2 - Y$
C332	11270	062506R	LDA	XTEMP	
C333	11271	066507R	LDB	XTEMP+1	
C334	11272	015004X	FDV	YTEMP	
	11273	010510R			
C335	11274	015002X	FAD	HALF	
	11275	000261R			
C336	11276	037301R	ISZ	INTE	
C337	11277	000000	NOP		
C338	11300	017445R	JSB	.PWR2	
C339	11301	000000	INTE	UCT 0	
C340	11302	127206R	JMP	.EXP, I	$ANS = (0.5 + X/Y) * 2 ** INTE$
C341	11303	002400	ZERE	CLA	
C342	11304	005400		CLB	
C343	11305	127206R	JMP	.EXP, I	$ANS = 0$
C344	11306	063406R	.EXP1	LDA X2TMP	
C345	11307	002020		SSA	
C346	11310	027303R	JMP	ZERE	
C347	11311	015274R	JSB	ERROR	
C348	11312	062222R	EXPER	LDA INF	
C349	11313	064051		LDB M2	
C350	11314	127206R	JMP	.EXP, I	
C351*					
C352	11315	177604	M124	DEC -124	
C353	11316	000364	.244	DEC 244	
C354	11317	053552	AAAA	DEC 87.417497202	
	11320	160416			
C355	11321	045477	BBBB	DEC 617.9722695	
	11322	015424			
C356	11323	043372	CCCC	DEC .03465735903	
	11324	070371			
C357	11325	047643	DDDD	DEC 9.9545957821	
	11326	001410			
C358	11327	056125	L2E	DEC 1.4426950409	
	11330	017002			

```

C360*          *****
C361*          SUBROUTINE TO COMPUTE CHEBY(X)
C362*          *****
C363*
C364*          CALLING SEQUENCE:
C365*
C366*          LDA X          (FLOATING)
C367*          LDB X+1
C368*          JSB .CHEB      (RESULT FLOATING)
C369*          DEF C          (TABLE OF CHEBY.COEFFS.,FLOATING)
C370*
C371 11331 000000 .CHEB NOP
C372 11332 016001X FMP K2
      11333 010514R
C373 11334 073406R STA X2TMP
C374 11335 077407R STB X2TMP+1 X2 =X*2
C375 11336 167531R LDB .CHEB,I
C376 11337 077414R STB CTMP C POINTS TO COEFFICIENT TABLE
C377 11340 160001 LDA 1,I
C378 11341 006004 INB
C379 11342 164001 LDB 1,I GET FIRST COEFF
C380 11343 073416R STA DTMP
C381 11344 077417R STB DTMP+1 D=C(N)
C382 11345 002400 CLA
C383 11346 073412R STA BTMP
C384 11347 073413R STA BTMP+1 B=0
C385 11350 037414R LOPC ISZ CTMP
C386 11351 037414R ISZ CTMP N=N-1
C387 11352 163414R LDA CTMP,I
C388 11353 002003 SZR,RSS C(N)=0 ?
C389 11354 027376R JMP COOT ZERO FLAGS END OF TABLE
C390 11355 063412R LDA BTMP NO
C391 11356 067413R LDB BTMP+1
C392 11357 073410R STA ATMP
C393 11360 077411R STB ATMP+1 A=B
C394 11361 063416R LDA DTMP
C395 11362 067417R LDB DTMP+1
C396 11363 073412R STA BTMP
C397 11364 077413R STB BTMP+1 B=D
C398 11365 016001X FMP X2TMP
      11366 011406R
C399 11367 016003X FSB ATMP
      11370 011410R
C400 11371 016002X FAD CTMP,I
      11372 111414R
C401 11373 073416R STA DTMP
C402 11374 077417R STB DTMP+1 D=C(N) -A+B*X2
C403 11375 027350R JMP LOPC
C404 11376 063416R COOT LDA DTMP
C405 11377 067417R LDB DTMP+1
C406 11400 016003X FSB ATMP
      11401 011410R
C407 11402 016001X FMP HALF
      11403 000261R
C408 11404 037531R ISZ .CHEB

```

```

C409 11405 127331R      JMP .CHEB,I      ANS=(D-A)/2
C410*
C411 11406 000000      X2TMP BSS 2
C412 11410 000000      ATMP  BSS 2
C413 11412 000000      BTMP  BSS 2
C414 11414 000000      CTMP  BSS 2
C415 11416 000000      DTMP  BSS 2

```

```

C417*      *****
C418*      SUBROUTINE TO COMPUTE THE ENTIER OF A NUMBER
C419*      WHOSE EXPONENT IS LESS THAN 15
C420*      *****

```

```

C421*
C422*      CALLING SEQUENCE:

```

```

C423*
C424*      LDA X          (FLOATING)
C425*      LDA X+1
C426*      JSB .IENT      (RESULT INTERGER)
C427*      JSB ERROR      (EXIT IF EXPO(X)>14)
C428*
C429*

```

```

C430 11420 000000      .IENT NOP
C431 11421 073406R      STA X2TMP      STORE HIGH PART
C432 11422 060001      LDA 1          MOVE LOW PART TO A
C433 11423 010074      AND MSK0      ISOLATE EXPONENT
C434 11424 000033      SLA,RAR
C435 11425 027431R      JMP *+4      IF NEGATIVE OK
C436 11426 042235R      ADA M15
C437 11427 002021      SSA,RSS      EXPO(X) > 14
C438 11430 127420R      JMP .IENT,I   YES, ERROR RETURN
C439 11431 037420R      ISZ .IENT     NO BUMP RETURN POINT
C440 11432 063406R      LDA X2TMP     RESTORE HIGH PART
C441 11433 017160R      JSB IFIX      CALL ENTIER
C442 11434 000000      NOP
C443 11435 060001      LDA 1          PUT RESULT INTO (A)
C444 11436 127420R      JMP .IENT,I

```

LIBRARY ROUTINES

```

C446*          *****
C447*          SUBROUTINE TO FLOAT AN INTEGER
C448*          *****
C449*
C450*          CALLED BY JSB FLOAT WITH INTEGER IN A
C451*          THE FLOATING POINT EQUIVALENT IS RETURNED
C452*          IN A & B
C453*
C454 11437 000000  FLOAT NOP
C455 11440 064070      LDB .15
C456 11441 075042R     STB EXP
C457 11442 005400      CLB
C458 11443 015647R     JSB .PACK
C459 11444 127437R     JMP FLOAT,I

C461*          *****
C462*          SUBROUTINE TO MULTIPLY BY A POWER OF TWO
C463*          *****
C464*
C465*          CALLING SEQUENCE
C466*
C467*          LDA X          (FLOATING)
C468*          LDB X+1
C469*          JSB .PWR2      (RESULT FLOATING)
C470*          DEC N          (INTEGER POWER)
C471*
C472*          RETURNS WITH  $X*2^N$  IN A&B
C473*          NO CHECK IS MADE FOR EXPONENT
C474*          OVERFLOW OR UNDERFLOW
C475*
C476 11445 000000  .PWR2 NOP
C477 11446 002003      SZR,RSS          X=0 ?
C478 11447 027461R     JMP .RET          YES, ANS=0
C479 11450 073406R     STA X2TMP
C480 11451 017223R     JSB .FLUN
C481 11452 077407R     STB X2TMP+1
C482 11453 143445R     ADA .PWR2,I
C483 11454 001200      RAL
C484 11455 010074      AND MSK0          NEW EXPO = (OLD EXPO) +N
C485 11456 079001      STA 1
C486 11457 047407R     ADB X2TMP+1      KEEP OLD MANTISSA
C487 11460 063406R     LDA X2TMP
C488 11461 037445R     .RET  ISZ .PWR2
C489 11462 127445R     JMP .PWR2,I

C491 11463 000000  TT1  BSS 1
C492 11464 000000  TT2  BSS 1
C493 00051          TT3  EQU TEMPS+4
C494 00052          TT4  EQU TEMPS+5
C495 11037          FFLAG EQU SBOX

```

```

C497*
C498*
C499*
C500 11465 017062R EMAT JSB LDIMP
C501 11466 035045R ISZ TEMPS MAT READ
C502 11467 012173R AND MSK1A :::
C503 11470 002002 SZ A MAT PRINT?
C504 11471 027620R JMP EMAT7 NO
C505 11472 017062R JSB LDIMP YES
C506 11473 012220R AND OPMSK SAVE
C507 11474 072057R STA MLBX1 TYPE
C508 11475 052213R CPA RDOP PRINT?
C509 11476 002001 RSS NO
C510 11477 015671R JSB PRNIN YES
C511*
C512 11500 017062R EMAT1 JSB LDIMP LOAD
C513 11501 012176R AND MSK1 OPERAND
C514 11502 002003 SZ A,RSS NULL? (END OF MAT PRINT)
C515 11503 025233R JMP XEC4 YES
C516 11504 017613R JSB SSYMT NO, SEARCH THE SYMBOL TABLE
C517 11505 005007 INB,SZB,RSS FOUND?
C518 11506 027302R JMP E8-1 NO
C519 11507 035045R ISZ TEMPS YES
C520 11510 160001 LDA 1,I SAVE ARRAY
C521 11511 012222R AND INF
C522 11512 072061R STA B1 BASE ADDRESS
C523 11513 062057R LDA MLBX1
C524 11514 052213R CPA RDOP READ?
C525 11515 027571R JMP EMAT5 YES
C526 11516 044055 ADB .2 NO
C527 11517 160001 LDA 1,I SAVE
C528 11520 072062R STA B1+1 DIMENSIONS
C529 11521 010074 AND MSK0 SET
C530 11522 003004 CMA,INA COLUMN
C531 11523 072063R STA B2 COUNTERS
C532 11524 072064R STA B2+1
C533 11525 160001 LDA 1,I SET
C534 11526 001727 ALF,ALF
C535 11527 010074 AND MSK0 ROW
C536 11530 003004 CMA,INA
C537 11531 072065R STA B3 COUNTER
C538 11532 015076R JSB LCHK2 SET DELIMITER
C539 11533 002400 CLA SET DELIMITER
C540 11534 072001R STA MCKS AS COMMA
C541 11535 062045R LDA TEMPS MORE
C542 11536 052031R CPA PRADD STATEMENT?
C543 11537 027550R JMP EMAT3 NO
C544 11540 017062R JSB LDIMP YES
C545 11541 012220R AND OPMSK EXTRACT DELIMITER
C546 11542 052200R CPA B1400 ::
C547 11543 035001R ISZ MCKS YES
C548 11544 027550R JMP EMAT3
C549 11545 005400 EMAT2 CLB COMMA
C550 11546 055001R CPB MCKS DELIMITER?
C551 11547 017440R JSB EDELM YES

```

0552	11550	017322R	EMAT3	JSB DDLD	LOAD
0553	11551	000061R		DEF B1	NEXT
0554	11552	035061R		ISZ B1	ELEMENT
0555	11553	035061R		ISZ B1	
0556	11554	017426R		JSB ENOUT	OUTPUT IT
0557	11555	035063R		ISZ B2	ROW COMPLETE?
0558	11556	027545R		JMP EMAT2	NO
0559	11557	017461R		JSB OUTLN	YES, DO
0560	11560	017461R		JSB OUTLN	SPACING
0561	11561	062064R		LDA B2+1	RESET
0562	11562	072063R		STA B2	COLUMN COUNTER
0563	11563	035065R		ISZ B3	ARRAY EXHAUSTED?
0564	11564	027550R		JMP EMAT3	NO
0565	11565	065045R	EMAT4	LDB TEMPS	YES, MORE
0566	11566	055031R		CPB PRADD	STATEMENT?
0567	11567	025233R		JMP XEC4	NO
0568	11570	027500R		JMP EMAT1	YES
0569*					
0570	11571	075063R	EMAT5	STB B2	SAVE SYMBOL TABLE POINTER
0571	11572	017062R		JSB LDIMP	EXTRACT
0572	11573	012220R		AND OPMSK	NEXT OPERATOR
0573	11574	065045R		LDB TEMPS	STATEMENT
0574	11575	055031R		CPB PRADD	EXHAUSTED?
0575	11576	002400		CLA	YES
0576	11577	052206R		CPA B1100	::
0577	11600	017744R		JSB REDIM	YES, REDIMENSION ARRAY
0578	11601	062063R		LDA B2	LOAD
0579	11602	040055		ADA .2	ARRAY
0580	11603	160000		LDA 0,I	DIMENSIONS
0581	11604	017130R		JSB MDIM	SET
0582	11605	001100		ARS	ARRAY
0583	11606	003004		CMA,INA	ELEMENT
0584	11607	072065R		STA B3	COUNTER
0585	11610	015324R	EMAT6	JSB FOATA	FETCH VALUE
0586	11611	017342R		JSB DDST	STORE
0587	11612	000061R		DEF B1	
0588	11613	035061R		ISZ B1	IT
0589	11614	035061R		ISZ B1	
0590	11615	035065R		ISZ B3	ARRAY EXHAUSTED?
0591	11616	027610R		JMP EMAT6	NO
0592	11617	027565R		JMP EMAT4	YES
0593*					
0594	11620	017613R	EMAT7	JSB SSYMT	SAVE
0595	11621	005004		INB	BASE ADDRESS
0596	11622	160001		LDA 1,I	OF .DESTINATION
0597	11623	012222R		AND INF	
0598	11624	072065R		STA B3	ARRAY
0599	11625	075063R		STB B2	SAVE SYMBOL TABLE ADDRESS
0600	11626	044055		ADB .2	SAVE
0601	11627	160001		LDA 1,I	ITS
0602	11630	072066R		STA B3+1	DIMENSIONS
0603	11631	002404		CLA,INA	ASSUME MAT
0604	11632	073440R		STA EDELM	REPLACEMENT
0605	11633	017062R	EMAT8	JSB LDIMP	LOAD NEXT
0606	11634	035045R		ISZ TEMPS	OPERAND

0607	11635	002020	SSA	ARRAY FUNCTION?
0608	11636	027723R	JMP EMA11	YES
0609	11637	012176R	EMAT8 AND MSK1	NO
0610	11640	002003	SZA, RSS	SCALAR MULTIPLICATION?
0611	11641	027713R	JMP EMA10	YES
0612	11642	017613R	JSB SSYMT	NO
0613	11643	005004	INB	SAVE
0614	11644	160001	LDA 1,1	BASE
0615	11645	012222R	AND INF	
0616	11646	072061R	STA B1	ADDRESS AND
0617	11647	044055	ADB .2	DIMENSIONS
0618	11650	160001	LDA 1,1	OF FIRST
0619	11651	072062R	STA B1+1	SOURCE ARRAY
0620	11652	065045R	LDB TEMPS	STATEMENT
0621	11653	055031R	CPB PRADD	EXHAUSTED?
0622	11654	027674R	JMP EMAT9	YES
0623	11655	017062R	JSB LDIMP	NO EXTRACT
0624	11656	001727	ALF, ALF	AND
0625	11657	010072	AND .63	RECORD
0626	11660	040045	ADA M6	EMAT
0627	11661	073440R	STA EDELM	OPERATOR
0628	11662	017062R	JSB LDIMP	SAVE
0629	11663	012176R	AND MSK1	
0630	11664	017613R	JSB SSYMT	BASE ADDRESS
0631	11665	005004	INB	
0632	11666	160001	LDA 1,1	AND DIMENSIONS
0633	11667	012222R	AND INF	
0634	11670	072063R	STA B2	
0635	11671	044055	ADB .2	OF SECOND
0636	11672	160001	LDA 1,1	
0637	11673	072064R	STA B2+1	SOURCE ARRAY
0638*				
0639	11674	063440R	EMAT9 LDA EDELM	TRANSFER TO
0640	11675	043700R	ADA LMAP	APPROPRIATE
0641	11676	114000	JSB 0,1	ROUTINE
0642	11677	025233R	JMP XEC4	
0643*				
0644	11700	111700R	LMAP DEF LBASE-1,1	
0645	11701	012151R	LBASE DEF REPLC	
0646	11702	012134R	DEF ADD	
0647	11703	012144R	DEF SUB	
0648	11704	012314R	DEF MULT	
0649	11705	012212R	DEF SZER	
0650	11706	012167R	DEF LCON	
0651	11707	012220R	DEF LIDN	
0652	11710	012424R	DEF LINV	
0653	11711	012252R	DEF TRAN	
0654	11712	012162R	DEF SMULT	
0655*				
0656	11715	050065	EMA10 LDA .10	SET SMULT
0657	11714	073440R	STA EDELM	OPERATOR
0658	11715	015255R	JSB FEICH	EVALUATE
0659	11716	072057R	STA MLBX1	AND SAVE
0660	11717	075060R	STB MLBX1+1	SCALAR
0661	11720	055045R	ISZ TEMPS	GO TO

0662	11721	036045R	ISZ TEMPS	PROCESS
0663	11722	027633R	JMP EMAT0	SOURCE ARRAY
0664*				
0665	11723	001727	EMA11 ALF,ALF	EXTRACT
0666	11724	001722	ALF,RAL :::	
0667	11725	010071	AND .31	TYPE
0668	11726	040043	ADA M8	RECORD EMAT
0669	11727	073440R	STA EDELM	OPERATOR TYPE
0670	11730	040043	ADA M8	INV OR
0671	11731	002020	SSA	TRN?
0672	11732	027737R	JMP EMA12	NO
0673	11733	017062R	JSB LDIMP	YES, LOAD
0674	11734	036045R	ISZ TEMPS	SOURCE
0675	11735	036045R	ISZ TEMPS	ARRAY
0676	11736	027637R	JMP EMAT8	SYMBOL
0677*				
0678	11737	066045R	EMA12 LDB TEMPS	REDIMENSIONING
0679	11740	056031R	CPB PRADD	PART?
0680	11741	027674R	JMP EMAT9	NO
0681	11742	017744R	JSB REDIM	YES
0682	11743	027674R	JMP EMAT9	

0684*			*****	
0685*			SUBROUTINE TO REDIMENSION ARRAY	
0686*			*****	
0687	11744	000000	REDIM NOP	
0688	11745	016001R	JSB MCKS	EVALUATE
0689	11746	005727	BLF,BLF	AND SAVE
0690	11747	075066R	STB B3+1	ROW COUNT
0691	11750	006404	CLB,INB	LOAD DEFAULT COLUMN COUNT
0692	11751	036045R	ISZ TEMPS	SINGLE
0693	11752	017062R	JSB LDIMP	DIMENSION
0694	11753	012220R	AND OPMSK	ARRAY?
0695	11754	052203R	CPA B2400 ::	
0696	11755	027760R	JMP REDI1	YES
0697	11756	016001R	JSB MCKS	NO, EVALUATE COLUMN COUNT
0698	11757	036045R	ISZ TEMPS	MOVE PAST
0699	11760	036045R	REDI1 ISZ TEMPS	RIGHT BRACKET
0700	11761	046066R	ADB B3+1	PACK
0701	11762	075066R	STB B3+1	DIMENSIONS
0702	11763	062063R	LDA B2	STORE IN
0703	11764	040055	ADA .2	SYMBOL
0704	11765	174000	STB 0,I	TABLE
0705	11766	040052	ADA M1	COMPUTE
0706	11767	160000	LDA 0,I	PHYSICAL
0707	11770	017130R	JSB MDIM	ARRAY SPACE
0708	11771	072060R	STA MLBX1+1	SIZE
0709	11772	062066R	LDA B3+1	COMPUTE
0710	11773	017130R	JSB MDIM	NEW SIZE
0711	11774	003004	CMA,INA	NEW
0712	11775	042060R	ADA MLBX1+1	SIZE

C713	11776	002020	SSA	ACCEPTABLE?
C714	11777	015274R	JSB ERROR	NO
C715	12000	127744R E7	JMP REDIM,I	YES

```

C717*
C718*
C719*
C720 12001 000000 MCKS NOP
C721 12002 015255R JSB FETCH CALL FOR EVALUATION
C722 12003 017147R JSB SBFIX CONVERT TO INTEGER (ROUNDED)
C723 12004 005004 INB UNBIAS SUBSCRIPT
C724 12005 050001 LDA 1 PUT INTO (A)
C725 12006 040075 ADA M256 LESS THAN
C726 12007 002021 SSA,RSS 256?
C727 12010 027221R JMP E6-1 NO
C728 12011 126001R JMP MCKS,I YES, RETURN SUBSCRIPT IN (B)

```

```

C730*
C731*****
C732*****      MATRIX      ROUTINES      *****
C733*****
C734*CALL FOR MATRIX OPERATION IS MADE WITH FOUR*
C735*PARAMETERS,ROUTINE NUMBER AND ADDRESS OF  *
C736*SYMBOL TABLE OF THREE MATRICES. FOR SCALAR *
C737*MULT, LAST IS ADDRESS OF SCALAR VALUE      *
C738*OPERATION IS OF FORM  B3=B1 OP B2          *
C739*THE ADDRESS OF THE BASE ADDRESS OF MATRICES*
C740*IS GIVEN IN B1,B2,B3. THE DIMENSIONS OF A  *
C741*MATRIX IS GIVEN IN B(1)+1, ROWS IN MOST SIG*
C742*PART(MSP) AND COLUMN IN LEAST SIG PART(LSP)*
C743*****
C744*
C745*
C746*
C747*****
C748***      SUBROUTINE  GENERAL      ***
C749*****
C750*B3=B1 OP B2  SUBROUTINE COMPUTES AN ELEMENT*
C751*OF B3 AND INCREMENTS TO NEXT ELEMENT. THE  *
C752*OPERATION THAT IS PERFORMED AND            *
C753* THE MATRICES INCREMENTED ARE              *
C754* MODIFIED BY ROUTINES ADD, SUB, REPL      *
C755*SCALAR MULT, CON,ZERO,IDN. ROUTINE CHECKS  *
C756*COMPATIBILITY OF THREE MATRICES USING SUB *
C757*COMPAKE (PARAMETERS SUPPLIED IN REG A,B)  *
C758*****
C759*
C760 12012 000000  GENER NOP          SUBROUTINE GENERAL
C761 12013 062064R      LDA B2+1      LOAD DIM FOR MATRIX 2
C762 12014 065062R      LDB B1+1      LOAD DIM FOR MATRIX 1
C763 12015 015063R      JSB COMPR     CHECKS ROW AND COL DIM
C764*                                ARE COMPATIBLE
C001 12016 062062R GEN2  LDA B1+1      LOAD DIM FOR MATRIX 1
C002 12017 065066R      LDB B3+1      LOAD DIM FOR MATRIX 3
C003 12020 015063R      JSB COMPR     CHECK ROW AND COL DIM
C004 12021 100200      MPY T3         COLUMNS IN (A), ROWS IN T3
      12022 013273R
C005 12023 003004      CMA,INA
C006 12024 073312R      STA LPIV      -ROWS*COLUMNS
C007*                                COMPUTES B3=B1 OP B2
C008 12025 017322R LOOP  JSB DDLD     LOAD
C009 12026 000061R      DEF B1        NEXT
C010 12027 035061R      ISZ B1        SOURCE
C011 12030 035061R      ISZ B1        ELEMENT
C012 12031 125032R      JMP ++1,I
C013 12032 000000  MOD  BSS 1
C014 12033 017342R      JSB DDST      STORE
C015 12034 000065R      DEF B3        NEXT
C016 12035 035065R      ISZ B3        DESTINATION
C017 12036 035065R      ISZ B3        ELEMENT
C018 12037 050046      LDA M5
C019 12040 042062R      ADA MOD2

```

```

0020 12041 002021      SSA,RSS
0021 12042 025045R      JMP **3
0022 12043 035063R      ISZ B2
0023 12044 035063R      ISZ B2
0024 12045 037312R      ISZ LPIV
0025 12046 025025R      JMP LOOP      COMPUTE NEXT ELEMENT
0026 12047 125012R      JMP GENER,I
0027 12050 012050R MOD1  DEF *
0028 12051 017403R      JSB .FAD.
0029 12052 100063R      DEF B2,I
0030 12053 025033R      JMP MOD+1
0031 12054 017412R      JSB .FSB.
0032 12055 100063R      DEF B2,I
0033 12056 025033R      JMP MOD+1
0034 12057 017421R      JSB .FMP.
0035 12060 000123R      DEF MBXL
0036 12061 025033R      JMP MOD+1
0037 12062 000000 MOD2  BSS 1
0038*
0039*
0040*****
0041****      SUBROUTINE COMPARE      ****
0042*****
0043*ROUTINE COMPARES DIM OF TWO MATRICES GIVEN *
0044*THEIR DIM IN REGISTERS A,B      *
0045*DIMENSIONS ARE GIVEN IN B(I)+2      *
0046*****
0047*
0048 12063 000000  COMPR NOP
0049 12064 050001      CPA 1      EQUAL?
0050 12065 002001      RSS      YES
0051 12066 016274R LERR  JSB ERROR      NO
0052 12067 001727      ALF,ALF      SAVE
0053 12070 010074      AND MSK0      # OF
0054 12071 073273R      STA T3      ROWS
0055 12072 060001      LDA 1
0056 12073 010074      AND MSK0      SAVE #
0057 12074 073274R      STA T4      OF COLUMNS
0058 12075 125063R      JMP COMPR,I
0059*
0060*
0061*****
0062*****      SUBROUTINE LCHK      *****
0063*****
0064*TESTS THAT NO ELEMENT IN A MATRIX IS      *
0065*UNASSIGNED. ENTRY1 CHECKS MATRICES GIVEN BY*
0066*B1 AND B2 AND ENTRY 2 CHECKS ONLY B1      *
0067*****
0068*
0069 12076 000000  LCHK2 NOP
0070 12077 062076R      LDA LCHK2
0071 12100 072102R      STA LCHK1
0072 12101 025106R      JMP **5
0073 12102 000000  LCHK1 NOP
0074 12103 065063R      LDB B2      BASE ADDR

```

```

C075 12104 062064R      LDA B2+1      ROW AND COL DIM.
C076 12105 016112R      JSB LCHK4      TEST EACH TERM OF B2
C077 12106 065061R      LDB B1        BASE ADDR
C078 12107 062062R      LDA B1+1      ROW AND COL DIM.
C079 12110 016112R      JSB LCHK4      TEST EACH TERM OF B1
C080 12111 126102R      JMP LCHK1,I
C081*
C082 12112 000000      LCHK4 NOP        SUBROUTINE TO TEST TERMS
C083 12113 077276R      STB T6         SAVE
C084 12114 017130R      JSB MDIM       COMPUTE SIZE OF MATRIX
C085 12115 001100      ARS            SET NEGATIVE
C086 12116 003004      CMA,1NA
C087 12117 073277R      STA T7        COUNTER FOR ELEMENTS
C088 12120 017322R      LCHK6 JSB DDLD  LOAD
C089 12121 013276R      DEF T6
C090 12122 037276R      ISZ T6        ELEMENT
C091 12123 037276R      ISZ T6
C092 12124 052263R      CPA MNEG      COMPARE WITH PRESET QTY.
C093 12125 025127R      JMP *+2
C094 12126 025131R      JMP LCHK5
C095 12127 055264R      CFB MNEG+1
C096 12130 016274R      JSB ERROR     ERROR 'MAT UNASSIGNED'
C097 12131 037277R      LCHK5 ISZ T7  DONE?
C098 12132 025120R      JMP LCHK6     NO
C099 12133 126112R      JMP LCHK4,I   YES
C100*
C101*
C102*****
C103**** SUBROUTINE MATRIX ADD ****
C104*****
C105*B1,B2,B3 CONTAIN ADDRESS OF BASE ADDRESS OF*
C106*THREE MATRICES. ROUTINE EXECUTES B3=B1+B2 *
C107*BY MODIFYING INSTR IN ROUTINE GENERAL *
C108*****
C109*
C110 12134 000000      ADD      NOP
C111 12135 060054      LDA .1
C112 12136 072062R      ADD1     STA MOD2
C113 12137 042050R      ADA MOD1
C114 12140 072032R      STA MOD
C115 12141 016102R      JSB LCHK1     TEST B1,B2 FOR UNASSIGNED TERMS
C116 12142 016012R      JSB GENER    ROUTINE GENERAL
C117 12143 126134R      JMP ADD,I     EXIT TO MAIN PROGRAM
C118*
C119*
C120*****
C121**** SUBROUTINE MATRIX SUBTRACT ****
C122*****
C123*B1,B2,B3 CONTAIN ADDRESS OF BASE ADDRESS OF*
C124*THREE MATRICES. ROUTINE EXECUTES B3=B1-B2 *
C125*BY MODIFYING INSTR IN ROUTINE GENERAL *
C126*****
C127*
C128 12144 000000      SUB      NOP        LET
C129 12145 062144R      LDA SUB        ADD DO

```

C130	12146	072154R	STA ADD	RETURN
C131	12147	060057	LDA .4	
C132	12150	026136R	JMP ADD1	

```

C134*
C135*****
C136**** SUBROUTINE MATRIX REPLACE ****
C137*****
C138*B1,B3 GIVE ADDRESS OF BASE ADDRESS OF GIVEN*
C139*MATRIX AND RECEIVING MATRIX RESPECTIVELY *
C140*****
C141 12151 000000 REPLC NOP LET
C142 12152 062151R LDA REPLC GENER DO
C143 12153 072012R STA GENER RETURN
C144 12154 060064 LDA .9
C145 12155 072062R REPL1 STA MOD2
C146 12156 042050R ADA MOD1
C147 12157 072032R STA MOD
C148 12160 016076R JSB LCHK2 TEST B1 FOR UNASSIGNED ELEMENTS
C149 12161 025016R JMP GEN2
C150*
C151*
C152*****
C153**** SUBROUTINE MATRIX SCALAR MULT ****
C154*****
C155*B1,B3 GIVE ADDRESS OF BASE ADDRESS OF GIVEN*
C156*MATRIX AND RECEIVING MATRIX RESPECTIVELY *
C157*MBXL HOLDS ADDRESS OF SCALAR VALUE *
C158*****
C159*
C160 12162 000000 SMULT NOP LET
C161 12163 062162R LDA SMULT GENER DO
C162 12164 072012R STA GENER RETURN
C163 12165 060062 LDA .7
C164 12166 026155R JMP REPL1
C165*
C166*
C167*****
C168**** SUBROUTINE MATRIX CON ****
C169*****
C170*SETS MATRIX TO ALL ONES. B3 IS ADDRESS OF *
C171*BASE ADDRESS OF MATRIX. *
C172*****
C173*
C174 12167 000000 LCON NOP
C175 12170 062261R LDA HONE
C176 12171 064055 LDB .2
C177 12172 072057R LCON1 STA MLBX1 SET
C178 12173 075060R STB MLBX1+1 CONSTANT
C179 12174 062066R LDA B3+1
C180 12175 017130R JSB MDIM SET
C181 12176 001100 ARS ELEMENT
C182 12177 003004 CMA,INA COUNTER
C183 12200 073512R STA LPIV
C184 12201 062057R LDA MLBX1 LOAD
C185 12202 065060R LDB MLBX1+1 CONSTANT
C186 12203 017342R LCON2 JSB DDST STORE
C187 12204 000065R DEF B3 IN
C188 12205 035065R ISZ B3 NEXT

```

```

0189 12206 036065R      ISZ B3                ELEMENT
0190 12207 037312R      ISZ LPIV          DONE?
0191 12210 026203R      JMP LCON2          NO
0192 12211 126167R      JMP LCON,I        YES
0193*
0194*
0195*****
0196***** SUBROUTINE MATRIX ZERO *****
0197*****
0198*SETS MATRIX TO ZERO. B3 IS ADDRESS OF BASE *
0199*ADDRESS OF MATRIX. B1,B2 ARE REDUNDANT *
0200*SET B1=0 AND USE SUBROUTINE CON,ENTRY2 *
0201*****
0202*
0203 12212 000000 SZER NOP
0204 12213 062212R      LDA SZER          CONVERT
0205 12214 072167R      STA LCON
0206 12215 002400      CLA                LCON
0207 12216 006400      CLB
0208 12217 026172R      JMP LCON1          TO SZER
0209*
0210*
0211*****
0212***** SUBROUTINE MATRIX IDN *****
0213*****
0214*ROUTINE SETS UP IDENTITY MATRIX *
0215*B3 IS ADDRESS OF BASE ADDRESS OF MATRIX *
0216*USE SZER TO SET MATRIX TO ALL ZEROS. ON *
0217*RETURN CHECK FOR SQUARE MATRIX. *
0218*****
0219*
0220 12220 000000 LIDN NOP
0221 12221 062065R      LDA B3
0222 12222 073301R      STA T9          SAVE BASE ADDRESS
0223 12223 016212R      JSB SZER          SET ALL MATRIX TO ZERO
0224 12224 062066R      LDA B3+1        IS
0225 12225 001727      ALF,ALF          ARRAY
0226 12226 052066R      CPA B3+1        SQUARE?
0227 12227 001010      ALS,SLA          YES
0228 12230 026066R      JMP LERR          NO
0229 12231 012176R      AND MSK1        SAVE ROW
0230 12232 072057R      STA MLBX1        LENGTH
0231 12233 001100      ARS              SAVE
0232 12234 003004      CMA,INA          ROW
0233 12235 072060R      STA MLBX1+1      COUNTER
0234 12236 067301R      LDB T9          RESTORE
0235 12237 076065R      STB B3          B3
0236 12240 062261R LIDN1 LDA HONE        STORE
0237 12241 017055R      JSB STBI
0238 12242 006004      INB              1.0 ON
0239 12243 060055      LDA .2
0240 12244 017055R      JSB STBI          DIAGONAL
0241 12245 006004      INB
0242 12246 046057R      ADB MLBX1        MOVE TO NEXT DIAGONAL ELEMENT
0243 12247 036060R      ISZ MLBX1+1      DONE?

```

0244 12250 026240R
0245 12251 126220R

JMP LIDN1 NO
JMP LIDN,I YES

```

0247*
0248*****
0249**** SUBROUTINE TRANSPOSE *****
0250*****
0251*TRANSPOSE OF FORM B3(M,N)=T(B1(N,M)) *
0252*B1,B3 GIVE ADDRESS OF BASE ADDRESS OF GIVEN*
0253*AND RECEIVING MATRICES RESPECTIVELY. *
0254*****
0255*
0256 12252 000000 TRAN NGP
0257 12253 015076R JSB LCHK2 TEST B1 FOR UNASSIGNED TERMS
0258* CHECK DIMENSIONS
0259 12254 062066R LDA B3+1 PARAMETERS OF B3
0260 12255 001727 ALF,ALF INTERCHANGE ROW AND COLUMN
0261 12256 065062R LDB B1+1 PARAMETERS OF B1
0262 12257 015063R JSB COMPR SUBROUTINE COMPARE
0263* DIMENSIONS COMPATIBLE
0264 12260 109200 MPY T3 #OF COLUMNS IN (A), # OF ROWS IN T3
      12261 013273R
0265 12262 073312R STA LP1V PRODUCT OF ROW*COL
0266 12263 063274R LDA T4 SET
0267 12264 003004 CMA,INA COLUMN
0268 12265 073275R STA T5 COUNTER
0269* T6 IS INDICATOR TO SELECT
0270* WHICH ELEMENT IN A COL OF
0271* B1 IS TO BE TRANSPOSED
0272 12266 002400 TRAN1 CLA
0273 12267 073276R STA T6 SET T6=0
0274 12270 067276R LNEXT LDB T6 LOAD
0275 12271 005000 BLS
0276 12272 045061R ADB B1 NEXT ELEMENT
0277 12273 060001 LDA B
0278 12274 017505R JSB FTHOP OF COLUMN
0279 12275 017342R JSB DDST STORE
0280 12276 000065R DEF B3 IN
0281 12277 035065R ISZ B3 ROW
0282 12300 035065R ISZ B3
0283 12301 063276R LDA T6 SET T6=T6+T4
0284 12302 043274R ADA T4 T6 POINTS TO NEXT TERM IN
0285 12303 073276R STA T6 A COLUMN TO BE TRANSPOSED
0286 12304 053312R CPA LP1V TEST FOR LAST IN COL
0287 12305 025307R JMP **2
0288 12306 026270R JMP LNEXT
0289* SET BASE ADDRESS TO FIRST
0290* TERM IN NEXT COLUMN
0291 12307 035061R ISZ B1
0292 12310 035061R ISZ B1
0293 12311 037275R ISZ T5
0294 12312 025266R JMP TRAN1 TRANSPOSE NEXT COL
0295 12313 125252R JMP TRAN,I EXIT TO MAIN PROGRAM

```

```

C297*
C298*****
C299**** SUBROUTINE MATRIX MULT *****
C300*****
C301*ROUTINE IS OF FORM B3(M,P)=B1(M,N)*B2(N,P) *
C302*B1,B2,B3 ARE ADDRESSES OF BASE ADDRESSES OF*
C303*THREE MATRICES *
C304*****
C305*
C306 12314 000000 MULT NOP
C307 12315 015102R JSB LCHK1 TEST B1,B2 FOR UNASSIGNED TERMS
C308* CHECK DIMENSIONS
C309 12316 062066R LDA B3+1 PARAMETERS OF B3
C310 12317 010074 AND MSK0 SAVE COLUMN COUNT
C311 12320 073276R STA T6
C312 12321 062064R LDA B2+1 PARAMETERS OF B2
C313 12322 010074 AND MSK0
C314 12323 053276R CPA T6 COLUMNS EQUAL
C315 12324 002001 RSS IN NUMBER?
C316 12325 025066R JMP LERR NO
C317* COMBINE B3,B2 PARAMETERS
C318* INTO (M,N) AND COMPARE
C319* WITH THOSE OF B1
C320 12326 062066R LDA B3+1 PARAMETERS OF B3
C321 12327 010075 AND M256
C322 12330 070001 STA 1 STORE ROW IN MSP OF B
C323 12331 062064R LDA B2+1 PARAMETERS OF B2
C324 12332 001727 ALF,ALF GET ROW COUNT
C325 12333 010074 AND MSK0 IN (A)
C326 12334 040001 ADA 1 COMBINE A AND B
C327 12335 065062R LDB B1+1 PARAMETERS OF B1
C328 12336 015063R JSB COMPR COMPARE ROW AND COL
C329* DIMENSIONS ARE COMPATIBLE
C330* M,N ARE STORED IN T3,T4
C331* SAVE B2 AS .DESTROYED IN
C332 12337 062063R LDA B2 MULT
C333 12340 073275R STA T5
C334 12341 063273R LDA T3 SET
C335 12342 003004 CMA,INA ROW
C336 12343 073301R STA T9 COUNTER
C337 12344 063276R MULT4 LDA T6
C338 12345 003004 CMA,INA
C339 12346 073302R STA T10 SET COUNTER
C340 12347 063275R LDA T5
C341 12350 072063R STA B2 RESTORE BASE ADDRESS B2
C342 12351 002400 MULT3 CLA
C343 12352 073303R STA T11 COUNTER FOR B2. INCR BY
C344* 2*P AND POINTS TO NEXT TERM
C345* IN COL TO BE MULTIPLIED
C346 12353 073304R STA T12 COUNTER FOR B1. INCR BY 2
C347* AND POINTS TO NEXT TERM
C348* IN ROW TO BE MULTIPLIED
C349 12354 005400 CLB
C350 12355 017342R JSB DDST CLEAR TO ZERO
C351 12356 000065R DEF B3

```

C352	12357	065061R	MULT2	LDB B1	COMPUTE PROD OF ONE TERM
C353	12360	047304R		ADB T12	IN ROW BY ONE TERM IN COL
C354	12361	077310R		STB T18	
C355	12362	065063R		LDB B2	
C356	12363	047303R		ADB T11	
C357	12364	069001		LDA B	
C358	12365	017505R		JSB FTHOP	
C359	12366	017421R		JSB .FMP.	
C360	12367	113310R		DEF T18,I	
C361	12370	017403R		JSB .FAD.	COMPUTES RUNNING SUM
C362	12371	109065R		DEF B3,I	
C363	12372	017342R		JSB DDST	
C364	12373	000065R		DEF B3	
C365	12374	037304R		ISZ T12	SELECT NEXT TERM IN ROW
C366	12375	037304R		ISZ T12	
C367	12376	063276R		LDA T6	SELECT NEXT TERM IN COL
C368	12377	001000		ALS	
C369	12400	043303R		ADA T11	
C370	12401	073303R		STA T11	
C371*					TEST IF HAVE MULT ONE ROW
C372*					BY ONE COLUMN
C373	12402	063274R		LDA T4	
C374	12403	001000		ALS	
C375	12404	053304R		CPA T12	
C376	12405	025407R		JMP *+2	
C377	12406	025357R		JMP MULT2	MULT AND ADD IN NEXT TERM
C378*					SUMMATION OF PRODUCTS FOR
C379*					ONE TERM OF B3 IS DONE
C380*					MULT SAME ROW BY NEXT COL
C381	12407	035065R		ISZ B3	INCR RECEIVING MAT
C382	12410	035065R		ISZ B3	
C383	12411	035063R		ISZ B2	BASE ADDRESS OF NEXT COL
C384	12412	035063R		ISZ B2	
C385*					TEST IF HAVE MULT ONE ROW
C386*					BY ALL COLUMNS
C387	12413	037302R		ISZ T10	SKIP IF INNERPRODUCT DONE
C388	12414	025351R		JMP MULT3	COMPUTE SAME ROW*NEXT COL
C389*					SELECT NEXT ROW
C390	12415	063274R		LDA T4	
C391	12416	001000		ALS	
C392	12417	042061R		ADA B1	
C393	12420	072061R		STA B1	ADDRESS OF NEXT ROW
C394	12421	037301R		ISZ T9	
C395	12422	025344R		JMP MULT4	MULT ROW BY ALL COLUMNS
C396	12423	125314R		JMP MULT,I	EXIT TO MAIN PROGRAM

```

C398*
C399*****
C400**** SUBROUTINE MATRIX INVERT ****
C401*****
C402*OPERATION OF FORM MAT B3 = INV B1 *
C403*B1,B3 ARE ADDRESSES OF BASE ADDRESS OF *
C404*MATRIX TO BE INVERTED AND RECEIVING MATRIX *
C405*RESPECTIVELY. B2 IS REDUNDANT. METHOD USED *
C406*IS GAUSSIAN ELIMINATION WITH COLUMN *
C407*PIVOTING *
C408*****
C409*
C410 12424 000000 LINV NOP SUBROUTINE MATRIX INVERT
C411 12425 015076R JSB LCHK2 TEST B1 FOR UNASSIGNED TERMS
C412 12426 062062R LDA B1+1 DIMENSIONS OF MATRIX B1
C413 12427 065066R LDB B3+1 DIMENSIONS OF MATRIX B3
C414 12430 015063R JSB COMPR CHECK DIMENSIONS
C415* ROW AND COL VALUES T3,T4
C416* MAKE COPY OF MATRIX B1
C417* IN FREE CORE
C418 12431 062065R LDA B3 SAVE
C419 12432 073305R STA T13 B3
C420 12433 062062R LLA B1+1 COMPUTE SIZE
C421 12434 017130R JSB MDIM OF MATRIX
C422 12435 003004 CMA,INA
C423 12436 001100 ARS SAVE
C424 12437 073272R STA T2 ELEMENT
C425 12440 001000 ALS COUNTER
C426 12441 065007R LDB FDISK SAVE
C427 12442 005004 INB ADDRESS OF
C428 12443 075063R STB B2 FREE CORE
C429 12444 075065R STB B3
C430 12445 007004 CMB,INB COMPUTE SIZE OF
C431 12446 045003R ADB LWAM FREE CORE AREA
C432 12447 040001 ADA 1 ENOUGH
C433 12450 002020 SSA CORE LEFT?
C434 12451 027240R JMP E1 NO
C435 12452 015151R JSB REPLC YES, COPY SOURCE MATRIX
C436 12453 063305R LDA T13 RESTORE
C437 12454 072065R STA B3 B3
C438 12455 015220R JSB LIDN SET DESTINATION TO IDENTITY
C439 12456 063305R LDA T13 RESTORE ITS
C440 12457 072065R STA B3 BASE ADDRESS
C441 12460 002400 CLA INITIALIZE
C442 12461 073304R STA T12 MAXIMUM
C443 12462 073305R STA T13 ELEMENT
C444 12463 062063R LDA B2 COPY B2 INTO B1 AS
C445 12464 072061R STA B1 B2 NEEDED LATER
C446 12465 017322R LIN11 JSB DDLD LOAD
C447 12466 000061R DEF B1 NEXT
C448 12467 035061R ISZ B1 ELEMENT
C449 12470 035061R ISZ B1
C450 12471 035061R ISZ B1
C451 12472 002020 SSA GET ABSOLUTE VALUE
C452 12473 017217R JSB ARINV IF NUMBER IS NEGATIVE

```

0453	12474	073310R	STA T18	SAVE NUMBER
0454	12475	077311R	STB T19	
0455	12476	015003X	FSB T12	SUBTRACT EXISTING MAX. VALUE
	12477	013304R		
0456	12500	002020	SSA	SKIP AND SWAP IF POSITIVE
0457	12501	025506R	JMP LIN10	
0458	12502	063310R	LDA T18	SWAP
0459	12503	067311R	LDB T19	
0460	12504	073304R	STA T12	
0461	12505	077305R	STB T13	
0462	12506	037272R	LIN10 ISZ T2	ALL ELEMENTS EXHAUSTED?
0463	12507	025465R	JMP LIN11	NO
0464	12510	063304R	LDA T12	COMPUTE RELATIVE TOLERANCE
0465	12511	067305R	LDB T13	TOL=ABSOLUTE TOL * MAX VALUE
0466	12512	015001X	FMP T16	ABSOLUTE TOLERANCE
	12513	013306R		
0467	12514	072057R	STA MLBX1	RELATIVE
0468	12515	075060R	STB MLBX1+1	TOLERANCE
0469	12516	002400	CLA	INITIALIZE PIVOT
0470	12517	073312R	STA LPIV	
0471	12520	037274R	ISZ T4	REQUIRE CONSTANT (ROW+1)
0472	12521	037312R	LINV1 ISZ LPIV	SELECT NEXT PIVOT
0473	12522	063312R	LDA LPIV	TEST IF HAVE PROCESSED
0474	12523	053274R	CPA T4	LAST PIVOT
0475	12524	125424R	JMP LINV,I	NORMAL EXIT TO MAIN PROG
0476*				SCAN PIVOTAL COLUMN FOR
0477*				LARGEST ELEMENT
0478	12525	063312R	LDA LPIV	COMPUTE ADDRESS OF PIVOT
0479	12526	067312R	LDB LPIV	COLUMN USING ROUTINE LWHR
0480	12527	073272R	STA T2	ROW COUNTER
0481	12530	017052R	JSB LWHR	ON RETURN, ADDRESS IN A
0482	12531	073271R	STA T1	
0483	12532	002400	CLA	
0484	12533	073304R	STA T12	T12,T13 IS STORE
0485	12534	073305R	STA T13	FOR GREATEST VALUE
0486	12535	017322R	LINV2 JSB UDL0	LOAD FP NUMBER
0487	12536	013271R	DEF T1	
0488	12537	002020	SSA	OBTAIN ABSOLUTE VALUE
0489	12540	017217R	JSB ARINV	IF NUMBER IS NEGATIVE
0490	12541	073310R	STA T18	STORE VALUE OF FP NUMBER
0491	12542	077311R	STB T19	
0492	12543	015003X	FSB T12	SUBTRACT EXISTING LARGEST VALUE
	12544	013304R		
0493	12545	002020	SSA	SKIP AND SWAP IF POSITIVE
0494	12546	025555R	JMP LINV7	T2 STILL CONTAINS MAX VALUE
0495	12547	063310R	LDA T18	STORE NEW MAX VALUE
0496	12550	067311R	LDB T19	
0497	12551	073304R	STA T12	
0498	12552	077305R	STB T13	
0499	12553	063272R	LDA T2	SET T5 TO POSITION IN
0500	12554	073275R	STA T5	COLUMN OF MAX VALUE
0501	12555	037272R	LINV7 ISZ T2	
0502	12556	063272R	LDA T2	TEST FOR LAST TERM IN COL
0503	12557	053274R	CPA T4	
0504	12560	025566R	JMP LINV8	SWAP ROWS

C505	12561	063273R	LDA T3	COMPUTE
C506	12562	001000	ALS	NEXT ADDRESS
C507	12563	043271R	ADA T1	IN PIVOT
C508	12564	073271R	STA T1	COLUMN
C509	12565	025535R	JMP LINV2	SELECT NEXT TERM
C510*				SWAP ROWS LPIV AND T5
C511	12566	063312R	LINV8 LDA LPIV	COMPUTE ADDRESS
C512	12567	005404	CLB,INB	OF PIVOTAL ROW
C513	12570	017052R	JSB LWHR	
C514	12571	073271R	STA T1	ADDRESS OF PIVOTAL ROW
C515	12572	063275R	LDA T5	
C516	12573	005404	CLB,INB	
C517	12574	017052R	JSB LWHR	
C518	12575	073272R	STA T2	ADDR OF ROW TO BE SWAPPED
C519	12576	063312R	LDA LPIV	
C520	12577	005404	CLB,INB	COMPUTE ADDRESS OF
C521	12600	017064R	JSB LWHR2	PIVOTAL ROW IN I-MATRIX
C522	12601	073301R	STA T9	
C523	12602	073302R	STA T10	KEEP COPY
C524	12603	063275R	LDA T5	
C525	12604	005404	CLB,INB	COMPUTE ADDR OF ROW TO
C526	12605	017064R	JSB LWHR2	BE SWAPPED IN I-MATRIX
C527	12606	073303R	STA T11	
C528	12607	063273R	LDA T3	
C529	12610	003004	CMA,INA	
C530	12611	073304R	STA T12	COUNTER FOR TERMS IN A ROW
C531	12612	017322R	LINV3 JSB DDLD	SWAP ONE ELEMENT OF ROW
C532	12613	013271R	DEF T1	
C533	12614	073310R	STA T18	
C534	12615	077311R	STB T19	
C535	12616	017322R	JSB DDLD	
C536	12617	013272R	DEF T2	
C537	12620	017342R	JSB DDST	
C538	12621	013271R	DEF T1	
C539	12622	037271R	ISZ T1	
C540	12623	037271R	ISZ T1	
C541	12624	063310R	LDA T18	
C542	12625	067311R	LDB T19	
C543	12626	017342R	JSB DDST	
C544	12627	013272R	DEF T2	
C545	12630	037272R	ISZ T2	
C546	12631	037272R	ISZ T2	
C547	12632	017322R	JSB DDLD	SWAP 1 ELEMENT OF ROW OF I MAT
C548	12633	013301R	DEF T9	
C549	12634	073310R	STA T18	
C550	12635	077311R	STB T19	
C551	12636	017322R	JSB DDLD	
C552	12637	013303R	DEF T11	
C553	12640	017342R	JSB DDST	
C554	12641	013301R	DEF T9	
C555	12642	037301R	ISZ T9	
C556	12643	037301R	ISZ T9	
C557	12644	063310R	LDA T18	
C558	12645	067311R	LDB T19	
C559	12646	017342R	JSB DDST	

0560	12647	013303R	DEF T11	
0561	12650	037303R	ISZ T11	
0562	12651	037303R	ISZ T11	
0563	12652	037304R	ISZ T12	SKIP IF DONE
0564	12653	025612R	JMP LINV3	SWAP NEXT ELEMENT
0565*				HAVE LARGEST ELEMENT IN
0566*				PIVOTAL POSITION. FIND
0567*				VALUE AND TEST TO ZERO
0568*				FOR SINGULAR MATRIX
0569	12654	063312R	LDA LPIV	COMPUTE
0570	12655	067312R	LDB LPIV	ADDRESS OF
0571	12656	017052R	JSB LWHR	PIVOT
0572	12657	073271R	STA T1	ELEMENT
0573	12660	017322R	JSB DDLD	PIVOT VALUE
0574	12661	013271R	DEF T1	
0575	12662	002020	SSA	OBTAIN ABSOLUTE VALUE
0576	12663	017217R	JSB ARINV	IF NUMBER IS NEGATIVE
0577	12664	016003X	FSB MLBX1	SUBTRACT TOLERANCE AND
	12665	000057R		
0578	12666	002020	SSA	COMPARE TO ZERO
0579	12667	016274R	JSB ERROR	PRINT 'NEARLY SING MATRIX'
0580*				DIVIDE PIVOT ROW AND ROW
0581*				IN I-MAT BY PIVOT VALUE
0582	12670	063271R	LDUM1 LDA T1	ADDRESS OF PIOT ELEMENT
0583	12671	073272R	STA T2	
0584	12672	062261R	LDA HONE	LOAD
0585	12673	064055	LDB .2	1.0
0586	12674	017430R	JSB .FDV.	
0587	12675	113271R	DEF T1,I	
0588	12676	073310R	STA T18	INVERSE OF PIVOT
0589	12677	077311R	SIB T19	
0590*				MULT ROW BY 1/PIVOT
0591*				STARTING AT PIVOT+1
0592	12700	063312R	LDA LPIV	
0593	12701	073303R	STA T11	COUNTER FOR ROW
0594	12702	037303R	LINV6 ISZ T11	INCREMENT COUNTER
0595	12703	063303R	LDA T11	
0596	12704	053274R	CPA T4	TEST FOR END OF ROW
0597	12705	025717R	JMP LIN12	
0598	12706	057272R	ISZ T2	ADDRESS OF NEXT ELEMENT
0599	12707	057272R	ISZ T2	
0600	12710	017322R	JSB DDLD	
0601	12711	013272R	DEF T2	
0602	12712	016001X	FMP T18	
	12713	013310R		
0603	12714	017342R	JSB DDST	
0604	12715	013272R	DEF T2	
0605	12716	025702R	JMP LINV6	
0606*				MULT ROW IN I-MATRIX BY
0607*				1/PIVOT. SKIP IF ELEMENT=0
0608	12717	063302R	LIN12 LDA T10	ADDRESS OF PIVOT ROW
0609	12720	073275R	STA T5	IN I-MATRIX
0610	12721	063273R	LDA T3	
0611	12722	003004	CMA,1NA SET	
0612	12723	073303R	STA T11	ROW COUNTER

0613	12724	017322R	LIN13	JSB DDLD	
0614	12725	013275R		DEF T5	
0615	12726	002005		SZA,RSS	SKIP MULTIPLICATION IF ZERO
0616	12727	006002		SZB	
0617	12730	026732R		JMP *+2	NOT ZERO
0618	12731	026736R		JMP LIN14	ZERO
0619	12732	016001X		FMP T18	
	12733	013310R			
0620	12734	017342R		JSB DDST	
0621	12735	013275R		DEF T5	
0622	12736	037275R	LIN14	ISZ T5	NEXT ELEMENT IN I-MATRIX
0623	12737	037275R		ISZ T5	
0624	12740	037303R		ISZ T11	DONE?
0625	12741	026724R		JMP LIN13	NO
0626*					PERFORM ROW MANIPULATIONS
0627*					AND SUBTRACTIONS TO REDUCE
0628*					PIVOT COLUMN TO ZERO
0629	12742	002400		CLA	
0630	12743	072061R		STA B1	
0631	12744	036061R	LINV4	ISZ B1	SELECT NEXT ROW
0632	12745	062061R		LDA B1	
0633	12746	053274R		CPA T4	TEST FOR LAST ROW
0634	12747	026521R		JMP LINV1	SELECT NEXT PIVOT
0635	12750	053312R		CPA LPIV	TEST TO SKIP PIVOTAL ROW
0636	12751	026744R		JMP LINV4	SKIP PIVOTAL ROW
0637	12752	062061R		LDA B1	
0638	12753	006404		CLB,INB	
0639	12754	017064R		JSB LWHR2	ADDRESS OF ROW TO BE TRANSFORMED
0640	12755	073303R		STA T11	IN I-MATRIX
0641*					COMPUTE MULTIPLIER WHICH
0642*					IS THAT ELEMENT IN ROW TO
0643*					BE TRANSFORMED WHICH LIES
0644*					IN THE PIVOTAL COLUMN
0645	12756	062061R		LDA B1	
0646	12757	067312R		LDB LPIV	
0647	12760	017052R		JSB LWHR	
0648	12761	073301R		STA T9	SAVE ADDRESS
0649	12762	017505R		JSB FTHOP	
0650	12763	073277R		STA T7	VALUE OF MULTIPLIER
0651	12764	077300R		STB T8	
0652*					DO ELIMINATION OF ROWS IN
0653*					ORIGINAL MATRIX. START AT
0654*					COLUMN LPIV+1
0655	12765	063312R		LDA LPIV	
0656	12766	073305R		STA T13	COUNTER
0657	12767	063271R		LDA T1	
0658	12770	073272R		STA T2	
0659	12771	037305R	LINV5	ISZ T13	
0660	12772	063305R		LDA T13	
0661	12773	053274R		CPA T4	TEST FOR LAST TERM IN ROW
0662	12774	027016R		JMP LIN15	
0663	12775	037301R		ISZ T9	T9 IS ADDRESS OF
0664	12776	037301R		ISZ T9	ELEMENT TO BE CHANGED
0665	12777	037272R		ISZ T2	T2 IS ADDR OF CORRESPONDING
0666	13000	037272R		ISZ T2	ELEMENT IN PIVOTAL ROW

```

0667 15001 063277R LDA T7
0668 15002 067500R LDB T8
0669 15003 017421R JSB .FMP.
0670 15004 113272R DEF T2,I
0671 15005 073310R STA T18 MULTIPLIER*VALUE IN
0001 15006 077311R STB T19 PIVOT ROW
0002 15007 017522R JSB DDLD
0003 15010 013301R DEF T9
0004 15011 016003X FSB T18
        15012 013310R
0005 15013 017342R JSB DDST TRANSFORM ELEMENT
0006 15014 013301R DEF T9
0007 15015 026771R JMP LINV5 SELECT NEXT TERM
0008* DO ELIMINATION OF ROWS IN
0009* IDENTITY MATRIX. START AT
0010* BEGINNING OF ROW AND LEAVE
0011* ELEMENT UNCHANGED WHEN ZERO
0012* IN PIVOTAL ROW.
0013 15016 063302R LIN15 LDA T10 ADDRESS OF
0014 15017 073275R STA T5 PIVOTAL ROW
0015 15020 063273R LDA T3
0016 15021 003004 CMA,INA SET
0017 15022 073305R STA T13 COUNTER
0018 15023 017322R LIN18 JSB DDLD
0019 15024 013275R DEF T5
0020 15025 037275R ISZ T5
0021 15026 037275R ISZ T5
0022 15027 002003 SZA,RSS SKIP IF ZERO
0023 15030 006002 SZB
0024 15031 027033R JMP *+2 NOT ZERO
0025 15032 027045R JMP LIN17 ZERO
0026 15033 016001X FMP T7 MULTIPLY BY THE MULTIPLIER
        15034 013277R
0027 15035 073310R STA T18
0028 15036 077311R STB T19
0029 15037 017322R JSB DDLD
0030 15040 013303R DEF T11
0031 15041 016003X FSB T18
        15042 013310R
0032 15043 017342R JSB DDST
0033 15044 013303R DEF T11
0034 15045 037303R LIN17 ISZ T11
0035 15046 037303R ISZ T11
0036 15047 037305R ISZ T13
0037 15050 027023R JMP LIN18 SELECT NEXT TERM
0038 15051 026744R JMP LINV4 ELIMINATE NEXT ROW
0039*
0040*
0041*****
0042***** SUBROUTINE LWHR *****
0043*****
0044*SUBROUTINE COMPUTES ADDRESS OF AN ELEMENT *
0045*IN MATRIX GIVEN BY B2. ROW AND COL VALUES *
0046*ARE SUPPLIED IN A,B. ADDRESS IS LEFT IN A *
0047*ENTRY LWHR2 COMPUTES ADDR IN MAT B3 *
```

```

C048*****
C049*
C050 13052 000000 LWHR NOP
C051 13053 077277R STB T7 SAVE COLUMN #
C052 13054 040052 ADA M1
C053 13055 100200 MPY T3 (A-1)*T3
      13056 013273R
C054 13057 043277R ADA T7
C055 13060 040052 ADA M1 +(B-1)
C056 13061 001000 ALS
C057 13062 042063R ADA B2 DDR=B2+2((A-1)*T3+(B-1))
C058 13063 127052R JMP LWHR,I
C059 13064 000000 LWHR2 NOP
C060 13065 077277R STB T7
C061 13066 040052 ADA M1
C062 13067 100200 MPY T3
      13070 013273R
C063 13071 043277R ADA T7
C064 13072 040052 ADA M1
C065 13073 001000 ALS
C066 13074 042063R ADA B3
C067 13075 127064R JMP LWHR2,I
C068*
C069**
C070*** RETRIEVE AND STORE SUB-PROGRAM PARAMETERS0***
C071**
C072*
C073 13076 000000 SENTER NOP
C074 13077 073121R STA SORCE
C075 13100 167076R LDB SENTER,I
C076 13101 077122R STB DEST
C077 13102 007004 CMB,INB
C078 13103 047076R ADB SENTER
C079 13104 044051 ADB M2
C080 13105 007004 CMB,INB
C081 13106 077123R STB CNTR
C082 13107 037076R ISZ SENTER
C083 13110 163121R .LOOP LDA SORCE,I
C084 13111 173122R STA DEST,I
C085 13112 063121R LDA SORCE
C086 13113 040052 ADA M1
C087 13114 073121R STA SORCE
C088 13115 037122R ISZ DEST
C089 13116 037123R ISZ CNTR
C090 13117 027110R JMP .LOOP
C091 13120 127076R JMP SENTER,I
C092 13121 000000 SORCE NOP
C093 13122 000000 DEST NOP
C094 13123 000000 CNTR NOP
C095 13124 000000 EXECR NOP GET RECORD FROM CRT/TICKY-TICKY
C096 13125 077260R STB BUFFER
C097 13126 073267R STA BUFFL
C098 13127 063124R LDA EXECR
C099 13130 073270R STA RET
C100 13131 060054 LDA .1

```

```

C101 13132 073265R STA RCODE
C102 13133 063136R LDA 0401
C103 13134 073266R STA CONWD
C104 13135 027254R JMP IEXEC
C105 13136 000401 0401 OCT 401
C106 13137 000000 EXECUT NOP GET RECORD FROM PHOTO READER
C107 13140 077260R STB BUFFER
C108 13141 003004 CMA,INA
C109 13142 073267R STA BUFFL
C110 13143 063137R LDA EXECUT
C111 13144 073270R STA RET
C112 13145 060054 LDA .1
C113 13146 073265R STA RCODE
C114 13147 060060 LDA .5
C115 13150 073266R STA CONWD
C116 13151 027254R JMP IEXEC
C117 13152 000000 EXECUS NOP SET END OF TAPE
C118 13153 016005X JSB EXEC
C119 13154 013157R DEF **3
C120 13155 000056 DEF .3
C121 13156 013160R DEF CON2
C122 13157 127152R JMP EXECUS,I
C123 13160 000705 CON2 OCT 000705
C124*
C125*** ENTER WITH CHARACTER COUNT IN THE 'A' REGISTER
C126*** AND THE BUFFER ADDRESS IN THE 'B' REGISTER.
C127*
C128 13161 000000 EXECW NOP WRITE RECORD ON CRT
C129 13162 077260R STB BUFFER
C130 13163 002021 SSA,RSS
C131 13164 003004 CMA,INA
C132 13165 073267R STA BUFFL
C133 13166 063161R LDA EXECW
C134 13167 073270R STA RET
C135 13170 060055 LDA .2
C136 13171 073265R STA RCODE
C137 13172 102501 LIA 01
C138 13173 012261R AND HALF
C139 13174 067264R LDB XIT
C140 13175 006021 SSB,RSS
C141 13176 027201R JMP **3
C142 13177 002002 SZA
C143 13200 027575R JMP RDYPT
C144 13201 102501 LIA 01
C145 13202 012263R AND MNEG
C146 13203 002002 SZA
C147 13204 060061 LDA .6
C148 13205 002003 SZA,RSS
C149 13206 060054 LDA .1
C150 13207 073266R STA CONWD
C151 13210 027254R JMP IEXEC
C152*
C153*** (ADDED 7-APR-71)
C154*
C155 13211 000000 LPUN NOP

```

```

0156 13212 006002      SZB
0157 13213 027232R     JMP LPUN1
0158 13214 062252R     LDA D133
0159 13215 073231R     STA CNTH0
0160 13216 016005X LPUN3 JSB EXEC      PUNCH RECORD ON H.S. PUNCH
0161 13217 013224R     DEF **5
0162 13220 000055      DEF .2
0163 13221 013227R     DEF CON1
0164 13222 013230R     DEF BUFR1
0165 13223 000054      DEF .1
0166 13224 037231R     ISZ CNTH0
0167 13225 027216R     JMP LPUN3
0168 13226 127211R     JMP LPUN,I
0169 13227 000104      CON1 OCT 104
0170 13230 000000      BUFR1 NOP
0171 13231 000000      CNTH0 NOP
0172 13232 077260R LPUN1 STB BUFFER
0173 13233 003004      CMA,INA
0174 13234 073267R     STA BUFL
0175 13235 063211R     LDA LPUN
0176 13236 073270R     STA RET
0177 13237 060057      LDA .4
0178 13240 073266R     STA CONWD
0179 13241 027252R     JMP LPUN2
0180 13242 000000      EXECP NOP      LIST RECORD ON LINE PRINTER
0181 13243 077260R     STB BUFFER
0182 13244 003004      CMA,INA
0183 13245 073267R     STA BUFL
0184 13246 063242R     LDA EXECP
0185 13247 073270R     STA RET
0186 13250 060061      LDA .6
0187 13251 073266R     STA CONWD
0188 13252 060055      LPUN2 LDA .2
0189 13253 073265R     STA RCODE
0190          EXT EXEC
0191 13254 016005X IEXEC JSB EXEC
0192 13255 013262R     DEF **5
0193 13256 013265R     DEF RCODE
0194 13257 013266R     DEF CONWD
0195 13260 000000      BUFFER NOP
0196 13261 013267R     DEF BUFL
0197 13262 060001      LDA 1
0198 13263 127270R     JMP RET,I
0199*
0200*
0201 13264 000000      XIT BSS 1
0202 13265 000000      RCODE BSS 1
0203 13266 000000      CONWD BSS 1
0204 13267 000000      BUFL BSS 1
0205 13270 000000      RET BSS 1
0206*
0207*
0208*****
0209*          CONSTANTS          *
0210*****

```

```

0211*
0212 13271 000000 T1 BSS 1      TEMPORARY CONSTANTS
0213 13272 000000 T2 BSS 1
0214 13273 000000 T3 BSS 1
0215 13274 000000 T4 BSS 1
0216 13275 000000 T5 BSS 1
0217 13276 000000 T6 BSS 1
0218 13277 000000 T7 BSS 1
0219 13300 000000 T8 BSS 1
0220 13301 000000 T9 BSS 1
0221 13302 000000 T10 BSS 1
0222 13303 000000 T11 BSS 1
0223 13304 000000 T12 BSS 1
0224 13305 000000 T13 BSS 1
0225 13306 041433 T16 DEC +1E-6    ABSOLUTE TOLERANCE
    13307 157733
0226 13310 000000 T18 BSS 1
0227 13311 000000 T19 BSS 1
0228 13312 000000 LPIV BSS 1
0229 13313 000000 FINIS BSS 700
0230 14607 000000 LWAMA BSS 1
0231 14610 000000 SBUF BSS 97
0232 14751 000000 SYMTB BSS 64
0233**
0234*** PERFORM LDA 1,I
0235**
0236 15051 000000 LDBI NOP
0237 15052 060001 LDA B
0238 15053 017204R JSB DREAD
0239 15054 127051R JMP LDBI,I
0240**
0241*** PERFORM STA 1,I
0242**
0243 15055 000000 STBI NOP
0244 15056 077130R STB TMP2
0245 15057 017151R JSB DSTOR
0246 15060 015130R DEF TMP2
0247 15061 127055R JMP STBI,I
0248**
0249*** PERFORM LDA TEMPS,I
0250**
0251 15062 000000 LDIMP NOP
0252 15063 062045R LDA TEMPS
0253 15064 017204R JSB DREAD
0254 15065 127062R JMP LDIMP,I

```

DISK BUFFERING ROUTINES

0256*

0257*****

0258*** DISK BUFFER SUBROUTINES *****

0259*****

0260* ADDRESS CONVERSION SUBROUTINE

0261* ENTER WITH ADDRESS IN A REG

0262*

0263 15066 000000 AC0N NOP

0264 15067 005400 CLB

0265 15070 077121R STB SCNT SET SECTOR COUNT TO ZERO

0266 15071 065011C LDB FTRAK

0267 15072 077122R STB TCNT SET TRACK COUNT TO START OF W.A.

0268 15073 002020 SSA ADDR > 32767

0269 15074 015274R JSB ERROR NO, ERROR!

0270 15075 043126R DIA1 ADA M3072 COMPUTE TRACK ADDR.

0271 15076 002021 SSA,RSS

0272 15077 027105R JMP INCTR

0273 15100 043124R ADA .3072

0274 15101 043125R DIA2 ADA M128 COMPUTE SECTOR ADDR.

0275 15102 002021 SSA,RSS

0276 15103 027107R JMP INCSC

0277 15104 127066R JMP AC0N,I

0278*

0279*

0280 15105 037122R INCTR ISZ TCNT

0281 15106 027075R JMP DIA1

0282*

0283 15107 037121R INCSC ISZ SCNT

0284 15110 067121R LDB SCNT

0285 15111 057135R CPH SIZE

0286 15112 017114R JSB ITRK

0287 15113 027101R JMP DIA2

0288*

0289 15114 000000 ITRK NOP

0290 15115 037122R ISZ TCNT

0291 15116 005400 CLB

0292 15117 077121R STB SCNT

0293 15120 127114R JMP ITRK,I

0294*

0295*

0296 15121 000000 SCNT BSS 1

0297 15122 000000 TCNT BSS 1

0298 15123 000000 TMP BSS 1

0299 00172 .128 EQU B200

0300 15124 005000 .3072 DEC 3072

0301 15125 177600 M128 DEC -128

0302 15126 172900 M3072 DEC -3072

0303 15127 000000 TMP1 BSS 1

0304 15130 000000 TMP2 BSS 2

0305 15132 000000 TMP3 BSS 1

0306 15133 000000 TMP4 BSS 1

0307 15134 000000 TMP5 BSS 1

0308 15135 000000 SIZE BSS 1

OF SECTORS PER DISK TRACK

0309 15136 000102 CW1 OCT 102

0310*

DISK BUFFERING ROUTINES

C311* SUB TO READ INTERMEDIATE CODE FROM DISK BUFFER

C312*

```

C313 15137 000000 READ1 NOP
C314 15140 015005X JSB EXEC
C315 15141 015150R DEF *+7
C316 15142 000054 DEF .1
C317 15143 015136R DEF CW1
C318 15144 100004C DEF BGNPG,I

```

FG 024

LN 0319 DEF BFRL

```

C319 15145 000000 DEF BFRL
C320 15146 015122R DEF TCNT
C321 15147 015121R DEF SCNT
C322 15150 127137R JMP READ1,I

```

C323**

C324* STORE INTERMEDIATE CODE AT VIRTUAL ADDRESS

C325**

```

C326 15151 000000 DSTOR NOP
C327 15152 077123R STB TMP SAVE A AND B
C328 15153 073127R STA TMP1 PROGRAM PARAMETERS
C329 15154 163151R LDA DSTOR,I
C330 15155 160000 LDA 0,I GET VIRTUAL ADDR
C331 15156 037151R ISZ DSTOR INCREMENT RETURN (P+2)
C332 15157 073133R STA TMP4 SAVE VIRTUAL ADDRESS
C333 15160 017213R JSB FNDPG FIND DISK PAGE CORRES. TO ADDR.
C334 15161 066001C LDB PGPT1 BOTH CORE PAGES SAME?
C335 15162 056002C CFB PGPT2
C336 15163 027171R JMP DBLWR YES, WRITE DATA TO BOTH
C337 15164 067127R LDB TMP1 NO, WRITE TO 1
C338 15165 174000 STB 0,I
C339 15166 067123R LDB TMP RESTORE (A) & (B)
C340 15167 063127R LDA TMP1
C341 15170 127151R JMP DSTOR,I
C342*
C343 15171 062001C DBLWR LDA PGPT1 WRITE SAME INFO TO BOTH
C344 15172 003004 CMA,INA PAGES
C345 15173 043133R ADA TMP4
C346 15174 042005C ADA PG1
C347 15175 067127R LDB TMP1 GET THE DATA
C348 15176 174000 STB 0,I STORE IN FIRST PAGE

```

FG 157

LN 0349 ADA BFRL

```

C349 15177 042000 ADA BFRL INCREMENT ADDR TO PAGE 2
C350 15200 174000 STB 0,I INCREMENT ADDR TO PAGE 2
C351 15201 067123R LDB TMP STORE IN SECOND PAGE
C352 15202 063127R LDA TMP1 RESTORE (A) & (B)
C353 15203 127151R JMP DSTOR,I

```

C354***

C355** READ INTERMEDIATE CODE VIRTUAL ADDRESS

C356***

```

C357 15204 000000 DREAD NOP
C358 15205 077123R STB TMP SAVE B
C359 15206 073133R STA TMP4 SAVE VIRTUAL ADDR

```

DISK BUFFERING ROUTINES

```

C360 15207 017213R JSB FNDPG      FIND DISK PAGE CORRES. TO ADDR.
C361 15210 160000   LDA 0,I      PICK UP DATA AT VIRTUAL ADDRESS
C362 15211 067123R  LDB IMP      RESTORE B
C363 15212 127204R  JMP DREAD,I

```

C364***

C365** FIND AND LOAD DISK PAGE CORRESPONDING TO

C366** VIRTUAL ADDRESS

C367***

```

C368 15213 000000 FNDPG NOP
C369 15214 063133R  LDA TMP4      V.A. < END OF IM CODE STORAGE?
C370 15215 003000   CMA
C371 15216 042010C  ADA PBPTR
C372 15217 002020   SSA
C373 15220 027251R  JMP FND2    NO, SET POINTERS FOR PAGE 2
C374 15221 027242R  JMP FND1    YES, SET POINTERS FOR PAGE 1
C375 15222 063133R FND3  LDA TMP4
C376 15223 003000   CMA      VIRTUAL ADDRESS >=
C377 15224 042000C  ADA PGPTR      BEGINNING OF CURRENT PAGE
C378 15225 002021   SSA,RSS
C379 15226 017260R  JSB SWAP      NO, GET PROPER PAGE
C380 15227 063133R  LDA TMP4      YES, VIRTUAL ADDRESS >
C381 15230 003000   CMA      END OF CURRENT PAGE?
C382 15231 042000C  ADA PGPTR

```

PG 157

LN 0383 ADA BFRL

```

C383 15232 042000   ADA BFRL
C384 15233 002020   SSA
C385 15234 017260R  JSB SWAP      YES, GET PROPER PAGE
C386 15235 062000C  LDA PGPTR      NO, COMPUTE
C387 15236 003004   CMA,INA
C388 15237 043133R  ADA TMP4      CORE (DISK) BUFFER ADDRESS
C389 15240 042004C  ADA BGNPG
C390 15241 127213R  JMP FNDPG,I
C391 15242 062001C FND1  LDA PGPT1
C392 15243 072000C  STA PGPTR
C393 15244 062005C  LDA PG1
C394 15245 072004C  STA BGNPG
C395 15246 062004R  LDA PT1
C396 15247 072003C  STA PGPT
C397 15250 027222R  JMP FND3
C398 15251 062002C FND2  LDA PGPT2
C399 15252 072000C  STA PGPTR
C400 15253 062006C  LDA PG2
C401 15254 072004C  STA BGNPG
C402 15255 062005R  LDA PT2
C403 15256 072003C  STA PGPT
C404 15257 027222R  JMP FND3

```

C405***

C406** FIND AND LOAD DISK PAGES

C407***

```

C408 15260 000000 SWAP NOP
C409 15261 002400   CLA      INITIALIZE PAGE POINTER
C410 15262 073134R SWP1  STA TMP5
C411 15263 003004   CMA,INA

```

DISK BUFFERING ROUTINES

```

C412 15264 043133R      ADA TMP4      VIRTUAL ADDR. > PAGE POINTER?
C413 15265 002020      SSA
C414 15266 027272R      JMP SWP2      NO, EXIT WITH ADDR
C415 15267 063134R      LDA TMP5      YES, INCREMENT PAGE POINTER

```

FG 158

```

LN 0416      ADA BFRL      BY BUFFER LENGTH
C416 15270 042000      ADA BFRL      BY BUFFER LENGTH
C417 15271 027262R      JMP SWP1      CHECK AGAIN

```

FG 159

```

LN 0418 SWP2 LDA BFRL      DECREMENT PAGE POINTER BY
C418 15272 062000 SWP2 LDA BFRL      DECREMENT PAGE POINTER BY
C419 15273 003004      CMA, INA      1 BUFFER LENGTH
C420 15274 043134R      ADA TMP5
C421 15275 073134R      STA TMP5      AND SAVE
C422 15276 062000C      LDA PGPTR      GET ADDRESS OF CURRENT CORE RES PAGE
C423 15277 017066R      JSB ACUN
C424 15300 015005X      JSB EXEC      RETURN TO DISK
C425 15301 015310R      DEF **7
C426 15302 000055      DEF .2
C427 15303 015136R      DEF CW1
C428 15304 100004C      DEF BGNPG, I

```

FG 159

```

LN 0429      DEF BFRL
C429 15305 000000      DEF BFRL
C430 15306 015122R      DEF TCNT
C431 15307 015121R      DEF SCNT
C432 15310 063134R      LDA TMP5      GET ADDRESS OF NEW CORE RES PAGE
C433 15311 172003C      STA PGPT, I      STORE IN POINTER
C434 15312 072000C      STA PGPTR
C435 15313 017066R      JSB ACUN
C436 15314 017137R      JSB READ1      FETCH PAGE FROM DISK
C437 15315 127260R      JMP SWAP, I
C438 15316 000000 SVOP1 BSS 2
C439 15320 000000 SVOP2 BSS 2
C440*****
C441*** SUBROUTINE DISK DOUBLE LOAD*****
C442*****
C443* PERFORM DLD ADR, I
C444*
C445* CALLING SEQUENCE: JSB DDLD
C446* DEF ADR
C447*
C448*****
C449*
C450 15322 000000 DOLD NOP
C451 15323 163322R      LDA DDLD, I      GET DISK ADR
C452 15324 160000      LDA A, I
C453 15325 012222R      AND INF      REMOVE V.A. BIT
C454 15326 073340R      STA VA      AND SAVE
C455 15327 037322R      ISZ DDLD      INCR RETURN (P+2)
C456 15330 017204R      JSB DREAD
C457 15331 073341R      STA SV.LO

```

DISK BUFFERING ROUTINES

```

0458 15332 037340R      ISZ VA              INCR DISK ADR
0459 15333 063340R      LDA VA
0460 15334 017204R      JSB DREAD
0461 15335 064000      LDB A
0462 15336 063341R      LDA SV.LO
0463 15337 127322R      JMP DDLD,I          RETURN
0464*
0465 15340 000000      VA BSS 1
0466 15341 000000      SV.LO BSS 1

```

```

0468*****
0469*****SUBROUTINE DISK DOUBLE STORE*****
0470*****
0471*
0472*      PERFORM      DST ADR,I
0473*
0474*      CALLING SEQUENCE:      JSB DDST
0475*                               DEF ADR
0476*
0477*****
0478*
0479 15342 000000      DDST      NOP
0480 15343 104400      DST SVOP1      SAVE OPERAND
0481 15344 015316R
0481 15345 163342R      LDA DDST,I      GET DISK ADR
0482 15346 160000      LDA A,I
0483 15347 012222R      AND INF
0484 15350 073340R      STA VA              AND SAVE
0485 15351 037342R      ISZ DDST      INCR RETURN (P+2)
0486 15352 063316R      LDA SVOP1
0487 15353 017151R      JSB DSTOR
0488 15354 015340R      DEF VA
0489 15355 037340R      ISZ VA
0490 15356 063317R      LDA SVOP1+1
0491 15357 017151R      JSB DSTOR
0492 15360 015340R      DEF VA
0493 15361 104200      DLD SVOP1      RESTORE (A) & (B)
0494 15362 015316R
0494 15363 127342R      JMP DDST,I
0495*****
0496***      SUBROUTINE FETCH PARAMETER      ***
0497*****
0498*
0499*      FETCHES PARAMETERS AT ADDRESSES
0500*      EITHER ON DISK OR IN CORE
0501*
0502*      CALLING SEQUENCE:      JSB F/PAR
0503*                               DEF ADR
0504*
0505*      WHERE ADR IS THE ADDRESS OF THE PARAMETER POINTER
0506*
0507*****

```

DISK BUFFERING ROUTINES

```

C508*
C509 15364 000000 F/PAR NOP
C510 15365 163364R LDA F/PAR,I
C511 15366 037364R ISZ F/PAR
C512 15367 160000 LDA A,I
C513 15370 070001 STA B GET PARAMETER ADDRESS AND SAVE
C514 15371 012222R AND INF
C515 15372 072367R STA P/ADR
C516 15373 006021 SSB,RSS
C517 15374 027400R JMP **4
C518 15375 017322R JSB DDLD YES, FETCH PAR FROM DISK
C519 15376 000367R DEF P/ADR
C520 15377 127364R JMP F/PAR,I
C521 15400 104200 DLD P/ADR,I FETCH PAR FROM CORE
      15401 100367R
C522 15402 127364R JMP F/PAR,I
C523 00367 P/ADR EQU MVIOC
C524* *****
C525* SUBROUTINE FLOATING ADD
C526* *****
C527*
C528* EXECUTE FAD ADR,I
C529*
C530* ADR= ADDRESS OF 2ND OPERAND. MAY BE
C531* IN CORE OR ON DISK
C532* CALLING SEQUENCE:
C533*
C534* JSB .FAD.
C535* DEF ADR
C536*
C537* IF BIT #15 OF ADR IS SET 2ND
C538* OPERAND IS ASSUMED TO BE ON DISK.
C539*
C540* *****
C541 15403 000000 .FAD. NOP
C542 15404 104400 DST SVOP1 SAVE OPERAND # 1
      15405 015316R
C543 15406 002400 CLA SELECT ADD OP.
C544 15407 017437R JSB ARITH GO DO IT
C545 15410 037403R ISZ .FAD. INCREMENT RETURN
C546 15411 127403R JMP .FAD.,I
C547*
C548* *****
C549* SUBROUTINE FLOATING SUBTRACT
C550* *****
C551*
C552* SEE FLOATING ADD FOR DESCRIPTION
C553*
C554* *****
C555*
C556 15412 000000 .FSB. NOP
C557 15413 104400 DST SVOP1 SAVE OPERAND # 1
      15414 015316R
C558 15415 060056 LDA .3 SELECT SUBTRACT OP.
C559 15416 017437R JSB ARITH GO DO IT

```

DISK BUFFERING ROUTINES

```

0560 15417 037412R      ISZ .FSB.      INCREMENT RETURN
0561 15420 127412R      JMP .FSB.,I
0562*
0563*      *****
0564*      SUBROUTINE FLOATING MULTIPLY
0565*      *****
0566*
0567*      SEE FLOATING ADD FOR DESCRIPTION
0568*
0569*      *****
0570*
0571 15421 000000 .FMP. NOP
0572 15422 104400      DST SVOP1      SAVE OPERAND # 1
0573 15423 015316R
0573 15424 060061      LDA .6      SELECT MULTIPLY OP.
0574 15425 017437R      JSB ARITH      GO DO IT
0575 15426 037421R      ISZ .FMP.      INCREMENT RETURN
0576 15427 127421R      JMP .FMP.,I
0577*
0578*      *****
0579*      SUBROUTINE FLOATING DIVIDE
0580*      *****
0581*
0582*      SEE FLOATING ADD FOR DESCRIPTION
0583*
0584*      *****
0585*
0586 15430 000000 .FDV. NOP
0587 15431 104400      DST SVOP1      SAVE OPERAND # 1
0588 15432 015316R
0588 15433 060064      LDA .9      SELECT DIVIDE OP.
0589 15434 017437R      JSB ARITH      GO DO IT
0590 15435 037430R      ISZ .FDV.      INCREMENT RETURN
0591 15436 127430R      JMP .FDV.,I
0592*
0593*      *****
0594*      GENERAL ARITHMETIC SUBROUTINE
0595*      *****
0596*
0597*      EXECUTES AN ARITHMETIC OPERATION BETWEEN
0598*      DATA IN THE A & B REGISTERS AND DATA STORED
0599*      IN CORE OR ON THE DISK. THE RESULT IS RETURNED
0600*      IN THE A & B REGISTERS.
0601*
0602*      *****
0603*
0604 15437 000000 ARITH NOP
0605 15440 043504R      ADA .TADR      COMPUTE ADDRESS OF ARITH OP.
0606 15441 073467R      STA UPMOD      AND SAVE IT
0607 15442 063437R      LDA ARITH      GET ADDRESS OF
0608 15443 040046      ADA M5          PARAMETER
0609 15444 160000      LDA A,I          ADDRESS
0610 15445 160000      LDA A,I
0611 15446 002021      SSA,RSS      MAT. FLAG SET?
0612 15447 027453R      JMP ARITHU      NO, PROCESS NORMALLY

```

DISK BUFFERING ROUTINES

```

C613 15450 012222R      AND INF      YES, 2ND OPERAND IS ON DISK
C614 15451 160000      LDA A,I      GET PARAMETER ADDRESS
C615 15452 027456R      JMP ARTH3    GO TO GET 2ND OPERAND
C616 15453 160000 ARTH0 LDA A,I      GET PARAMETER ADDRESS
C617 15454 002021      SSA,RSS      2ND OPERAND ON DISK
C618 15455 027460R      JMP ARTH1    NO, FETCH IT FROM CORE
C619 15456 017505R ARTH3 JSB FTHOP   YES, FETCH IT FROM DISK
C620 15457 027462R      JMP ARTH2
C621 15460 104200 ARTH1 DLD A,I      FETCH 2ND OPERAND FROM CORE
      15461 100000
C622 15462 104400 ARTH2 DST SVOP2    SAVE TEMPORARILY
      15463 015320R
C623 15464 104200      DLD SVOP1     GET 1ST OPERAND
      15465 015316R
C624 15466 127467R      JMP *+1,I    GO PERFORM OPERATION
C625 15467 000000 OPMOD NOP
C626 15470 016002X      FAD SVOP2
      15471 015320R
C627 15472 127437R      JMP ARITH,I
C628 15473 016003X      FSB SVOP2
      15474 015320R
C629 15475 127437R      JMP ARITH,I
C630 15476 016001X      FMP SVOP2
      15477 015320R
C631 15500 127437R      JMP ARITH,I
C632 15501 016004X      FDV SVOP2
      15502 015320R
C633 15503 127437R      JMP ARITH,I
C634 15504 015470R .TADR DEF OPMOD+1
C635**
C636***    FETCH OPERAND FROM DISK TO A AND B
C637**
C638**
C639***    ADDRESS OF OPERAND INITIALLY IN A REGISTER
C640**    PERFORMS FOLLOWING OPERATIONS
C641**      STA B
C642**      LDA B,I
C643**      INB
C644**      LDB B,I
C645**
C646**    ALL ADDRESS RELATE TO DISK BUFFER VIRTUAL ADDRESS
C647**
C648 15505 000000 FTHOP NOP
C649 15506 012222R      AND INF      REMOVE D.A. BIT
C650 15507 070001      STA B
C651 15510 017051R      JSB LDBI
C652 15511 073130R      STA IMP2
C653 15512 006004      INB
C654 15513 017051R      JSB LDBI
C655 15514 064000      LDB A
C656 15515 063130R      LDA IMP2
C657 15516 127505R      JMP FTHOP,I
C658**
C659**
C660 15517 000000 LDOP2 NOP

```

DISK BUFFERING ROUTINES

0661	15520	063152R	LDA BIN02
0662	15521	002020	SSA
0663	15522	027525R	JMP **3
0664	15523	160000	LDA A,1
0665	15524	127517R	JMP LDOP2,I
0666	15525	017204R	JSB DREAD
0667	15526	127517R	JMP LDOP2,I
0668			END ENTRY

PG 159

0007 ERRORS *TOTAL **RTE ASMB 92067-16011