

表用

0001 *****

0002 *
 0003 * NAKAMIZU TINY BASIC
 0004 * FOR 6800
 0005 * VER.6.0
 0006 *
 0007 * ASSEMBLED
 0008 * BY H. YAMASHITA
 0009 * 1978.10
 0010 *
 0011 *****

0012 *
 0013 * OPERATING SYSTEM
 0014 *

0015	341C	INI4 EQU	\$341C
0016	35E6	INEEE EQU	\$35E6
0017	3476	OUTEEE EQU	\$3476
0018	369B	BREAK EQU	\$369B
0019	3C06	NEG EQU	\$3C06
0020	3BF7	PICK EQU	\$3BF7
0021	3BD7	SET EQU	\$3BD7
0022	3BE8	RESET EQU	\$3BE8
0023	3BF0	REVERS EQU	\$3BF0
0024	3CB6	VRAMP EQU	\$3CB6
0025		*	
0026	A094	DISP EQU	\$A094
0027		*	

0028 *****

0029 *
 0030 * SYSTEM ADRS
 0031 *

0032	A047	STACK EQU	\$A047
0033	A080	IXSTCK EQU	\$A080
0034	00C0	VRAM EQU	\$C0
0035	E006	PIACRB EQU	\$E006
0036		*	
0037		*****	

I/O

GRAPHIC SUB

OPTION
 (正逆フリップルーチン)

I/O の DISPLAY ポインタ

スタック

VRAM 上位ADRS
 H6 KEY SCAN 用

0038		*			
0039		* BASIC WORK			
0040		*			
0041	0000	ORG		0	
0042	0000	UTOP	RMB	54	
0043	0036	BOP	RMB	2	
0044	0038	EOP	RMB	2	
0045	003A	MEMEND	RMB	2	
0046	003C	IXU	RMB	20	
0047	0050	WKA	RMB	2	
0048	0052	WKB	RMB	2	
0049	0054	WKC	RMB	2	
0050	0056	XS	RMB	2	
0051	0058	SPS	RMB	2	
0052	005A	MDS	RMB	2	
0053	005C	FS	EQU	*	
0054	005C	ADP	RMB	4	
0055	0060	RNDS	RMB	2	
0056	0062	FLOD	RMB	1	
0057	0063	FAUT	RMB	1	
0058	0064	LPCT	RMB	1	
0059	0065	CHCT	RMB	1	
0060	0066	WKUSE	RMB	2	
0061		*			
0062	0068		RMB	2	
0063		*			
0064	006A	LNB	RMB	2	
0065	006C	EOB	RMB	2	
0066	006E	EADRS	RMB	2	
0067	0070	XSP	RMB	2	
0068	0072	CSP	RMB	2	
0069	0074	FSP	RMB	2	
0070	0076	DP	RMB	2	
0071	0078	BFFR	RMB	72	
0072	00C0	EOBF	EQU	*	
0073	00C0		RMB	8	
0074	00C8	CSTCK	EQU	*	
0075	00C8		RMB	56	
0076	0100	FSTCK	EQU	*	
0077		*			
0078		*****			

変数エリア '0' ~ 'Z'
 システム変数 'C', 'F', 'J'

IX変数エリア %0 ~ %9

- 用 WORK AREA

MOD SAVE

SAVE FLAG
 + ADPS 設定 WORK
 乱数レジスタ
 LOAD FLAG
 AUTO FLAG
 LOOP COUNTER
 出力文字数レジスタ
 USING FLAG + WORK

行番号レジスタ

入力バッファ末 ADPS

TEXT 用 PROGRAM COUNTER

IX 用 STACK POINTER

算術 //

FOR NEXT //

READ DATA POINTER

入力バッファ } 算術 STACK

} FOR NEXT STACK

```

0079      *
0080      0100      ORG      $100
0081      *
0082 0100 7E 0000.  START  JMP      CSTART
0083 0103 7E 0000.  WSTRT  JMP      END
0084      *
0085 0106 0C60      BGNTXT FDB      $0C60
0086 0108 33FF      ENDTXT FDB      $33FF
0087      *
0088 010A 96 62      LINEIN LDA A  FLOD
0089 010C 26 00.      BNE      CSTIN
0090 010E 96 63      LDA A  FLOD+1
0091 0110 27 00.      BEQ      LN2
0092 0112 CE 006A      LDX      #LNB
0093 0115 5F          CLR B
0094 0116 86 0A      LDA A  #10
0095 0118 BD 0000.      JSR      ADBA
0096 0118 BD 0000.      JSR      PNUM
0097 011E BD 0000.      JSR      OUTS
0098 0121 20 00.      BRA      LN6
0099      *
0100 0123 8D 00.      LN1      BSR      CR1
0101 0125 86 3E      LN2      LDA A  #'>'
0111 13
0102 0127 8D 00.      BSR      COT
0103 0129 8C          FCB      $8C
0104 012A 8D 00.      LN3      BSR      CR2
0105 012C CE 0079      LN4      LDX      #BFFR+1
0106 012F 09          LN5      DEX
0107 0130 8C 0077      CPX      #BFFR-1
0108 0133 27 F5      BEQ      LN3
0109 0135 98 60      LN6      EOR A  RNDS
0122 12
0110 0137 97 60      STA A  RNDS
0111 0139 8D 00.      BSR      IN
0112 013B A7 00      STA A  0,X
0113 013D 81 0F      CMP A  #4F
0114 013F 27 EE      BEQ      LN5
0115 0141 81 0D      CMP A  #4D
0116 0143 26 00.      BNE      LN7
0117 0145 6F 00      CLR      0,X
0118 0147 DF 6C      STX      EOB
0119 0149 20 00.      BRA      CR2
0120      *

```

COLD START
WARM START

TEXT開始ADRS DATA
TEXT最終 // DATA

入力港区切り換え

自動動作番号処理

7000700 出力

(2バイト スキップ)

乱数初期値変化

RUB OUT ?

CR 処理

0121 014B 81 18	LN7	CMP A	##18
0144 06			
0122 014D 27 D4		BEQ	LN1
0123 014F 81 1F		CMP A	##1F
0124 0151 25 E2		BCS	LN6
0125 0153 08		INX	
0126 0154 8C 00C0		CPX	#E0BF
0127 0157 26 DC		BNE	LN6
0128 0159 C6 01	ERR1	LDA B	#1
0129 015B 7E 0000.		JMP	ERR
0130	*		
0131 015E BD 35E6	IN	JSR	INEEE
013A 23			
0132 0161 81 03		CMP A	#3
0133 0163 27 00.		BEQ	END
0134 0165 39		RTS	
0135	*		
0136 0166 8D F6	CSTIN	BSR	IN
010D 58			
0137 0168 81 02		CMP A	#2
0138 016A 26 FA		BNE	CSTIN
0139 016C 20 BE		BRA	LN4
0140	*		
0141 016E BD 0000.	CR1	JSR	CR
0124 49			
0142 0171 CE 0078	CR2	LDX	#BFFR
012B 45			
014A 26			
0143 0174 7F 0065		CLR	CHCT
0144 0177 39	RTN0	RTS	
0145	*		
0146 0178 DE 36	PTST	LDX	BOP
0147 017A 6F 00		CLR	0,X
0148 017C 08		INX	
0149 017D DF 38	DEL2	STX	EOP
0150 017F 86 80		LDA A	##80
0151 0181 A7 00		STA A	0,X
0152 0183 39		RTS	
0153	*		
0154 0184 36	BRK	PSH A	
0155 0185 BD 369B		JSR	BREAK
0156 0188 26 02		BNE	++4
0157 018A 8D D2		BSR	IN
0158 018C 32		PUL A	
0159 018D 39		RTS	
0160	*		

一行お消し?

CNT-C ?

TEXT 終りのマ-?

BREAK ?

0161	*		
0162 018E 8D E8	LOAD	BSR	PTST
0163 0190 8D 00.	APPEND	BSR	CNT11
0164 0192 97 62		STA A	FLOD
0165 0194 20 00.		BRA	EDIT
0166	*		
0167 0196 86 11	CNT11	LDA A	#11
0191 04			
0168 0198 7E 0000.	COT	JMP	OUT
0128 6F			
0169	*		
0170 019B 07	AUTO	TPA	
0171 019C 97 63		STA A	FAUT
0172 019E BD 0000.		JSR	CLCPL
0173 01A1 80 0A		SUB A	#10
0174 01A3 C2 00		SBC B	#0
0175 01A5 97 6B	SETLN	STA A	LNB+1
0176 01A7 D7 6A		STA B	LNB
0177 01A9 39		RTS	
0178	*		
0179 01AA BD 0000.	STOP	JSR	PRINT
0180 01AD BD 0000.		JSR	PSHX
0181 01B0 20 00.		BRA	END1
0182	*		
0183 01B2 BD 341C	CSTART	JSR	INI4
0101 0182			
0184 01B5 FE 0106		LDX	BGNTXT
0185 01B8 DF 36		STX	BOP
0186 01BA FE 0108		LDX	ENDTXT
0187 01BD DF 3A		STX	MEMEND
0188	*		
0189 01BF 8D B7	NEW	BSR	PTST
0190 01C1 8E A047	END	LDS	#STACK
0104 01C1			
0164 5C			
0191 01C4 BD 0000.		JSR	INITP
0192 01C7 CE 0000.	END1	LDX	#READY
01B1 15			
0193 01CA BD 0000.		JSR	MGOUT
0194 01CD BD 0000.		JSR	CR
0195 01D0 07		TPA	
0196 01D1 97 6E		STA A	EADRS
0197	*		

自動行番号

"STOP" 処理

COLD START
INITIALIZE

"NEW"
"END"

READY

ダイレクトモード 処理
(実行アドレスを負に33)

↓
ERR 217 処理

(マシンのエディット・モード
メイン・ルーチン)

0198	01D3	8E	A047	EDIT	LDS	#STACK
0199	01D6	BD	010A	ED1	JSR	LINEIN
0200	01D9	BD	0000.		JSR	PKUP
0201	01DC	27	F8		BEQ	ED1
0202	01DE	BD	0000.		JSR	ATOB
0203	01E1	24	00.		BCC	DRCT1
0204	01E3	8D	C0		BSR	SETLN
0205	01E5	8D	00.		BSR	ED2
0206	01E7	20	EA		BRA	EDIT
0207				*		
0208	01E9	DF	56	ED2	STX	XS
	01E6	02				
0209	01EB	9F	58		STS	SPS
0210	01ED	BD	0000.		JSR	SCHLIN
0211	01F0	DF	54		STX	WKC
0212	01F2	25	00.		BCS	INSERT
0213	01F4	BD	0000.		JSR	NXTL1
0214	01F7	08			INX	
0215	01F8	0F			SEI	
0216	01F9	35			TXS	
0217	01FA	DE	54		LDX	WKC
0218				*		
0219	01FC	09			DEX	
0220	01FD	20	00.		BRA	ED4
0221				*		
0222	01FF	08		ED3	INX	
0223	0200	33			PUL B	
0224	0201	E7	00		STA B	0,X
0225	0203	26	FA		BNE	ED3
0226	0205	08		ED4	INX	
	01FE	06				
0227	0206	33			PUL B	
0228	0207	E7	00		STA B	0,X
0229	0209	C1	00		CMF B	##80
0230	020B	26	F2		BNE	ED3
0231				*		
0232	020D	DF	38		STX	EOP
0233	020F	DE	56		INSERT LDX	XS
	01F3	1B				
0234	0211	A6	00		LDA A	0,X
0235	0213	27	00.		BEQ	INST4
0236	0215	96	6D		LDA A	EOB+1
0237	0217	90	57		SUB A	XS+1
0238	0219	8B	03		ADD A	#3

ASCII → BINARY
行番号なしの DIRECT MODE
行番号 SET
エディター - A

(エディター)

行番号の ADDR をささす。

は同じ行番号をささす。

同じ行番号をささす。

その行消去して
後の TEXT を前へ
つめる。

転送

挿入する 144 数計算

0239	021B	DE 38		LDX	EOP
0240	021D	9B 39		ADD A	EOP+1
0241	021F	97 39		STA A	EOP+1
0242	0221	24 03		BCC	*+5
0243	0223	7C 0038		INC	EOP
0244	0226	BD 0000.		JSR	SIZE
0245	0229	24 00.		BCC	INST1
0246	022B	DF 38		STX	EOP
0247	022D	20 00.		BRA	ERR2
0248			*		
0249	022F	20 00.	DRCT1	BRA	DIRCT
	01E2	4C			
0250			*		
0251	0231	9E 38	INST1	LDS	EOP
	022A	06			
0252	0233	08		INX	
0253	0234	09	INST2	DEX	
0254	0235	A6 00		LDA A	0,X
0255	0237	36		PSH A	
0256	0238	9C 54		CPX	WKC
0257	023A	26 F8		BNE	INST2
0258	023C	9E 6A		LDS	LNB
0259	023E	AF 00		STS	0,X
0260	0240	08		INX	
0261	0241	9E 56		LDS	XS
0262	0243	34		DES	
0263	0244	08	INST3	INX	
0264	0245	32		PUL A	
0265	0246	A7 00		STA A	0,X
0266	0248	26 FA		BNE	INST3
0267	024A	9E 58	INST4	LDS	SPS
	0214	35			
0268	024C	0E		CLI	
0269	024D	39		RTS	
0270			*		
0271	024E	CE 0000	INITP	LDX	#0
	01C5	024E			
0272	0251	DF 62		STX	FLOD
0273	0253	DF 63		STX	FRUT
0274	0255	DF 66		STX	WKUSE
0275	0257	CE 0100		LDX	#FSTCK
0276	025A	DF 74		STX	FSP
0277	025C	CE A000		LDX	#IXSTCK
0278	025F	DF 70		STX	XSP
0279	0261	CE 00C8		LDX	#CSTCK
0280	0264	DF 72		STX	CSP

140) 容量 9, 17

は OK

は NG

計算した新しい最終アドレス
IX は 古い ポインタ

挿入する分の
スペースを開ける

行番号 (BINARY)
転送

TEXT を メモリに
おける TEXT エリアへ
転送

FLAG & POINTER
INITIALIZE

2100 以降の AR


```

0281 0266 FE 0106      LDX    BGNTXT
0282 0269 DF 76        STX    DP
0283 026B 39           RTS
0284                    *
0285                    * DIRECT MODE
0286                    *
0287 026C BD 0000.     DIRECT JSR    CHVAR
0288 0230 3B
0288 026F 24 00.       BCC     RN3
0289 0271 CE 0000.     LDX     #COMMAND
0290 0274 20 00.       BRA     RN4
0291                    *
0292 0276 08           NXTL1 INX
0293 01F5 0276
0293 0277 08           NXTL2 INX
0294 0278 6D 00       REM     TST     0,X
0295 027A 26 FB       BNE     NXTL2
0296 027C 39           RTS
0297                    *
0298 027D DE 6E       SCHEND LDX     EADDRS
0299 027F 2A 00.       BPL     SL1
0300 0281 DE 36       SCHLIN LDX     BOP
0301 01EE 0281
0301 0283 08           INX
0302 0284 96 6B       SL1     LDA A   LNB+1
0303 0280 03
0303 0286 D6 6A       LDA B   LNB
0304 0288 2B 00.       BMI     ERR3
0305 028A 26 00.       BNE     SL3
0306 028C 4D           TST A
0307 028D 26 00.       BNE     SL3
0308 028F C6 03       ERR3    LDA B   #3
0309 0289 05
0309 0291 8C           FCB     #8C
0310 0292 C6 02       ERR2    LDA B   #2
0311 022E 63
0311 0294 7E 0000.     JMP     ERR
0312                    *
0313 0297 8D D0       SL2     BSR     NXTL1
0314 0299 08          RS1     INX
0315 029A BD 0000.     SL3     JSR     CPBA
0316 028B 0E
0316 028E 0B
0316 029D 22 F8       BHI     SL2
0317 029F 39          RTH1    RTS
0318                    *

```

(ダイレクトモード)

次の行をさがす。

(行番号を探す)

実行アドレスが異なる
前から探す。

行番号が見つかる
あるいは ERR

行番号 0 なる ERR

0319 02A0 BD 0000.	SUBIF	JSR	CLCPL
0320 02A3 5D		TST B	
0321 02A4 26 01		BNE	*+3
0322 02A6 4D		TST A	
0323 02A7 39		RTS	
0324	*		
0325 02A8 8D F6	IF	BSR	SUBIF
0326 02AA 27 CC		BEQ	REM
0327 02AC 20 00.		BRA	RN2
0328	*		
0329 02AE 96 3B	SIZE	LDA A	MEMEND+1
0227 02AE			
0330 02B0 D6 3A		LDA B	MEMEND
0331 02B2 90 39		SUB A	EOP+1
0332 02B4 D2 38		SBC B	EOP
0333 02B6 39		RTS	
0334	*		
0335 02B7 7E 01C7	END2	JMP	END1
0336	*		
0337 02BA 8D 92	RUN	BSR	INITP
0338 02BC DE 36		LDX	BOP
0339 02BE 08		INX	
0340 02BF DF 6E	RN1	STX	EADRS
0341 02C1 A6 00		LDA A	0,X
0342 02C3 48		ASL A	
0343 02C4 96 6E		LDA A	EADRS
0344 02C6 23 EF		BLS	END2
0345 02C8 08		INX	
0346 02C9 08		INX	
0347 02CA BD 0184	RN2	JSR	BRK
02AD 1C			
0348 02CD BD 0000.		JSR	CHVAR
0349 02D0 25 05		BCS	*+7
0350 02D2 BD 0000.	RN3	JSR	LET
0270 61			
0351 02D5 20 00.		BRA	RN5
0352	*		
0353 02D7 CE 0000.		LDX	#STATE
0354 02DA 8E A047	RN4	LDS	#STACK
0275 64			
0355 02DD BD 0000.		JSR	TBLJP
0356 02E0 BD 0000.	RN5	JSR	PKUP
02D6 09			
0357 02E3 26 FB		BNE	RN5
0358 02E5 08		INX	
0359 02E6 24 D7		BCC	RN1
0360 02E8 20 E0		BRA	RN2

IF. UNTIL の

条件判断

"IF"

残り 141 - 計算

[実行モード]

TEXT の終りまで行った
時又は DIRECT MODE
の時は終了。

DIRECT MODE
の EADRS の最上位ビット
を 1 にしている

変数メモリー

代入処理

ステータス処理

16進入力処理

```

0361 *
0362 02EA 49 HB1 ROL A
0363 02EB 48 ASL A
0364 02EC 48 ASL A
0365 02ED 48 ASL A
0366 02EE 48 ASL A
0367 02EF 79 0054 HB2 ROL WKC
0368 02F2 59 ROL B
0369 02F3 48 ASL A
0370 02F4 26 F9 BNE HB2
0371 02F6 08 HTOB INX
0372 02F7 8D 00. BSR TSTHEX
0373 02F9 25 EF BCS HB1
0374 02FB 96 54 LDA A WKC
0375 02FD 39 RTS
0376 *
0377 02FE BD 0000. TSTHEX JSR TSTN
      02F8 05
0378 0301 25 00. BCS RTN2
0379 0303 80 07 SUB A #7
0380 0305 81 3A CMP A #'9'+1
0381 0307 0C CLC
0382 0308 2D 00. BLT RTN2
0383 030A 81 40 CMP A #'0'
0384 030C 39 RTN2 RTS
      0302 09
      0309 02
0385 *
0386 030D E1 00 CPBA CMP B 0,X
      029B 030D
0387 030F 26 02 BNE *+4
0388 0311 A1 01 CMP A 1,X
0389 0313 39 RTS
0390 *
0391 0314 DF 56 TSF STX XS
0392 0316 CE 0000. LDX #FRNEXT
0393 0319 9F 58 TBLSCH STS SPS
0394 031B 0F SEI
0395 031C 9E 56 TS1 LDS XS
0396 031E 34 DES
0397 031F 08 INX
0398 0320 08 TS2 INX
0399 0321 32 PUL A
0400 0322 16 TAB
0401 0323 E0 00 SUB B 0,X
0402 0325 58 ASL B
0403 0326 27 00. BEQ TS3

```

16進入力処理

2バイト比較

行末 → Z=1 C=0
 文字 → Z=1 C=1
 行末 → Z=0 C=1

(TABLE SEARCH)
 SP ← TEXT
 IX ← TABLE

1文字単位

0404 0328 E6 00		LDA B 0,X
0405 032A 00		INX
0406 032B 2A FB		BPL *-3
0407 032D 81 2E		CMP A #'.'
0408 032F 27 00.		BEQ TS4
0409 0331 53		COM B
0410 0332 26 E8		BNE TS1
0411 0334 20 00.		BRA TS5
0412	*	
0413 0336 24 E8	TS3	BCC TS2
0327 0E		
0414 0338 08		INX
0415 0339 31	TS4	INS
0330 08		
0416 033A 9F 56		STS XS
0417 033C 9E 58	TS5	LDS SPS
0335 06		
0418 033E 0E		CLI
0419 033F 39		RTS
0420	*	
0421 0340 7F 0066	PRINT CLR	WKUSE
01AB 0340		
0422 0343 8D 00.		BSR PK1
0423 0345 27 00.		BEQ PR4
0424 0347 8D 00.	PR1	BSR PR7
0425 0349 27 00.		BEQ PR2
0426 034B DF 56		STX XS
0427 034D CE 0000.		LDX #FMT
0428 0350 BD 0000.		JSR TBLJP
0429 0353 8D 00.	PR2	BSR PK1
034A 08		
0430 0355 27 00.		BEQ PR4
0431 0357 81 3B		CMP A #' ; '
0432 0359 27 00.		BEQ PR3
0433 035B 81 2C		CMP A #' , '
0434 035D 26 00.		BNE ERR4
0435 035F 8D 00.		BSR PR6
0436 0361 08	PR3	INX
035A 06		
0437 0362 8D 00.		BSR PK1
0438 0364 26 E1		BNE PR1
0439 0366 39		RTS
0440	*	

省略形処理

TABLE 未定義

一致したものか
見つかった

USING FLAG CLR

1 BYTE PICKUP

注文

2つの ' or " あり

FORMATTER
TABLE JUMP

' ; ' あり

' , ' あり

```

0441 0367 7E 0000. PK1 JMP PKUP
0344 22
0354 12
0363 03
0442 *
0443 036A 7E 0000. PR4 JMP CR
0346 23
0356 13
0444 *
0445 036D 8D 0000. PR5 JSR OUTS
0446 0370 96 65 PR6 LDA A CHCT
0360 0F
0447 0372 85 07 BIT A #7
0448 0374 26 F7 BNE PR5
0449 0376 39 RTS
0450 *
0451 0377 8D 0000. CHR JSR CLCPL
0452 037A 7E 0000. AOT JMP OUT
0453 *
0454 037D 8D 0000. TAB1 JSR CLCPL
0455 0380 16 TAB
0456 0381 20 00. BRA TAB2
0457 *
0458 0383 8D 0000. JSR OUTS
0459 0386 D1 65 TAB2 CMP B CHCT
0382 03
0460 0388 22 F9 BHI *-5
0461 038A 39 RTS
0462 *
0463 038B 8D DA PR7 BSR PK1
0348 42
0464 038D 81 22 CMP A #' '
0465 038F 27 04 BEQ *+6
0466 0391 81 27 CMP A #' ' '
0467 0393 26 00. BNE RTN3
0468 0395 16 TAB
0469 0396 08 INX
0470 0397 20 02 BRA *+4
0471 *
0472 0399 8D DF PR8 BSR AOT
0473 039B A6 00 LDA A 0,X
0474 039D 27 00. BEQ ERR4
0475 039F 08 INX
0476 03A0 11 CBA
0477 03A1 26 F6 BNE PR8
0478 03A3 39 RTN3 RTS
0394 0E

```

' ' 処理

CHR4 処理

TAB 処理

リテラル・ストリング
処理

0479		*		
0480	03A4 C6 04	ERR4	LDA B	#4
	035E 45			
	039E 05			
0481	03A6 8C		FCB	\$8C
0482	03A7 C6 05	ERR5	LDA B	#5
0483	03A9 7E 0000.		JMP	ERR
0484		*		
0485	03AC A6 00	DIV	LDA A	0,X
0486	03AE AA 01		ORA A	1,X
0487	03B0 27 F5		BEQ	ERR5
0488	03B2 8D 00.		BSR	SETCT
0489	03B4 8D 00.		BSR	MDV3
0490	03B6 49		ROL A	
0491	03B7 40		NEG A	
0492	03B8 16		TAB	
0493	03B9 8D 00.		BSR	MDV4
0494	03BB 49		ROL A	
0495	03BC 88 01		EOR A	#1
0496	03BE 46		ROR A	
0497	03BF 8D 00.		BSR	MDV2
0498	03C1 97 5B		STA A	MD5+1
0499	03C3 D7 5A		STA B	MD5
0500	03C5 BD 0300		JSR	CPBA
0501	03C8 26 00.		BNE	DV1
0502	03CA 4F		CLR A	
0503	03CB 16		TAB	
0504	03CC 6C 03		INC	3,X
0505	03CE 26 00.		BNE	DV1
0506	03D0 6C 02		INC	2,X
0507	03D2 08	DV1	INX	
	03C9 08			
	03CF 02			
0508	03D3 08		INX	
0509	03D4 39		RTS	
0510		*		
0511	03D5 8D 00.	MDIV	BSR	SETCT
0512	03D7 8D 00.	MDV1	BSR	MDV3
0513	03D9 49		ROL A	
0514	03DA 59		ROL B	
0515	03DB A0 01		SUB A	1,X
0516	03DD E2 00		SBC B	0,X
0517	03DF 8D 00.		BSR	MDV4
0518	03E1 7A 0064	MDV2	DEC	LPCT
	03C0 20			
0519	03E4 26 F1		BNE	MDV1

(除算)

除数=0にJ ERR

C737
反転

余り SAVE

商

M(2,X) M(3,X)

(商)リザルタン-9-
M(12) ← (IV+2)/CD

$$\begin{array}{c} \leftarrow \\ \text{D} \quad \text{A} \quad \text{M}(2,X) \quad \text{M}(3,X) \\ \text{M}(0,X) \quad \text{M}(1,X) \quad (\text{被除数}) \\ (\text{除数}) \end{array}$$

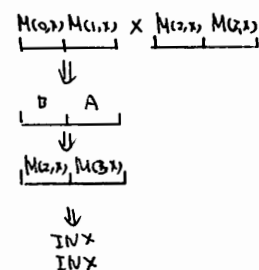
↓

$$\begin{array}{c} \text{B} \quad \text{A} \quad \text{余り} \\ \text{---} \end{array}$$

0520	03E6 69 03	MDU3	ROL	3,X
	03B5 30			
	03D8 00			
0521	03E8 69 02		ROL	2,X
0522	03EA 39		RTS	
0523		*		
0524	03EB 37	MDU4	PSH B	
	03BA 30			
	03E0 0A			
0525	03EC E8 00		EOR B	0,X
0526	03EE 33		PUL B	
0527	03EF 00		SEC	
0528	03F0 2A 00.		BPL	RTN4
0529	03F2 AB 01	ADD	ADD A	1,X
0530	03F4 E9 00		ADC B	0,X
0531	03F6 0C		CLC	
0532	03F7 39	RTN4	RTS	
	03F1 05			
0533		*		
0534	03F8 86 10	SETCT	LDA A	#16
	03B3 44			
	03D6 21			
0535	03FA 97 64		STA A	LPCT
0536	03FC 4F		CLR A	
0537	03FD 5F		CLR B	
0538	03FE 39		RTS	
0539		*		
0540	03FF 8D F7	MULTI	BSR	SETCT
0541	0401 48	MLT1	ASL A	
0542	0402 59		ROL B	
0543	0403 69 01		ROL	1,X
0544	0405 69 00		ROL	0,X
0545	0407 24 04		BCC	*+6
0546	0409 AB 03		ADD A	3,X
0547	040B E9 02		ADC B	2,X
0548	040D 7A 0064		DEC	LPCT
0549	0410 26 EF		BNE	MLT1
0550	0412 08		INX	
0551	0413 08		INX	
0552	0414 7E 0000.		JMP	STBA
0553		*		

(内部物理)

(乗算)



0554 0417 EE 00	PNUM	LDX	0,X
011C 0417			
0555 0419 DF 52		STX	WKB
0556 041B CE 0052		LDX	#WKB
0557 041E 8D 00.		BSR	ABSX
0558 0420 25 00.		BCS	PU1
0559 0422 D6 66		LDA B	WKUSE
0560 0424 27 00.		BEQ	PU2
0561 0426 17		TBA	
0562 0427 8C		FCB	\$8C
0563 0428 86 2D	PU1	LDA A	#'-'
0421 06			
0564 042A 8D 00.		BSR	OUT
0565 042C 09	PU2	DEX	
0425 06			
0566 042D 09		DEX	
0567 042E C6 0A		LDA B	#10
0568 0430 E7 01		STA B	1,X
0569 0432 6F 00		CLR	0,X
0570 0434 57		ASR B	
0571 0435 37	PN1	PSH B	
0572 0436 8D 9D		BSR	NDIV
0573 0438 33		PUL B	
0574 0439 36		PSH A	
0575 043A 5A		DEC B	
0576 043B 26 F8		BNE	PN1
0577 043D CE 0078		LDX	#BFFR
0578 0440 C6 04		LDA B	#4
0579 0442 96 64	PN2	LDA A	LPCT
0580 0444 32		PUL A	
0581 0445 26 00.		BNE	PN3
0582 0447 97 64		STA A	LPCT
0583 0449 26 00.		BNE	PN3
0584 044B 7D 0066		TST	WKUSE
0585 044E 27 00.		BEQ	PN4
0586 0450 96 67		LDA A	WKUSE+1
0587 0452 80 30		SUB A	#'0'
0588 0454 8D 00.	PN3	BSR	PN5
0446 0D			
044A 09			
0589 0456 5A	PN4	DEC B	
044F 06			
0590 0457 26 E9		BNE	PN2

(数字出力)

数(BINARY)
を転送

絶対値
は負数に3.

USING 処理

'-'出力

除数 10 SET

LOOPCOUNTER 5

5回
107-1413.

① 余り PUSH

LPCT 最初 0

② 余り PUL

0 カラシ

USING 処理

ASCII 変換に
出力

0591 0459 32		PUL A	
0592 045A 8B 30	PN5	ADD A	#'0'
0455 04			
0593 045C 7D 0063		TST	FAUT
0594 045F 27 00.		BEQ	OUT
0595 0461 A7 00		STA A	0,X
0596 0463 08		INX	
0597 0464 7C 0065	OUT	INC	CHCT
0199 0464			
037B 0464			
042B 38			
0460 03			
0598 0467 BD 0184	OUTC	JSR	BRK
0599 046A 7E 3476		JMP	OUTEEE
0600	*		
0601 046D 6D 00	ABSX	TST	0,X
041F 4D			
0602 046F 2A 00.		BPL	RTN5
0603 0471 60 01	NEGX	NEG	1,X
0604 0473 26 02		BNE	*+4
0605 0475 6A 00		DEC	0,X
0606 0477 63 00		COM	0,X
0607 0479 39	RTN5	RTS	
0470 08			
0608	*		
0609 047A 81 40	CHASC	CMP A	#'E'
0610 047C 25 00.		BCS	CHV1
0611 047E 81 5B		CMP A	#'Z'+1
0612 0480 39		RTS	
0613	*		
0614 0481 0C	CHV1	CLC	
047D 03			
0615 0482 39		RTS	
0616	*		
0617 0483 8D 00.	CHVAR	BSR	PKUP
026D 0483			
02CE 0483			
0618 0485 DF 56		STX	XS
0619 0487 81 21		CMP A	#'!'
0620 0489 27 00.		BEQ	TBL
0621 048B 81 2E		CMP A	#'.'
0622 048D 27 00.		BEQ	TBL
0623 048F 81 2A		CMP A	#'*
0624 0491 27 00.		BEQ	TBL
0625 0493 8D E5		BSR	CHASC
0626 0495 24 EA		BCC	CHV1

0 全PUL 最後の桁は
0 カラシない

ASCII 変換
に出カ

← 符号チェンク (7 桁 8 桁)
E4D (正 C=0
後 C=1)
← 符号反転

ASCII CHECK
'0' ~ '9' → C=1
他 → C=0

変数チェンク
TABLE へ行く必要の
あるもの C=1
ないもの C=0

0627	0497	A6 01	LDA	A	1,X
0628	0499	81 2E	CMP	A	#'-'
0629	049B	27 00.	BEQ		TBL
0630	049D	8D DB	BSR		CHASC
0631	049F	24 E0	BCC		CHV1
0632	04A1	39	RTS		
0633			*		
0634	04A2	08	ICPKUP	INX	
0635	04A3	A6 00	PKUP	LDA	A 0,X
	01DA	04A3			
	02E1	04A3			
	0368	04A3			
	0484	1E			
0636	04A5	81 20	CMP	A	#'-'
0637	04A7	27 F9	BEQ		ICPKUP
0638	04A9	4D	TST	A	
0639	04AA	27 00.	BEQ		RTN6
0640	04AC	81 3A	CMP	A	#'-'
0641	04AE	0D	TBL	SEC	
	048A	23			
	048E	1F			
	0492	1B			
	049C	11			
0642	04AF	39	RTN6	RTS	
	04AB	03			
0643			*		
0644	04B0	8D F1	EX1	BSR	PKUP
0645	04B2	81 2D	CMP	A	#'-'
0646	04B4	26 00.	BNE		EX2
0647	04B6	8D 00.	BSR		EX7
0648	04B8	20 00.	BRA		EX4
0649	04BA	81 2B	EX2	CMP	A #'+'
	04B5	04			
0650	04BC	27 00.	BEQ		EX3
0651	04BE	09	DEX		
0652	04BF	8D 00.	EX3	BSR	TH1
	04BD	01			
0653	04C1	DE 56	EX4	LDX	XS
	04B9	07			
0654	04C3	8D DE	BSR		PKUP
0655	04C5	81 2B	CMP	A	#'+'
0656	04C7	26 00.	BNE		EX6
0657	04C9	8D 00.	BSR		TH1
0658	04CB	8D 00.	EX5	BSR	EX8
0659	04CD	20 F2	BRA		EX4
0660			*		

(PICK UP)

スノ-ス 読みとり

区切り 4x7

区切り 100 Z=1 C=0

区切り 11 Z=1 C=1

他 Z=0 C=1

[第2レベル演算]

マシナ処理

マシナ処理

加算

0661 04CF 81 2D	EX6	CMP A	#'-'
04C8 06			
0662 04D1 26 00.		BNE	RTN7
0663 04D3 8D 00.		BSR	EX7
0664 04D5 20 F4		BRA	EX5
0665	*		
0666 04D7 8D 00.	EX7	BSR	TH1
04B7 1F			
04D4 02			
0667 04D9 20 96		BRA	NEGX
0668	*		
0669 04DB BD 0000.	EX8	JSR	CPUL
04CC 0E			
0670 04DE 08		INX	
0671 04DF 08		INX	
0672 04E0 AB 01	ADBA	ADD A	1,X
0119 04E0			
0673 04E2 E9 00		ADC B	0,X
0674 04E4 A7 01	STBA	STA A	1,X
0415 04E4			
0675 04E6 E7 00		STA B	0,X
0676 04E8 39	RTN7	RTS	
04D2 15			
0677	*		
0678 04E9 BD 0000.	ABS	JSR	CUL1
0679 04EC 8D 00.		BSR	TH2
0680 04EE BD 046D		JSR	ABSX
0681 04F1 DE 56		LDX	XS
0682 04F3 39		RTS	
0683	*		
0684 04F4 CA 01	EX11	ORA B	#1
0685 04F6 8C		FCB	\$8C
0686 04F7 CA 02	EX12	ORA B	#2
0687 04F9 8C		FCB	\$8C
0688 04FA CA 04	EX13	ORA B	#4
0689 04FC 08		INX	
0690 04FD C1		FCB	\$C1
0691 04FE 5F	EX14	CLR B	
0692 04FF 8D A2		BSR	PKUP
0693 0501 81 3E		CMP A	#'>'
0694 0503 27 EF		BEQ	EX11
0695 0505 81 3D		CMP A	#'='
0696 0507 27 EE		BEQ	EX12
0697 0509 81 3C		CMP A	#'<'
0698 050B 27 ED		BEQ	EX13
0699 050D 4F		CLR A	
0700 050E 39		RTS	

減算

常時24h内の
加算 サブルーチン

"ABS"

比較論理演算
4h7)

結果 ACCB

> 0000 0001
= 0000 0010
< 0000 0100
>= 0000 0011
<= 0000 0100
<> 0000 0101

0701		*		
0702	050F 8D 00.	TH1	BSR	TERM
	04C0 4E			
	04CA 44			
	04D8 36			
0703	0511 DF 56	TH2	STX	XS
	04ED 23			
0704	0513 DE 72		LDX	CSP
0705	0515 39	RTN8	RTS	
0706		*		
0707	0516 8D 98	EXPR	BSR	EX1
0708	0518 8D E4		BSR	EX14
0709	051A 5D		TST B	
0710	051B 27 F8		BEQ	RTN8
0711	051D 37		PSH B	
0712	051E 8D 90		BSR	EX1
0713	0520 BD 0000.		JSR	CPUL
0714	0523 DE 72		LDX	CSP
0715	0525 A0 01		SUB A	1,X
0716	0527 E2 00		SBC B	0,X
0717	0529 33		PUL B	
0718	052A 2D 00.		BLT	LT GT 上位
0719	052C 2E 00.		BGT	GT LT
0720	052E 4D		TST A	
0721	052F 27 00.		BEQ	EQ
0722	0531 57	GT	ASR B	
	052D 03			
0723	0532 57	EQ	ASR B	
	0530 01			
0724	0533 C4 01	LT	AND B	#1
	052B 07			
0725	0535 E7 01		STA B	1,X
0726	0537 6F 00		CLR	0,X
0727	0539 DE 54		LDX	WKC
0728	053B 39		RTS	
0729		*		
0730	053C 8D 00.	ATOB	BSR	TSTN
	01DF 053C			
0731	053E 24 00.		BCC	RTN9
0732	0540 5F		CLR B	
0733	0541 D7 55		STA B	WKC+1
0734	0543 D7 54	ATB1	STA B	WKC
0735	0545 84 0F		AND A	#\$F
0736	0547 97 64		STA A	LPCT
0737	0549 96 55		LDA A	WKC+1
0738	054B 48		ASL A	
0739	054C 59		ROL B	

(第3レベル演算)

諸理演算42.7

演算 (再帰的)

DAは後で

比較

後-前

源

諸理演算

(ASCII → BINARY)

10842.7

0740 054D 48	ASL A
0741 054E 59	ROL B
0742 054F 9B 55	ADD A WKC+1
0743 0551 D9 54	ADC B WKC
0744 0553 48	ASL A
0745 0554 59	ROL B
0746 0555 9B 64	ADD A LPCT
0747 0557 97 55	STA A WKC+1
0748 0559 C9 00	ADC B #0
0749 055B 08	INX
0750 055C 8D 00.	BSR TSTN
0751 055E 25 E3	BCS ATB1
0752 0560 96 55	LDA A WKC+1
0753 0562 0D	SEC
0754 0563 39 RTN9	RTS
053F 23	
0755 *	
0756 0564 BD 04A3 TSTN JSR PKUP	
02FF 0564	
053D 26	
055D 06	
0757 0567 81 30	CMP A #'0'
0758 0569 0C	CLC
0759 056A 2D 02	BLT *+4
0760 056C 81 3A	CMP A #'0'
0761 056E 39	RTS
0762 *	
0763 056F 8D 00. TERN BSR FUNCT	
0510 5E	
0764 0571 BD 04A3 TR1 JSR PKUP	
0765 0574 81 2A	CMP A #'*'
0766 0576 26 00.	BNE TR3
0767 0578 8D 00.	BSR TR4
0768 057A BD 03FF JSR MULTI	
0769 057D DF 72 TR2 STX CSP	
0770 057F DE 56 LDX XS	
0771 0581 20 EE BRA TR1	
0772 0583 81 2F TR3 CMP A #'/'	
0577 0B	
0773 0585 26 DC BNE RTN9	
0774 0587 8D 00. BSR TR4	
0775 0589 BD 03AC JSR DIV	
0776 058C 20 EF BRA TR2	
0777 *	
0778 058E 8D 00. TR4 BSR FUNCT	
0579 14	
0588 05	
0779 0590 7E 0511 JMP TR2	

数値のつづて限り
BINARY変換

紙線 B A

(数値は C=1)
他 C=0

(10進チェック)

(10進 C=1)
他 C=0

[第1レベル演算]

演算処理

除算処理

式の演算 (両側向)

0781 0593 5F	INHEX	CLR B
0782 0594 D7 54	STA B	WKC
0783 0596 BD 02F6	JSR	HTOB
0784 0599 20 00.	BRA	CPS2
0785	*	
0786 059B 8D 00.	PEEK	BSR FUNCT
0787 059D BD 0000.	JSR	CPUL
0788 05A0 8D 00.	BSR	TBAX
0789 05A2 A6 00	LDA A	0,X
0790 05A4 5F	CLR B	
0791 05A5 20 00.	BRA	CPS1
0792	*	
0793 05A7 A1 02	LITRL	CMP A 2,X
0794 05A9 26 00.	BNE	ERR7
0795 05AB A6 01	LDA A	1,X
0796 05AD 5F	CLR B	
0797 05AE 08	INX	
0798 05AF 08	INX	
0799 05B0 08	INX	
0800 05B1 20 00.	BRA	CPS2
0801	*	
0802 05B3 08	FUNCT	INX
0570 42		
058F 23		
059C 16		
0803 05B4 8D 86	BSR	ATOB
0804 05B6 25 00.	BCS	CPS2
0805 05B8 81 23	CMP A	#'8'
0806 05BA 27 DF	BEQ	PEEK
0807 05BC 81 22	CMP A	#'0'
0808 05BE 27 E7	BEQ	LITRL
0809 05C0 81 27	CMP A	#''''
0810 05C2 27 E3	BEQ	LITRL
0811 05C4 81 24	CMP A	#'5'
0812 05C6 27 CB	BEQ	INHEX
0813 05C8 BD 0483	JSR	CHVAR
0814 05CB 24 00.	BCC	CUL
0815 05CD CE 0000.	LDX	#FUNC
0816	*	
0817 05D0 BD 0319	TBLJP	JSR TBLSCH
02DE 05D0		
0351 05D0		
0818 05D3 A6 01	LDA A	1,X
0819 05D5 36	PSH A	
0820 05D6 A6 00	LDA A	0,X
0821 05D8 36	PSH A	
0822 05D9 DE 56	LDX	XS
0823 05DB 39	RTS	

'#' 16進値

BINARY変換

格納する PSH

'#' PEEK

31進 ADDR 計算

EA → IX

格納する PSH

リテラル 定数

(ASCII 定数) 処理

(関数 TABLE JUMP)

ASCII → BINARY

計算する

Peek

リテラル

16進値

変数 9,10

変数 (2項計算)

以外

関数 TABLE への JUMP

ADDR スタックに PSH

RTS で JUMP

0824	*		
0825 05DC 8D 00.	CPUSH	BSR	TBAX
0826 05DE A6 01		LDA A	1,X
0827 05E0 E6 00		LDA B	0,X
0828 05E2 DE 56	CPS1	LDX	XS
05A6 3B			
0829 05E4 DF 54	CPS2	STX	WKC
059A 49			
05B2 31			
05B7 2C			
0830 05E6 DE 72		LDX	CSP
0831 05E8 09		DEX	
0832 05E9 09		DEX	
0833 05EA 9C 6C		CPX	EOB
0834 05EC 2F 00.		BLE	ERR6
0835 05EE BD 04E4	CPS3	JSR	STBA
0836 05F1 DF 72		STX	CSP
0837 05F3 DE 54	CPS4	LDX	WKC
0838 05F5 39		RTS	
0839	*		
0840 05F6 C6 06	ERR6	LDA B	#6
05ED 08			
0841 05F8 8C		FCB	\$8C
0842 05F9 C6 07	ERR7	LDA B	#7
05AA 4E			
0843 05FB 8C		FCB	\$8C
0844 05FC C6 08	ERR8	LDA B	#8
0845 05FE 7E 0000.		JMP	ERR
0846	*		
0847 0601 8D 00.	CUL	BSR	VAR
05CC 34			
0848 0603 24 D7		BCC	CPUSH
0849 0605 A6 00	CUL1	LDA A	0,X
04EA 0605			
0850 0607 81 28		CMP A	#('
0851 0609 26 EE		BNE	ERR7
0852 060B 08	CUL2	INX	
0853 060C BD 0516	CUL3	JSR	EXPR
0854 060F A6 00	CUL4	LDA A	0,X
0855 0611 81 29		CMP A	#')'
0856 0613 26 E4		BNE	ERR7
0857 0615 08		INX	
0858 0616 39		RTS	
0859	*		

(計算機システム)

BA ← (BA)

計算機システム

変数, 変数. INDEX
の ADDRESS

は '()' の

'(' ~ ')' の
処理

変数 (両側の)

0860	0617	DF 56	TBAX	STX	XS
	05A1	75			
	05DD	39			
0861	0619	97 55		STA A	WKC+1
0862	061B	D7 54		STA B	WKC
0863	061D	20 D4		BRA	CPS4
0864			*		
0865	061F	BD 0000.	MOD	JSR	SUB5
0866	0622	27 00.		BEQ	MOD1
0867	0624	D6 5A		LDA B	MDS
0868	0626	96 5B		LDA A	MDS+1
0869	0628	20 BA		BRA	CPS2
0870			*		
0871	062A	08	MOD1	INX	
	0623	06			
0872	062B	BD 0516		JSR	EXPR
0873	062E	8D 00.		BSR	COMMA
0874	0630	26 CA		BNE	ERR8
0875	0632	08		INX	
0876	0633	8D D7		BSR	CUL3
0877	0635	DF 54	MOD2	STX	WKC
0878	0637	DE 72		LDX	CSP
0879	0639	BD 03AC		JSR	DIV
0880	063C	20 B0		BRA	CPS3
0881			*		
0882	063E	A6 00	COMMA	LDA A	0,X
	062F	0E			
0883	0640	81 2C		CHP A	#','
0884	0642	39		RTS	
0885			*		
0886	0643	D6 61	RND	LDA B	RNDS+1
0887	0645	96 60		LDA A	RNDS
0888	0647	1B		ABA	
0889	0648	C9 95		ADC B	#495
0890	064A	89 AB		ADC A	#4AB
0891	064C	97 61		STA A	RNDS+1
0892	064E	D7 60		STA B	RNDS
0893	0650	8D 92		BSR	CPS2
0894	0652	8D B1		BSR	CUL1
0895	0654	20 DF		BRA	MOD2
0896			*		
0897	0656	BD 0000.	USER	JSR	SUB10
0898	0659	8D 00.		BSR	US2
0899	065B	20 85		BRA	CPS1
0900			*		

IX ← BA

"MOD"

引数なしの時

引数付の時

2つの引数も'算術スタック'にPSH

割算して余りを得る

"RND"

引数なしの時

算術スタック PSH

引数の計算

割算して余りを求め残りを返す

"USER"

← 機械語が5555555555555555
RT TURN

MOD (A,B)

1. A
2. B

0901	065D 8D 00.	US2	BSR	CCP1	
	065A 02				
0902	065F 36		PSH	A	
0903	0660 37		PSH	B	
0904	0661 8D 00.		BSR	US3	
0905	0663 36		PSH	A	
0906	0664 37		PSH	B	
0907	0665 8D 00.		BSR	US3	
0908	0667 36		PSH	A	
0909	0668 37		PSH	B	
0910	0669 07		TPA		
0911	066A 36		PSH	A	
0912	066B 8D A2		BSR	CUL4	
0913	066D DF 56		STX	XS	
0914	066F 3B		RTI		
0915		*			
0916	0670 8D CC	US3	BSR	COMMA	
	0662 0D				
	0666 09				
0917	0672 26 00.		BNE	US4	
0918	0674 08		INX		
0919	0675 7E 0000.	CCP1	JMP	CLCPL	
	065E 16				
0920		*			
0921	0678 4F	US4	CLR	A	
	0673 04				
0922	0679 5F		CLR	B	
0923	067A 0D	NOTV	SEC		
0924	067B 39		RTS		
0925		*			
0926	067C 8D 04A3	VAR	JSR	PKUP	
	0602 79				
0927	067F 81 25		CMP	A #'Z'	
0928	0681 27 00.		BEQ	ARR	
0929	0683 81 40		CMP	A #'E'	
0930	0685 25 F3		BCS	NOTV	
0931	0687 81 5D		CMP	A #'J'	
0932	0689 22 EF		BHI	NOTV	
0933	068B 84 3F		AND	A ##3F	
0934	068D 48		ASL	A	
0935	068E 5F		CLR	B	
0936	068F 08		INX		
0937	0690 39		RTS		
0938		*			

第131回

第231回

第331回

引数がない時 0にち

(変数, IX変数, IX変数)
ADRS 計算

IX 変数 or IX 変数

単純変数 9, 2, 7

変数 ADRS

10進4,7.

0939 0691 08	ARR	INX	
0682 0E			
0940 0692 BD 0564		JSR	TSTN
0941 0695 24 00.		BCC	ERR14
0942 0697 08		INX	
0943 0698 BD 0000.		JSR	SUB5
0944 069B 27 00.		BEQ	IND
0945 069D 84 0F	DIR	AND A	#\$F
0946 069F 48		ASL A	
0947 06A0 8B 3C		ADD A	#IXV
0948 06A2 5F		CLR B	
0949 06A3 39		RTS	
0950	*		
0951 06A4 36	IND	PSH A	
069C 07			
0952 06A5 BD 060B		JSR	CUL2
0953 06A8 32		PUL A	
0954 06A9 8D F2		BSR	DIR
0955 06AB BD 0617		JSR	TBAX
0956 06AE BD 0000.		JSR	CPUL1
0957 06B1 48		ASL A	
0958 06B2 59		ROL B	
0959 06B3 BD 03F2		JSR	ADD
0960 06B6 DE 56		LDX	XS
0961 06B8 39		RTS	
0962	*		
0963 06B9 BD 067C	LET3	JSR	VAR
0964 06BC 25 00.		BCS	ERR9
0965 06BE 97 53	LET4	STA A	WKB+1
0966 06C0 D7 52		STA B	WKB
0967 06C2 A6 00		LDA A	0,X
0968 06C4 81 3D		CMP A	#'='
0969 06C6 26 00.		BNE	ERR9
0970 06C8 08		INX	
0971 06C9 8D 00.		BSR	CLCPL
0972 06CB DF 56		STX	XS
0973 06CD DE 52		LDX	WKB
0974 06CF 39		RTS	
0975	*		
0976 06D0 C6 0E	ERR14	LDA B	#14
0696 39			
0977 06D2 8C		FCB	\$8C
0978 06D3 C6 09	ERR9	LDA B	#9
06BD 15			
06C7 0B			
0979 06D5 8C		FCB	\$8C

IX 変数

ADRS 計算

* 変数名スタック
PSH

変数名 (変数名)

*
ADRS 計算変数名スタック
の PUL

ADRS 計算

(入変処理)

左側 ADRS
SAVE右側計算
(変数名)

```

0980 06D6 C6 0A ERR10 LDA B #10
0981 06D8 7E 0000. JMP ERR
0982 *
0983 06DB 7F 005F THEN CLR ADP+3
0984 06DE BD 053C JSR AT0B
0985 06E1 25 00. BCS G01
0986 06E3 7E 02CA JMP RN2
0987 *
0988 06E6 BD 05B3 POKE JSR FUNCT
0989 06E9 8D 00. BSR CPUL
0990 06EB 8D D1 BSR LET4
0991 06ED 16 TAB
0992 06EE 20 00. BRA LET2
0993 *
0994 06F0 9F 58 PSHX STS SPS
      01AE 06F0
0995 06F2 0F SEI
0996 06F3 35 TXS
0997 06F4 DE 70 LDX XSP
0998 06F6 8C A04A CPX #STACK+3
0999 06F9 27 DB BEQ ERR10
1000 06FB 09 DEX
1001 06FC 09 DEX
1002 06FD AF 00 STS 0,X
1003 06FF DF 70 PX1 STX XSP
1004 0701 30 TSX
1005 0702 9E 58 LDS SPS
1006 0704 0E CLI
1007 0705 39 RTS
1008 *
1009 0706 DE 70 PULX LDX XSP
1010 0708 8C A080 CPX #IXSTCK
1011 070B 27 C9 BEQ ERR10
1012 070D 9F 58 STS SPS
1013 070F 0F SEI
1014 0710 AE 00 LDS 0,X
1015 0712 08 INX
1016 0713 08 INX
1017 0714 20 E9 BRA PX1
1018 *

```

} 行番号の GO A

他は スタートメント 処理

"POKE" 処理

} ADRS 計算 (関数等)

} 代入処理

(IX スタック PSH)

IX スタック 本処理

(IX スタック PUL)

1019 0716 8D 00. CLCPL BSR EXPR2

019F 0716

02A1 0716

0378 0716

037E 0716

0676 0716

06CA 4B

1020

*

1021 0718 DF 54

CPUL

STX

WKC

04DC 0718

0521 0718

059E 0718

06EA 2D

1022 071A DE 72

CPUL1

LDX

CSP

06AF 071A

1023 071C 8C 00C8

CPX

#CSTCK

1024 071F 27 B2

BEQ

ERR9

1025 0721 A6 01

LDA A

1,X

1026 0723 E6 00

LDA B

0,X

1027 0725 08

INX

1028 0726 08

INX

1029 0727 DF 72

STX

CSP

1030 0729 DE 54

LDX

WKC

1031 072B 39

RTN13

RTS

1032

*

1033 072C DE 56

EXPR1

LDX

XS

1034 072E 7E 0516

EXPR2

JMP

EXPR

0717 16

1035

*

1036 0731 4F

GOTO

CLR A

1037 0732 97 5F

GOSUB

STA A

ADP+3

1038 0734 8D E0

BSR

CLCPL

1039 0736 BD 01A5

G01

JSR

SETLN

06E2 53

1040 0739 5D

TST B

1041 073A 26 00.

BNE

G02

1042 073C 4D

TST A

1043 073D 27 EC

BEQ

RTN13

1044 073F 7D 005F

G02

TST

ADP+3

073B 03

1045 0742 27 02

BEQ

*+4

1046 0744 8D AA

BSR

PSHX

1047 0746 DE 6E

LDX

EADRS

1048 0748 BD 030D

JSR

CPBA

1049 074B 24 00.

BCC

G03

1050 074D 76 006E

ROR

EADRS

三行目
D, A

紫門ス
PUL

紫門ス
木ノ

"G0"

"G0SUB"

三行目
行番号 SET

0 なる Y の封

G0SUB なる
IX スタック PSH

飛ん先か 現在の行
より前か後か?

1051	0750	BD 027D	G03	JSR	SCHEND
	074C	03			
1052	0753	25 00.		BCS	ERR11
1053	0755	7E 02BF		JMP	RN1
1054			*		
1055	0758	81 23		LET	CHP A #'#'
	02D3	0758			
1056	075A	27 8A		BEQ	POKE
1057	075C	BD 06B9	LET1	JSR	LET3
1058	075F	A7 01		STA A	1,X
1059	0761	E7 00	LET2	STA B	0,X
	06EF	71			
1060	0763	DE 56		LDX	XS
1061	0765	39		RTS	
1062			*		
1063	0766	8D F4	FOR	BSR	LET1
1064	0768	96 53		LDA A	WKB+1
1065	076A	D6 52		LDA B	WKB
1066	076C	BD 05E4		JSR	CPS2
1067	076F	BD 0314		JSR	TSP
1068	0772	8C 0000.		CPX	#T0
1069	0775	26 00.		BNE	ERR12
1070	0777	8D B3		BSR	EXPR1
1071	0779	BD 0314		JSR	TSP
1072	077C	8C 0000.		CPX	#STEP
1073	077F	27 00.		BEQ	FOR1
1074	0781	86 01		LDA A	#1
1075	0783	5F		CLR B	
1076	0784	BD 05E2		JSR	CPS1
1077	0787	20 00.		BRA	FOR2
1078			*		
1079	0789	8D A1	FOR1	BSR	EXPR1
	0780	08			
1080	078B	DF 56		STX	XS
1081	078D	96 57	FOR2	LDA A	XS+1
	0788	04			
1082	078F	D6 56		LDA B	XS
1083	0791	BD 05E4		JSR	CPS2
1084	0794	96 53		LDA A	WKB+1
1085	0796	D6 52		LDA B	WKB
1086	0798	DE 74		LDX	FSP
1087	079A	8C 0100		CPX	#FSTCK
1088	079D	27 00.		BEQ	FOR5
1089	079F	8D 00.		BSR	TSTPM
1090	07A1	24 00.		BCC	FOR4
1091	07A3	DE 74		LDX	FSP
1092	07A5	8C 00C8		CPX	#CSTCK
1093	07A8	26 00.		BNE	FOR5

行番号 カ-9

(代入文)

代入

"FOR"

制御変数に初期値を代入

} 制御変数のADDRS

算術スタック PSH

} 'TO' 文

最終アドレス計算 → PSH

} "STEP" 文

} STEP 値に 1 を加

STEP 計算 → PSH

} ループの終了ADDRS
PSH

} 制御変数ADDRS

FOR NEXT スタックポインタ

} 変数に 1 を加

} ループ文終了

1094 07AA C6 0C	ERR12 LDA B #12
0776 33	
1095 07AC 8C	FCB \$8C
1096 07AD C6 0B	ERR11 LDA B #11
0754 58	
1097 07AF 8C	FCB \$8C
1098 07B0 C6 0D	ERR13 LDA B #13
1099 07B2 7E 0000.	JMP ERR
1100	*
1101 07B5 8D 00.	FOR4 BSR INX8
07A2 12	
1102 07B7 8D 00.	FOR5 BSR FORPSH
079E 18	
07A9 0D	
1103 07B9 20 00.	BRA NXT3
1104	*
1105 07BB BD 030D	TSTPM JSR CPBA
07A0 1A	
1106 07BE 27 00.	BEQ RTN11
1107 07C0 8D 00.	BSR INX8
1108 07C2 8C 0100	TPM1 CPX #FSTCK
1109 07C5 26 F4	BNE TSTPM
1110 07C7 0D	SEC
1111 07C8 39	RTN11 RTS
07BF 08	
1112	*
1113 07C9 8D 00	FORPSH BSR *+2
07B8 10	
1114 07CB 8D 00	BSR *+2
1115 07CD BD 0718	JSR CPUL
1116 07D0 09	DEX
1117 07D1 09	DEX
1118 07D2 7E 04E4	JMP STBA
1119	*
1120 07D5 BD 067C	NEXT JSR VAR
1121 07D8 DF 56	STX XS
1122 07DA DE 74	LDX FSP
1123 07DC 24 00.	BCC NXT1
1124 07DE A6 01	LDA A 1,X
1125 07E0 E6 00	LDA B 0,X
1126 07E2 8D DE	NXT1 BSR TPM1
07DD 04	
1127 07E4 25 CA	BCS ERR13
1128 07E6 DF 74	STX FSP

実行スタック

FOR NEXT STACKA
転送

テストパラメータ
2バイト比較

他のテスト
FOR NEXT Loop
のパラメータと
比較

ループ内
実行を1回
確認

(同じものがないC-1)
(同じものがあるC0)

8バイト

実行スタック

IXの値を2回
転送

"NEXT"

制御変数のADDRESS計算

← FOR NEXT スタックポインタ

制御変数なし
目的の決定

パラメータ

1129 07E8 A6 05		LDA A 5,X
1130 07EA E6 04		LDA B 4,X
1131 07EC EE 00		LDX 0,X
1132 07EE BD 04E0		JSR ADBA
1133 07F1 DE 74		LDX FSP
1134 07F3 E1 02		CHP B 2,X
1135 07F5 0D		SEC
1136 07F6 2D 00.		BLT NXT2
1137 07F8 0C		CLC
1138 07F9 2E 00.		BGT NXT2
1139 07FB A1 03		CHP A 3,X
1140 07FD 27 00.		BEQ NXT4
1141 07FF 46	NXT2	ROR A
07F7 07		
07FA 04		
1142 0800 A8 04		EOR A 4,X
1143 0802 2B 00.		BMI NXT4
1144 0804 8D 00.		BSR INX8
1145 0806 DF 74	NXT3	STX FSP
07BA 4B		
1146 0808 DE 56		LDX XS
1147 080A 39		RTS
1148	*	
1149 080B 08	INX8	INX
07B6 54		
07C1 49		
0805 05		
1150 080C 08		INX
1151 080D 08		INX
1152 080E 08		INX
1153 080F 08		INX
1154 0810 08		INX
1155 0811 08		INX
1156 0812 08		INX
1157 0813 39		RTS
1158	*	
1159 0814 EE 06	NXT4	LDX 6,X
07FE 15		
0803 10		
1160 0816 39		RTS
1161	*	

} STEPのセテリ

1132-9 ADRS
STEPを加算

最終1105-2と上位比較

} 上位が小さい
正 C=0
負 C=1

} 上位が小さい

↓
Loopの符号を考慮に

} 上位が小さい

Loop終了

FORの次のTEXTの
ADRS.

1162 0817 DF 52	INPUT	STX	WKB	
1163 0819 96 52		LDA A	WKB	} DIRECT MODE ちJERR
1164 081B 27 00.		BEQ	ER1	
1165 081D DE 6C		LDX	EOB	} 入力バッファ最終値
1166 081F DF 50		STX	WKA	
1167 0821 DE 52		LDX	WKB	
1168 0823 BD 038B		JSR	PR7	11751L ストリング PRINT
1169 0826 26 00.		BNE	INP1	
1170 0828 8D 00.		BSR	INP3	
1171 082A BD 067C	INP1	JSR	VAR	入力データの ADDR5 計算
0827 02				
1172 082D 25 00.		BCS	ER1	} 入力データの IN 値、IN 値、IN 値以外を ADDR5 PSH
1173 082F 36		PSH A		
1174 0830 37		PSH B		
1175 0831 DF 52		STX	WKB	
1176 0833 DE 50		LDX	WKA	
1177 0835 8D 00.		BSR	PKUP1	
1178 0837 26 00.		BNE	INP2	入力データの IN 値を '?' 出力
1179 0839 86 3F	INA	LDA A	'?'	} 'L' 出力
1180 083B 8D 00.		BSR	OUT1	
1181 083D 8D 00.		BSR	OUTS	
1182 083F BD 012C		JSR	LN4	入力バッファ IN
1183 0842 BD 04A3	INP2	JSR	PKUP	
0838 09				
1184 0845 81 24		CHP A	'\$'	} 入力 MISS 値、
1185 0847 25 00.		BCS	MISS	
1186 0849 81 5B		CHP A	'L'	
1187 084B 25 00.		BCS	OK	
1188 084D DF 54	MISS	STX	WKC	} MISS 値 RE-ENTER を 出力して 入力やり直し
0848 04				
1189 084F CE 0000.		LDX	#REENT	
1190 0852 BD 0000.		JSR	MGOT	
1191 0855 20 E2		BRA	INA	
1192	*			
1193 0857 BD 0716	OK	JSR	CLCPL	データの演算
084C 0A				
1194 085A 8D 00.		BSR	INP3	
1195 085C DF 50		STX	WKA	
1196 085E 30		TSX		} PSH した ADDR5 を IN に LOAD
1197 085F EE 00		LDX	0,X	
1198 0861 31		INS		
1199 0862 31		INS		
1200 0863 BD 04E4		JSR	STBA	
1201 0866 DE 52		LDX	WKB	
1202 0868 8D 00.		BSR	PKUP1	
1203 086A 27 00.		BEQ	RTN12	

1204 086C 81 2C		CMP A	#','
1205 086E 26 00.		BNE	ER1
1206 0870 08		INX	
1207 0871 20 B7		BRA	INP1
1208	*		
1209 0873 7E 0159	ER1	JMP	ERR1
081C 56			
082E 44			
086F 03			
1210	*		
1211 0876 8D 00.	INP3	BSR	INP4
0829 4C			
085B 1A			
1212 0878 26 00.		BNE	RTN12
1213 087A 08		INX	
1214 087B 39	RTN12	RTS	
086B 0F			
0879 01			
1215	*		
1216 087C 36	INP4	PSH A	
0877 04			
1217 087D 8D 00.		BSR	PKUP1
1218 087F 81 2C		CMP A	#','
1219 0881 32		PUL A	
1220 0882 39		RTS	
1221	*		
1222 0883 7E 04A3	PKUP1	JMP	PKUP
0836 4C			
0869 19			
087E 04			
1223	*		
1224 0886 86 20	OUTS	LDA A	#','
011F 0886			
036E 0886			
0384 0886			
083E 47			
1225 0888 7E 0464	OUT1	JMP	OUT
083C 4B			
1226	*		
1227 088B 8D 00.	DDLY	BSR	DELAY
1228 088D DF 54	DELAY	STX	WKC
088C 00			
1229 088F CE E100		LDX	#E100
1230 0892 09	DEL1	DEX	
1231 0893 26 FD		BNE	DEL1
1232 0895 DE 54		LDX	WKC
1233 0897 39	RTN10	RTS	

' , ' ち3
*3-1000入

1.5秒 DELAY
0.5秒 DELAY

```

1234 *
1235 0898 86 12 SAVE LDA A ##12
1236 089A 8D EC BSR OUT1
1237 089C 97 5C STA A FS
1238 089E 8D EB BSR DDLY
1239 08A0 20 00. BRA LST1
1240 *
1241 08A2 7F 005C LIST CLR FS
1242 08A5 5F LST1 CLR B
    08A1 03
1243 08A6 D7 66 STA B WKUSE
1244 08A8 86 01 LDA A #1
1245 08AA BD 01A5 JSR SETLN
1246 08AD C6 7F LDA B ##7F
1247 08AF 86 FF LDA A ##FF
1248 08B1 36 PSH A
1249 08B2 8D CF BSR PKUP1
1250 08B4 32 PUL A
1251 08B5 27 00. BEQ LST2
1252 08B7 BD 0716 JSR CLCPL
1253 08BA BD 01A5 JSR SETLN
1254 08BD 8D BD BSR INP4
1255 08BF 26 00. BNE LST2
1256 08C1 08 INX
1257 08C2 BD 0716 JSR CLCPL
1258 08C5 97 59 LST2 STA A SPS+1
    08B6 0E
    08C0 04
1259 08C7 D7 58 STA B SPS
1260 08C9 DE 58 LDX SPS
1261 08CB 2E 00. BGT LST3
1262 08CD CE 7FFF LDX ##7FFF
1263 08D0 DF 58 STX SPS
1264 08D2 BD 0281 LST3 JSR SCHLIN
    08CC 05
1265 08D5 8C FCB #8C
1266 *
1267 08D6 8D 00. LST4 BSR LST5
1268 08D8 96 59 LDA A SPS+1
1269 08DA D6 58 LDA B SPS
1270 08DC BD 030D JSR CPBA
1271 08DF 24 F5 BCC LST4
1272 08E1 DF 54 CNTCRD STX WKC
1273 08E3 CE 0000. LDX #CNT
1274 08E6 8D 00. BSR NGOT
1275 08E8 7E 01C1 TOEND JMP END
1276 *

```

出力をCASE7に
設定

行番号を1にする

最終行番号

引数9に1

1引数なし

1引数

1引数2引数なし

2引数

最終行番号

第2引数が0の場合
最初までリスト

最終行番号にちなんで
LIST

CASE7 入出力解除
の CONTROL CORD出力

エディットモードにもと

(リスト処理サブルーチン)

1277	08EB 86 02	LST5	LDA A #2	
	08D7 13			
1278	08ED 8D 99		BSR OUT1	
1279	08EF 8D 00.		BSR LST7	
1280	08F1 08		INX	
1281	08F2 86 20		LDA A #' '	
1282	08F4 8D 00.		BSR MGOT1	
1283	08F6 8D 00.		BSR CR	
1284	08F8 96 5C		LDA A FS	
1285	08FA 26 91		BNE DELAY	
1286	08FC 39		RTS	
1287		*		
1288	08FD DF 54	CR	STX WKC	
	016F 08FD			
	01CE 08FD			
	036B 08FD			
	08F7 05			
1289	08FF CE 0000.		LDX #CRLF	
1290	0902 8D 00.	MGOT	BSR MGOUT	
	0853 0902			
	08E7 1A			
1291	0904 7F 0065		CLR CHCT	
1292	0907 DE 54	LST6	LDX WKC	
1293	0909 39		RTS	
1294		*		
1295	090A DF 54	LST7	STX WKC	
	08F0 19			
1296	090C BD 0417	LST8	JSR PNUM	
1297	090F 20 F6		BRA LST6	
1298		*		
1299	0911 BD 0464	MGOT1	JSR OUT	
	08F5 1B			
1300	0914 08		INX	
1301	0915 A6 00	MGOUT	LDA A 0,X	
	01CB 0915			
	0903 11			
1302	0917 26 F8		BNE MGOT1	
1303	0919 08		INX	
1304	091A 39		RTS	
1305		*		

作の頭とす CONTROL
CRの出力

行番号出力

スペースに続いて
TEXT 1行出力

'CR'出力

SAVE した DELAY を
置く

BINARY を ASCII へ
出力

IX の示す所から 100
まで 出力する。

1306	091B 8E A047	ERR	LDS	#STACK
	015C 091B			
	0295 091B			
	03AA 091B			
	05FF 091B			
	06D9 091B			
	07B3 091B			
1307	091E 8D DD		BSR	CR
1308	0920 CE 0000.		LDX	#ERROR
1309	0923 8D F0		BSR	MGOUT
1310	0925 17		TBA	
1311	0926 5F		CLR B	
1312	0927 D7 66		STA B	WKUSE
1313	0929 8D 00.		BSR	CCP3
1314	092B DE 6E		LDX	ENDRS
1315	092D 2B B9		BNI	TOEND
1316	092F 8D CC		BSR	CR
1317	0931 8D B8		BSR	LST5
1318	0933 20 B3		BRA	TOEND
1319		*		
1320	0935 E6 00	USING	LDA B	0.X
1321	0937 D7 66		STA B	WKUSE
1322	0939 E6 01		LDA B	1.X
1323	093B D7 67		STA B	WKUSE+1
1324	093D 08		INX	
1325	093E 08		INX	
1326		*		
1327	093F BD 0716	CCP2	JSR	CLCPL
1328	0942 36	CCP3	PSH A	
	092A 17			
1329	0943 37		PSH B	
1330	0944 DF 54		STX	WKC
1331	0946 30		TSX	
1332	0947 8D C3		BSR	LST8
1333	0949 33		PUL B	
1334	094A 32		PUL A	
1335	094B 7F 0066		CLR	WKUSE
1336	094E 39		RTS	
1337		*		
1338	094F 44	OHL	LSR A	
1339	0950 44		LSR A	
1340	0951 44		LSR A	
1341	0952 44		LSR A	

SPの初期化

改行に
'ERROR NO.'を出力

ERROR NUMBERを
出力

DIRECT MODE 以外では
ERRORの行を
LISTする。

USING 処理

式の値演算

BAの値を
ASCII化して出力

上位4ビットの16進出力

1342 0953 84 0F	OHR	AND A	#\$F
1343 0955 81 0A		CMP A	#\$A
1344 0957 25 02		BCS	*+4
1345 0959 8B 07		ADD A	#7
1346 095B 8B 30		ADD A	#'0'
1347 095D 7E 0464	TOUT	JMP	OUT
1348	*		
1349 0960 36	04H	PSH A	
1350 0961 17		TBA	
1351 0962 8D 00.		BSR	02H
1352 0964 32		PUL A	
1353 0965 36	02H	PSH A	
0963 01			
1354 0966 8D E7		BSR	OHL
1355 0968 32		PUL A	
1356 0969 20 E8		BRA	OHR
1357	*		
1358 096B 86 24	HD	LDA A	#'\$'
1359 096D 8D EE		BSR	TOUT
1360 096F A6 00		LDA A	0.X
1361 0971 08		INX	
1362 0972 81 54		CMP A	#'T'
1363 0974 27 00.		BEQ	HDT
1364 0976 BD 0716	HDF	JSR	CLCPL
1365 0979 8D E5		BSR	04H
1366 097B 7E 0886	TOUTS	JMP	OUTS
1367	*		
1368 097E BD 0716	HDT	JSR	CLCPL
0975 08			
1369 0981 8D E2		BSR	02H
1370 0983 20 F6		BRA	TOUTS
1371	*		
1372 0985 BD 015E	GET	JSR	IN
1373 0988 5F		CLR B	
1374 0989 7E 05E4	GET1	JMP	CPS2
1375	*		
1376 098C BD 3CB6	COPY	JSR	VRAMP
1377 098F 20 00.		BRA	G1

下位4ビットの
16進出力

2byte 4桁の16進出力

1byte 2桁の16進出力

"HDT" or "HDF"

'\$' 出力

HDF 処理

HDT 処理

"GET\$" 内部

(OPTION) "COPY"
スタートシフト

OPTION

1378		*		
1379		* GRAPHIC		
1380		*		
1381	0991 BD 3C06	NEGB	JSR	NEG
1382	0994 20 00.		BRA	G1
1383		*		
1384	0996 8D 00.	PICKB	BSR	SUB2
1385	0998 8D 00.		BSR	SUB4
1386	099A BD 3BF7		JSR	PICK
1387	099D 17		TBA	
1388	099E 5F		CLR B	
1389	099F DE 56		LDX	XS
1390	09A1 7E 05E4	GE1	JMP	CPS2
1391		*		
1392	09A4 A6 00	SETPT	LDA A	0,X
1393	09A6 81 57		CMP A	#'W'
1394	09A8 26 00.		BNE	G2
1395	09AA 8D 00.		BSR	SUB3
1396	09AC BD 3BD7		JSR	SET
1397	09AF DE 56	G1	LDX	XS
	0990 1E			
	0995 19			
1398	09B1 39		RTS	
1399		*		
1400	09B2 81 42	G2	CMP A	#'B'
	09A9 08			
1401	09B4 26 00.		BNE	G3
1402	09B6 8D 00.		BSR	SUB3
1403	09B8 BD 3BE8		JSR	RESET
1404	09BB 20 F2		BRA	G1
1405		*		
1406	09BD 81 52	G3	CMP A	#'R'
	09B5 07			
1407	09BF 26 00.		BNE	ERR15
1408	09C1 8D 00.		BSR	SUB3
1409	09C3 BD 3BF0		JSR	REVERS
1410	09C6 20 E7		BRA	G1
1411		*		
1412	09C8 E6 00	SUB5	LDA B	0,X
	0620 09C8			
	0699 09C8			
1413	09CA C1 28		CMP B	#'C'
1414	09CC 39		RTS	
1415		*		

[グラフィック 命令]

"NEG"

"!P(x,y)" 円数

"!W(x,y)" スタート
ポイント

"!B(x,y)" スタート
ポイント

"!R(x,y)" スタート
ポイント

1416 09CD 8D 00.	SUB3	BSR	SUB1
09AB 21			
09B7 15			
09C2 0A			
1417 09CF 81 40	SUB4	CMP A	#64
0999 35			
1418 09D1 24 00.		BCC	ERR16
1419 09D3 C1 30		CMP B	#48
1420 09D5 24 00.		BCC	ERR16
1421 09D7 39		RTS	
1422	*		
1423 09D8 08	SUB1	INX	
09CE 09			
1424 09D9 8D ED		BSR	SUB5
1425 09DB 26 00.		BNE	ERR
1426 09DD 08	SUB6	INX	
1427 09DE 8D 0000.	SUB2	JSR	SUB7
0997 46			
1428 09E1 DA 5C		ORA B	ADP
1429 09E3 26 00.		BNE	ERR16
1430 09E5 16		TAB	
1431 09E6 96 5D		LDA A	ADP+1
1432 09E8 39		RTS	
1433	*		
1434 09E9 C6 0F	ERR15	LDA B	#15
09C0 28			
1435 09EB 8C		FCB	#8C
1436 09EC C6 10	ERR16	LDA B	#16
09D2 19			
09D6 15			
09E4 07			
1437 09EE 7E 091B		JMP	ERR
1438	*		
1439 09F1 24 F9	CMP	BCC	ERR16
1440 09F3 11		CBA	
1441 09F4 2E F6		BGT	ERR16
1442 09F6 39		RTS	
1443	*		
1444 09F7 8D CF	CLR	BSR	SUB5
1445 09F9 26 00.		BNE	ALLCLR
1446 09FB 8D E0		BSR	SUB6
1447 09FD C1 10		CMP B	#16
1448 09FF 8D F0		BSR	CMP

GRAPHIC
31桁範囲の

427

23桁
13桁 ACCA
23桁 ACCB

CLR(X,Y) KEY(X,Y)
31桁範囲 427

"CLR(X,Y)"

は31桁ない

1449 0A01 36	C1	PSH A
1450 0A02 8D 00.		BSR C3
1451 0A04 CB 20		ADD B #32
1452 0A06 89 00		ADC A #0
1453 0A08 97 5E		STA A ADP+2
1454 0A0A D7 5F		STA B ADP+3
1455 0A0C 33		PUL B
1456 0A0D 8D 00.		BSR C3
1457 0A0F 97 5C		STA A ADP
1458 0A11 D7 5D		STA B ADP+1
1459 0A13 C6 DF		LDA B #\$FF-
1460 0A15 DE 5C		LDX ADP
1461 0A17 E7 00	C2	STA B 0,X
1462 0A19 08		INX
1463 0A1A 9C 5E		CPX ADP+2
1464 0A1C 26 F9		BNE C2
1465 0A1E DE 56		LDX XS
1466 0A20 39		RTS
1467	*	
1468 0A21 C6 0F	ALLCLR	LDA B #15
09FA 26		
1469 0A23 4F		CLR A
1470 0A24 20 DB		BRA C1
1471	*	
1472 0A26 4F	C3	CLR A
0A03 22		
0A0E 17		
1473 0A27 54		LSR B
1474 0A28 56		ROR B
1475 0A29 56		ROR B
1476 0A2A 56		ROR B
1477 0A2B 89 C0		ADC A #VRAM
1478 0A2D 39		RTS
1479	*	
1480 0A2E 7E 06D3	ERR9	JMP ERR9
09DC 51		
1481	*	
1482 0A31 8D 00.	CURS	BSR SUB10
1483 0A33 8D A9		BSR SUB2
1484 0A35 81 20		CMP A #32
1485 0A37 24 B3		BCC ERR16
1486 0A39 C1 10		CMP B #16
1487 0A3B 24 AF		BCC ERR16
1488 0A3D 97 65		STA A CHCT
1489 0A3F 8D E5		BSR C3

第2引数

第1引数

← SPACE CORD
ふたつの V-RAM では \$20
に変更

指定範囲 CLR

引数 3 の ADDR 計算

← V-RAM 開始 ADDR の
上位 16bit

"CURS (X,Y)"
↓ ↓
ACC A ACC B

引数

CURS の 処理

1490	0A41	B7	A094		STA A	DISP
1491	0A44	96	65		LDA A	CHCT
1492	0A46	1B			ABA	
1493	0A47	B7	A095		STA A	DISP+1
1494	0A4A	39			RTS	
1495				*		
1496	0A4B	BD	0605	SGN	JSR	CUL1
1497	0A4E	BD	0718		JSR	CPUL
1498	0A51	5D			TST B	
1499	0A52	2B	00.		BNI	MINS
1500	0A54	26	00.		BNE	PLS
1501	0A56	4D			TST A	
1502	0A57	27	00.		BEQ	ZERO
1503	0A59	5F		PLS	CLR B	
	0A55	03				
1504	0A5A	86	01		LDA A	#1
1505	0A5C	20	00.		BRA	ZERO
1506				*		
1507	0A5E	86	FF	MINS	LDA A	#4FF
	0A53	0A				
1508	0A60	16			TAB	
1509	0A61	7E	05E4	ZERO	JMP	CPS2
	0A58	08				
	0A5D	03				
1510				*		
1511	0A64	7E	05F9	ER7	JMP	ERR7
1512				*		
1513	0A67	BD	09C8	SUB10	JSR	SUB5
	0657	0A67				
	0A32	34				
1514	0A6A	26	F8		BNE	ER7
1515	0A6C	08			INX	
1516	0A6D	39			RTS	
1517				*		
1518	0A6E	8D	F7	SUB11	BSR	SUB10
1519	0A70	BD	0716	SUB7	JSR	CLCPL
	09DF	0A70				
1520	0A73	97	5D		STA A	ADP+1
1521	0A75	D7	5C		STA B	ADP
1522	0A77	BD	063E		JSR	COMMA
1523	0A7A	26	B2		BNE	ER9
1524	0A7C	08			INX	
1525	0A7D	BD	0716		JSR	CLCPL
1526	0A80	36			PSH A	
1527	0A81	BD	060F		JSR	CUL4
1528	0A84	32			PUL A	
1529	0A85	DF	56		STX	XS
1530	0A87	39			RTS	

計算したADRSと
OSのアドレスレジスタ
に書きこむ

"SGIV(X)" 関数

引数計算
B, A

正 → 1
ゼロ → 0
負 → -1

1バイト以内の
2引数処理

第1引数

第2引数

1531	*		
1532 0A88 8D E4	AND	BSR	SUB11
1533 0A8A 94 5D		AND A	ADP+1
1534 0A8C D4 5C		AND B	ADP
1535 0A8E 7E 05E4	AND1	JMP	CPS2
1536	*		
1537 0A91 8D DB	OR	BSR	SUB11
1538 0A93 9A 5D		ORA A	ADP+1
1539 0A95 DA 5C		ORA B	ADP
1540 0A97 20 F5		BRA	AND1
1541	*		
1542 0A99 8D D3	XOR	BSR	SUB11
1543 0A9B 98 5D		EOR A	ADP+1
1544 0A9D D8 5C		EOR B	ADP
1545 0A9F 20 ED		BRA	AND1
1546	*		
1547 0AA1 BD 0716	RES	JSR	CLCPL
1548 0AA4 DF 50		STX	WKA
1549 0AA6 DE 36		LDX	BOP
1550 0AA8 BD 0299		JSR	RS1
1551 0AAB 25 00.		BCS	ER11
1552 0AAD 09		DEX	
1553 0AAE 20 00.		BRA	RE4
1554	*		
1555 0AB0 DF 50	READ	STX	WKA
1556 0AB2 DE 76		LDX	DP
1557 0AB4 BD 04A3		JSR	PKUP
1558 0AB7 81 2C		CMP A	#','
1559 0AB9 27 00.		BEG	RE3
1560 0ABB 4D		TST A	
1561 0ABC 27 00.		BEG	RE2
1562 0ABE 7E 0159		JMP	ERR1
1563	*		
1564 0AC1 BD 0276	RE1	JSR	NXTL1
1565 0AC4 A6 01	RE2	LDA A	1,X
0ABD 06			
1566 0AC6 81 00		CMP A	#\$80
1567 0AC8 27 00.		BEG	ERR17
1568 0ACA A6 03		LDA A	3,X
1569 0ACC 08		INX	
1570 0ACD 81 2A		CMP A	#'*'
1571 0ACF 26 F0		BNE	RE1
1572 0AD1 08		INX	
1573 0AD2 08		INX	

"AND(X,Y)" 関数

"OR(X,Y)" 関数

"XOR(X,Y)" 関数

"RESTORE" スタートメント
引数付き

TEXT 最初から行番を
付す

"READ" 関数

DATA 本行
次の TEXT Pick up
';' までの DATA

文末までの DATA 文を
付す

次の文末を付す。

TEXT の最後まの行つて
5 ERROR

'*' でなかったら
次の文末を付す。

1574	0AD3 08	RE3	INX	
	0ABA 18			
1575	0AD4 BD 0516		JSR	EXPR
1576	0AD7 DF 76	RE4	STX	DP
	0AAF 27			
1577	0AD9 DE 50		LDX	WKA
1578	0ADB 39		RTS	
1579		*		"DEL"
1580	0ADC 8D C3	DEL	BSR	RES
1581	0ADE DE 76		LDX	DP
1582	0AE0 08		INX	
1583	0AE1 BD 017D		JSR	DEL2
1584	0AE4 7E 01C1		JMP	END
1585		*		
1586	0AE7 7E 07AD	ER11	JMP	ERR11
	0AAC 3A			
1587		*		
1588	0AEA C6 11	ERR17	LDA B	#17
	0AC9 20			
1589	0AEC 7E 091B		JMP	ERR
1590		*		
1591	0AEF BD 09C8	KEY	JSR	SUB5
1592	0AF2 27 00.		BED	K0
1593	0AF4 C6 07		LDA B	#7
1594	0AF6 4F		CLR A	
1595	0AF7 20 00.		BRA	K1
1596		*		
1597	0AF9 BD 09DD	K0	JSR	SUB6
	0AF3 05			
1598	0AFC C1 06		CMP B	#6
1599	0AFE BD 09F1		JSR	CMP
1600	0B01 5C		INC B	
1601	0B02 B7 E006	K1	STA A	PIACRB
	0AF8 09			
1602	0B05 36		PSH A	
1603	0B06 B6 E006		LDA A	PIACRB
1604	0B09 81 55		CMP A	##55
1605	0B0B 23 00.		BLS	K2
1606	0B0D 32		PUL A	
1607	0B0E 4C		INC A	
1608	0B0F 11		CBA	
1609	0B10 26 F0		BNE	K1
1610	0B12 86 64		LDA A	#100
1611	0B14 C1		FCB	\$C1
1612	0B15 31	K2	INS	
	0B0C 08			
1613	0B16 5F		CLR B	
1614	0B17 7E 05E4		JMP	CPS2

DATAの他を計算

Yの行から内とNEWと
同じ処理する

"KEY" 関数 (32 HGFH)

別記

```

1615 *
1616 0B1A BD 02A0 UNTIL JSR SUBIF
1617 0B1D 27 00. BEQ UNT
1618 0B1F BD 0706 JSR PULX
1619 0B22 DE 54 LDX WKC
1620 0B24 39 RTS
1621 *
1622 0B25 DE 70 UNT LDX XSP
      0B1E 06
1623 0B27 EE 00 LDX 0,X
1624 0B29 00 INX
1625 0B2A 39 RTS
1626 *
1627 0B2B 03 CNT FCB 3,$13,$14,

```

0

```

0B2C 13
0B2D 14
0B2E 00
0B2F 0B2B

```

```

1628 *
1629 0B2F 45 ERROR FCB 'ERROR NO.

```

,0

```

0B30 52
0B31 52
0B32 4F
0B33 52
0B34 20
0B35 4E
0B36 4F
0B37 2E
0B38 00
0B39 0B2F

```

```

1630 *
1631 0B39 00 READY FCB $0,$A, 'RE

```

RDY',7,0

```

0B3A 0A
0B3B 52
0B3C 45
0B3D 41
0B3E 44
0B3F 59
0B40 07
0B41 00
0B42 0B39

```

1632

*

"UNTIL"

条件並の値
は 不成立

成立したと IX STACK を
PULして 次の文へ

不成立なら

IX STACK の内容を
取りだし、その ADDRへ

SAVE, LIST, 解除
の後の CONTROL CODE

1633	0B42 00	CRLF	FCB	\$D,\$R,0
	0B43 0A			
	0B44 00			
	0900 0B42			
1634		*		
1635	0B45 52	REENT	FCB	'RE-ENTER'
	0B46 45			
	0B47 2D			
	0B48 45			
	0B49 4E			
	0B4A 54			
	0B4B 45			
	0B4C 52			
	0B4D 00			
	0B50 0B45			
1636		*		
1637	0080	Z	EQU	\$80
1638		*		
1639	0B4C	COMAND	EQU	*-2
	0272 0B4C			
1640	0B4E 4C		FCB	'LIST'+Z
	0B4F 49			
	0B50 53			
	0B51 D4			
1641	0B52 0BA2		FDB	LIST
1642	0B54 4C		FCB	'LOAD'+Z
	0B55 4F			
	0B56 41			
	0B57 C4			
1643	0B58 018E		FDB	LOAD
1644	0B5A 52		FCB	'RUN'+Z
	0B5B 55			
	0B5C CE			
1645	0B5D 02BA		FDB	RUN
1646	0B5F 45		FCB	'EX'+Z
	0B60 D8			
1647	0B61 F0B1		FDB	\$F0B1
1648	0B63 41		FCB	'AUTO'+Z
	0B64 55			
	0B65 54			
	0B66 CF			
1649	0B67 019B		FDB	AUTO
1650	0B69 4E		FCB	'NEW'+Z
	0B6A 45			
	0B6B D7			
1651	0B6C 01BF		FDB	NEW

※
 [注: TABLEの KEY WORDの
 最後の文字の上イE1ゆけに
 1を573

}
 ← MONITORの ADDR

1652 0B6E 53	FCB	'SAVE'+Z
0B6F 41		
0B70 56		
0B71 C5		
1653 0B72 0B98	FDB	SAVE
1654 0B74 41	FCB	'APPEND'+Z
0B75 50		
0B76 50		
0B77 45		
0B78 4E		
0B79 C4		
1655 0B7A 0190	FDB	APPEND
1656 0B7C 44	FCB	'DEL'+Z
0B7D 45		
0B7E CC		
1657 0B7F 0ADC	FDB	DEL
1658		
1659 0B7F	* STATE EQU	*-2
02D8 0B7F		
1660 0B81 4E	FCB	'NEXT'+Z
0B82 45		
0B83 58		
0B84 D4		
1661 0B85 07D5	FDB	NEXT
1662 0B87 55	FCB	'UNTIL'+Z
0B88 4E		
0B89 54		
0B8A 49		
0B8B CC		
1663 0B8C 0B1A	FDB	UNTIL
1664 0B8E 47	FCB	'GOTO'+Z
0B8F 4F		
0B90 54		
0B91 CF		
1665 0B92 0731	FDB	GOTO
1666 0B94 47	FCB	'GOSUB'+Z
0B95 4F		
0B96 53		
0B97 55		
0B98 C2		
1667 0B99 0732	FDB	GOSUB
1668 0B9B 54	FCB	'THEN'+Z
0B9C 48		
0B9D 45		
0B9E CE		
1669 0B9F 06DB	FDB	THEN

1670	0BA1 46	FCB	'FOR'+Z
	0BA2 4F		
	0BA3 D2		
1671	0BA4 0766	FDB	FOR
1672	0BA6 44	FCB	'DO'+Z
	0BA7 CF		
1673	0BA8 06F0	FDB	PSHX
1674	0BAA A1	FCB	'!' +Z
1675	0BAB 09A4	FDB	SETPT
1676	0BAD 52	FCB	'RET'+Z
	0BAE 45		
	0BAF D4		
1677	0BB0 0706	FDB	PULX
1678	0BB2 49	FCB	'INPUT'+Z
	0BB3 4E		
	0BB4 50		
	0BB5 55		
	0BB6 D4		
1679	0BB7 0817	FDB	INPUT
1680	0BB9 49	FCB	'IF'+Z
	0BBA C6		
1681	0BBB 02A8	FDB	IF
1682	0BBD 50	FCB	'PRINT'+Z
	0BBE 52		
	0BBF 49		
	0BC0 4E		
	0BC1 D4		
1683	0BC2 0340	FDB	PRINT
1684	0BC4 43	FCB	'CLR'+Z
	0BC5 4C		
	0BC6 D2		
1685	0BC7 09F7	FDB	CLR
1686	0BC9 43	FCB	'CURS'+Z
	0BCA 55		
	0BCB 52		
	0BCC D3		
1687	0BCD 0A31	FDB	CURS
1688	0BCF 4E	FCB	'NEG'+Z
	0BD0 45		
	0BD1 C7		
1689	0BD2 0991	FDB	NEGB
1690	0BD4 52	FCB	'RESTORE'+
	0BD5 45		
	0BD6 53		
	0BD7 54		
	0BD8 4F		

} GRAPHIC 文字テキスト

Z

0BD9 52			
0BDA C5			
1691 0BDB 0AA1	FDB	RES	
1692 0BDD 52	FCB	'REM'+Z	
0BDE 45			
0BDF CD			
1693 0BE0 0278	FDB	REM	
1694 0BE2 AA	FCB	'*'+Z	
1695 0BE3 0278	FDB	REM	
1696 0BE5 45	FCB	'END'+Z	
0BE6 4E			
0BE7 C4			
1697 0BE8 01C1	FDB	END	
1698 0BEA 53	FCB	'STOP'+Z	
0BEB 54			
0BEC 4F			
0BED D0			
1699 0BEE 01AA	FDB	STOP	
1700 0BF0 43	FCB	'COPY'+Z	
0BF1 4F			
0BF2 50			
0BF3 D9			
1701 0BF4 098C	FDB	COPY	
1702 0BF6 FF	FCB	\$\$F	
1703 0BF7 0A2E	FDB	ER9	
1704			
1705			
05CE 0BF7	* FUNC	EQU	*-2
1706 0BF9 52	FCB	'RND'+Z	
0BFA 4E			
0BFB C4			
1707 0BFC 0643	FDB	RND	
1708 0BFE 52	FCB	'READ'+Z	
0BFF 45			
0C00 41			
0C01 C4			
1709 0C02 0AB0	FDB	READ	
1710 0C04 41	FCB	'ABS'+Z	
0C05 42			
0C06 D3			
1711 0C07 04E9	FDB	ABS	
1712 0C09 4D	FCB	'MOD'+Z	
0C0A 4F			
0C0B C4			
1713 0C0C 061F	FDB	MOD	

DATA 文

OPTION とは 各自が
自分の スタートメントに
変える

一致する
スタートメントが なかった
ERROR NO. 9

1714 0C0E 21		FCB	'!P(' + Z
0C0F 50			
0C10 A8			
1715 0C11 0996		FDB	PICKB
1716 0C13 55		FCB	'USER' + Z
0C14 53			
0C15 45			
0C16 D2			
1717 0C17 0656		FDB	USER
1718 0C19 41		FCB	'AND' + Z
0C1A 4E			
0C1B C4			
1719 0C1C 0A88		FDB	AND
1720 0C1E 4F		FCB	'OR' + Z
0C1F D2			
1721 0C20 0A91		FDB	OR
1722 0C22 58		FCB	'XOR' + Z
0C23 4F			
0C24 D2			
1723 0C25 0A99		FDB	XOR
1724 0C27 53		FCB	'SGN' + Z
0C28 47			
0C29 CE			
1725 0C2A 0A4B		FDB	SGN
1726 0C2C 4B		FCB	'KEY' + Z
0C2D 45			
0C2E D9			
1727 0C2F 0AEF		FDB	KEY
1728 0C31 47		FCB	'GET\$' + Z
0C32 45			
0C33 54			
0C34 A4			
1729 0C35 0985		FDB	GET
1730 0C37 FF		FCB	\$FF
1731 0C38 05F9		FDB	ERR?
1732	*		
1733 0C38	FRNEXT EBU	*-2	
0317 0C38			
1734 0C3A 54		FCB	'TO' + Z
0C3B CF			
1735 0C3C 0000	TO	FDB	0
0773 0C3C			
1736 0C3E 53		FCB	'STEP' + Z
0C3F 54			
0C40 45			
0C41 D0			

- 3433 1076 かたかな
 た3
 ERROR No. 7.

1737	0C42	STEP	EQU	*
	077D 0C42			
1738		*		
1739	0C40	FMT	EQU	*-2
	034E 0C40			
1740	0C42 43		FCB	'CHR'+Z
	0C43 48			
	0C44 52			
	0C45 A4			
1741	0C46 0377		FDB	CHR
1742	0C48 54		FCB	'TAB'+Z
	0C49 41			
	0C4A C2			
1743	0C4B 037D		FDB	TAB1
1744	0C4D 48		FCB	'HD'+Z
	0C4E C4			
1745	0C4F 096B		FDB	HD
1746	0C51 55		FCB	'USING'+Z
	0C52 53			
	0C53 49			
	0C54 4E			
	0C55 C7			
1747	0C56 0935		FDB	USING
1748	0C58 FF		FCB	\$FF
1749	0C59 093F		FDB	CCP2
1750		*		
1751	0C5B	PROG	EQU	*
1752	0C5B		END	

[FORMATTE K]

INI4	341C	INEEE	35E6	OUTEEE	3476
BREAK	369B	NEG	3C06	PICK	3BF7
SET	3BD7	RESET	3BE8	REVERS	3BF0
VRAMPR	3CB6	DISP	A094	STACK	A047
IXSTCK	A000	VRAM	00C0	PIACRB	E006
UTOP	0000	BOP	0036	EOP	0038
NEMEND	003A	IXV	003C	WKA	0050
WKB	0052	WKC	0054	XS	0056
SPS	0058	MDS	005A	FS	005C
ADP	005C	RNDS	0060	FL0D	0062
FAUT	0063	LPCT	0064	CHCT	0065
WKUSE	0066	LNB	006A	EOB	006C
EADRS	006E	XSP	0070	CSP	0072
FSP	0074	DP	0076	BFFR	0078
EOBF	00C0	CSTCK	00C8	FSTCK	0100

START	0100	CSTART	01B2	WSTRT	0103
END	01C1	BGNTXT	0106	ENDTXT	0108
LINEIN	010A	CSTIN	0166	LN2	0125
ADBA	04E0	PNUM	0417	OUTS	0886
LN6	0135	LN1	0123	CR1	016E
COT	0198	LN3	012A	CR2	0171
LN4	012C	LN5	012F	IN	015E
LN7	014B	ERR1	0159	ERR	091B
CR	08FD	RTN0	0177	PTST	0178
DEL2	017D	BRK	0184	LOAD	018E
APPEND	0190	CNT11	0196	EDIT	01D3
OUT	0464	AUTO	019B	CLCPL	0716
SETLN	01A5	STOP	01AA	PRINT	0340
PSHX	06F0	END1	01C7	NEW	01BF
INITP	024E	READY	0B39	MGOUT	0915
ED1	01D6	PKUP	04A3	ATOB	053C
DRCT1	022F	ED2	01E9	SCHLIN	0281
INSERT	020F	NXTL1	0276	ED4	0205
ED3	01FF	INST4	024A	SIZE	02AE
INST1	0231	ERR2	0292	DIRECT	026C
INST2	0234	INST3	0244	CHVAR	0483
RN3	02D2	COMAND	0B4C	RN4	02DA
NXTL2	0277	REM	0278	SCHEND	027D
SL1	0284	ERR3	028F	SL3	029A
SL2	0297	RS1	0299	CPBA	030D
RTN1	029F	SUBIF	02A0	IF	02A8
RN2	02CA	END2	02B7	RUN	02BA
RN1	02BF	LET	0758	RN5	02E0
STATE	0B7F	TBLJP	05D0	HB1	02EA
HB2	02EF	HTOB	02F6	TSTHEX	02FE
TSTN	0564	RTN2	030C	TSF	0314
FRNEXT	0C38	TBLSCH	0319	TS1	031C
TS2	0320	TS3	0336	TS4	0339
TS5	033C	PK1	0367	PR4	036A
PR1	0347	PR7	038B	PR2	0353
FMT	0C40	PR3	0361	ERR4	03A4
PR6	0370	PR5	036D	CHR	0377
AOT	037A	TAB1	037D	TAB2	0386
RTN3	03A3	PR8	0399	ERR5	03A7
DIV	03AC	SETCT	03F8	MDV3	03E6
MDV4	03EB	MDV2	03E1	DV1	03D2
MDIV	03D5	MDV1	03D7	RTN4	03F7
ADD	03F2	MULTI	03FF	MLT1	0401
STBA	04E4	ABSX	046D	PU1	0428
PU2	042C	PN1	0435	PN2	0442

PN3	0454	PN4	0456	PN5	045A
OUTC	0467	RTN5	0479	NEGX	0471
CHASC	047A	CHV1	0481	TBL	04AE
ICPKUP	04A2	RTN6	04AF	EX1	04B0
EX2	04BA	EX7	04D7	EX4	04C1
EX3	04BF	TM1	050F	EX6	04CF
EX5	04CB	EX8	04DB	RTN7	04E8
CPUL	0718	ABS	04E9	CUL1	0605
TM2	0511	EX11	04F4	EX12	04F7
EX13	04FA	EX14	04FE	TERM	056F
RTN8	0515	EXPR	0516	LT	0533
GT	0531	EQ	0532	RTN9	0563
ATB1	0543	FUNCT	05B3	TR1	0571
TR3	0583	TR4	058E	TR2	057D
INHEX	0593	CPS2	05E4	PEEK	059B
TBAX	0617	CPS1	05E2	LITRL	05A7
ERR7	05F9	CUL	0601	FUNC	0BF7
CPUSH	05DC	ERR6	05F6	CPS3	05EE
CPS4	05F3	ERR8	05FC	VAR	067C
CUL2	060B	CUL3	060C	CUL4	060F
MOD	061F	SUB5	09C8	MOD1	062A
COMMA	063E	MOD2	0635	RND	0643
USER	0656	SUB10	0A67	US2	065D
CCP1	0675	US3	0670	US4	0678
NOTV	067A	ARR	0691	ERR14	06D0
IND	06A4	DIR	069D	CPUL1	071A
LET3	06B9	ERR9	06D3	LET4	06BE
ERR10	06D6	THEN	06DB	GO1	0736
POKE	06E6	LET2	0761	PX1	06FF
PULX	0706	EXPR2	072E	RTN13	072B
EXPR1	072C	GOTO	0731	GOSUB	0732
GO2	073F	GO3	0750	ERR11	07AD
LET1	075C	FOR	0766	TO	0C3C
ERR12	07AA	STEP	0C42	FOR1	0789
FOR2	078D	FOR5	07B7	TSTPM	07BB
FOR4	07B5	ERR13	07B0	INX8	080B
FORPSH	07C9	NXT3	0806	RTN11	07C8
TPM1	07C2	NEXT	07D5	NXT1	07E2
NXT2	07FF	NXT4	0814	INPUT	0817
ER1	0873	INP1	082A	INP3	0876
PKUP1	0883	INP2	0842	INA	0839
OUT1	0888	MISS	084D	OK	0857
REENT	0845	MGOT	0902	RTN12	087B
INP4	087C	DDLY	088B	DELAY	088D
DEL1	0892	RTN10	0897	SAVE	0898

LST1	08A5	LIST	08A2	LST2	08C5
LST3	08D2	LST4	08D6	LST5	08EB
CNTCRD	08E1	CNT	082B	TOEND	08E8
LST7	090A	MGOT1	0911	CRLF	0B42
LST6	0907	LST8	090C	ERROR	0B2F
CCP3	0942	USING	0935	CCP2	093F
OHL	094F	OHR	0953	TOOUT	095D
O4H	0960	O2H	0965	HD	096B
HDT	097E	HDF	0976	TOOUTS	097B
GET	0985	GET1	0989	COPY	098C
G1	09AF	NEGB	0991	PICKB	0996
SUB2	09DE	SUB4	09CF	GE1	09A1
SETPT	09A4	G2	09B2	SUB3	09CD
G3	09BD	ERR15	09E9	SUB1	09D8
ERR16	09EC	ER9	0A2E	SUB6	09DD
SUB7	0A70	CMP	09F1	CLR	09F7
ALLCLR	0A21	C1	0A01	C3	0A26
C2	0A17	CURS	0A31	SGN	0A4B
MINS	0A5E	PLS	0A59	ZERO	0A61
ER7	0A64	SUB11	0A6E	AND	0A88
AND1	0A8E	OR	0A91	XOR	0A99
RES	0AA1	ER11	0AE7	RE4	0AD7
READ	0AB0	RE3	0AD3	RE2	0AC4
RE1	0AC1	ERR17	0AEA	DEL	0ADC
KEY	0AEF	K0	0AF9	K1	0B02
K2	0B15	UNTIL	0B1A	UNT	0B25
Z	0080	PROG	0C5B		

TOTAL ERROR 00